UNIVERSITY OF CALIFORNIA,
IRVINE

Uncovering Cell State Transition Driver Genes Through Applied Critical Transition Theory

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Mathematical, Computational, and Systems Biology

by

Kai Haskell Silkwood

Dissertation Committee:
Professor Arthur Lander, Chair
Professor Anand Ganesan
Professor Min Zhang

2026

DEDICATION

To

Katherine and Roger

my best friends

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

ACKNOWLEDGMENTS

I thank my current and past lab members, Mika Caldwell, Emmanuel Dollinger, Jimmy Zhu, Dylan Parker, and Surabhi Haniyur for their guidance and support.

Chapter 2 of this dissertation is a reproduction of the material as it appears in Silkwood, K., Dollinger, E., Gervin, J., Atwood, S., Nie, Q., Lander, A. Leveraging gene correlations in single cell transcriptomic data. *BMC Bioinformatics* 25, 305 (2024), used with permission from Springer Nature (CC BY 4.0). The coauthors listed in this publication are Emmanuel Dollinger, Joshua Gervin, Scott Atwood, Qing Nie, and Arthur D. Lander. Arthur Lander directed and supervised research which forms the basis for the dissertation.

Chapter 3 of this dissertation is a reproduction of the material as it appears in Dollinger, E.P., Silkwood, K., Atwood, S., Nie, Q., Lander, A. Statistically principled feature selection for single cell transcriptomics. *BMC Bioinformatics* **26**, 238 (2025), used with permission from Springer Nature (CC BY 4.0). The coauthors listed in this publication are Emmanuel P. Dollinger, Scott Atwood, Qing Nie, and Arthur D. Lander. I appear as second author here, having derived the statistical framework, generated the synthetic datasets, and wrote the R package. Emmanuel Dollinger implemented the method, performed all analyses of real datasets, and wrote the Python package. Arthur Lander directed and supervised research which forms the basis for the dissertation.

Chapter 4 of this dissertation was written in collaboration with the Weinberger lab at the University of Miami, specifically Leor Weinberger and Binyamin Zuckerman, who provided the datasets used and participated in meaningful conversations during the development of the method.

Chapter 5 was written in collaboration with the Marcucio lab at the University of California, San Francisco, specifically Ralph Marcucio, Diane Hu, and Chan Hee Mok. Diane Hu performed experiments validating genes identified in data analysis. Chan Hee Mok performed the initial quality control steps on the data. I thank them for their continued collaboration and support.

Financial support was provided by the University of California, Irvine, NIH grants R01DE019638 and P01CA288662, and training grant GM136624.

Thank you to my friends and family for constantly reminding me there is a bright world outside of academia.

VITA

Kai Silkwood

2026 PhD in Mathematical, Computational, and Systems Biology, University of California, Irvine

2021 Master's Degree in Business and Science, Keck Graduate Institute

2018 Bachelor of Science Degree in Synthetic Chemistry, University of California, San Diego

FIELD OF STUDY
Computational Systems Biology

PUBLICATIONS

First-author publications

- Silkwood, K., Dollinger, E., Atwood S., Nie Q., Lander AD.
Leveraging gene correlations in single-cell transcriptomic data.
BMC Bioinformatics. 2024 Sep 18;25(1):305
DOI: 10.1186/s12859-024-05926-z

Co-author publications

- Derick Han, Heather S. Johnson, Madhuri P. Rao, Gary Martin, Harsh Sancheti, Kai H. Silkwood, Carl W. Decker, Kim Tho Nguyen, Joseph G. Casian, Enrique Cadenas, Neil Kaplowitz,
Mitochondrial remodeling in the liver following chronic alcohol feeding to rats,
Free Radical Biology and Medicine, Volume 102, 2017, Pages 100-110, ISSN 0891-5849
DOI: 10.1016/j.freeradbiomed.2016.11.020
- Dollinger EP, Silkwood K, Atwood S, Nie Q, Lander AD.
Statistically principled feature selection for single-cell transcriptomics.
BMC Bioinformatics. 2025 Oct 2;26(1):238.
DOI: 10.1186/s12859-025-06240-y
- Carl W. Decker, Joseph G. Casian, Kim Tho Nguyen, Luke A. Horton, Madhuri P. Rao, Kai H. Silkwood, and Derick Han
The Critical Role of Mitochondria in Drug-Induced Liver Injury
Molecules, Systems and Signaling in Liver Injury, Wen-Xing Ding and Xiao-Ming Yin, Springer International Publishing, 159-182, 2017
- Kato, K., Nguyen, K.T., Decker, C.W., Silkwood, K.H., Eck, S.M., Hernandez, J.B., Garcia, J., Han, D. Tunneling nanotube formation promotes survival against 5-fluorouracil in MCF-7 breast cancer cells.
FEBS Open Bio. 2022 12: 203-210.
DOI: 10.1002/2211-5463.13324

ABSTRACT OF THE DISSERTATION

Uncovering Cell State Transition Driver Genes Through Applied Critical Transition Theory

by
Kai Silkwood
Doctor of Philosophy in Mathematical, Computational, and Systems Biology
University of California, Irvine, 2026
Professor Arthur Lander, Chair

Cell state transitions can be modeled as critical transitions marked by an increase in gene-gene correlation among a subset of expressed genes. These genes, the cell state transition drivers, are of interest for their role in guiding the cell into a new state in response to some signal. Single cell RNA sequencing (scRNAseq) should allow us to measure these changes in correlation across a transition trajectory. However, the distribution of scRNAseq counts does not fit the expectations that traditional correlation analyses rely on, leading to millions of false discoveries. Here, I develop a method to accurately calculate and statistically validate gene-gene correlations from scRNAseq data. Then, I demonstrate that those same principles can be used to derive statistics for feature selection tasks and that using those statistics leads to better clustering results. Because critical transition theory predicts that correlations among driver genes rise as the cell approaches the tipping point and decay once they enter a new state, transient peaks in correlation should mark candidate transition drivers. Therefore, I propose a method for analyzing gene-gene correlation trajectories which uncovers cell state transition driver genes. This method captures known cell state transition drivers in epithelial to mesenchymal transition and neuromesodermal progenitor differentiation. Finally, I use this method to identify *SHH*-expression driver genes *PITX2* and *NPAS3* in the developing chick frontonasal prominence. The importance of these genes is then experimentally validated through knockout experiments. These

developments allow for principled correlation analyses of scRNAseq datasets and demonstrate the utility of applied critical transition theory in identifying critical transition driver genes.

INTRODUCTION

Cells constantly transition between transcriptional states in response to different stimuli. These transitions can be large and (mostly) irreversible, such as those seen in development where an initially small number of pluripotent cells iteratively differentiate into increasingly diverse and specialized states. Similarly, in the adaptive immune system, naïve B-cells can enter one of many, potentially novel, states in response to invading pathogens. On the other hand, these transitions can also be small and temporary - one example being when cells tune their relative enzyme production rates in response to fluctuations in metabolite availability.

Classically, cell state changes have been conceptualized using Waddington's epigenetic landscape[1]. This landscape is a metaphor likening a transitioning cell to a ball rolling down a hill into different valleys (i.e., different cell states). It describes how pluri- or multipotent cells transition from a state of high cellular potential – where they can become many different cell types – to more restricted states as they differentiate. Along the way, these cells are met with proverbial forks in the road where epigenetic decisions are made which restrict those cells to only the fates which lie further down the selected path. In its initial conception, the epigenetic landscape was phenomenological; it was not known exactly how it could be generated.

This began to change in the 1960's, first with the discovery of the lac operon by Jacob and Monod[2]. From there, it became increasingly evident that a gene's expression could itself regulate the expression of other genes. Further, it was noted that these genes could form arbitrarily complex, feedback loop-enriched networks with one another i.e., gene regulatory networks (GRN)[3]. In the steady state, high-dimensional dynamical networks such as these were predicted to give rise to relatively few stable attractors in gene expression space[3,4]. In other words, boundaries imposed by the GRN constrain the available gene expression space in

which a cell can explore by its intrinsic stochastic fluctuations in gene expression alone. These stable attractors were proposed to define cell states[3,4], thus formalizing the notions of cell state "valleys" conceptualized by Waddington. Later, HL60 cell experiments demonstrated that different trajectories through gene expression space could lead to the same end state, thus offering empirical support for the stable attractor model[5].

Under this model, for a cell to transition in state, it would first need to overcome the barriers placed by the GRN. As alluded to earlier, the expression of any given gene is not constant even among cells of the same state but instead fluctuates stochastically leading to heavy-tailed distributions of transcripts and proteins[6–9]. These stochastic fluctuations are commonly referred to as "intrinsic noise" and can be relatively well modeled by Markovian statistics[7,10–12]. The role of intrinsic noise in governing the fate of stem cells has held considerable interest since at least the 1960's where it was noted that differentiation was stochastic in the growth of spleen erythrogenic colonies[13]. This idea was later reinforced when it was shown that paired hematopoietic progenitors (i.e., two child cells from the same parent) were often variable despite consistent *in vitro* conditions[14]. Similar stochastic differentiation events were observed in various other progenitor systems and, with the advent of high-throughput sequencing technologies, linked to noise in the transcriptome [15–17]. Of course, cell state transitions are not completely random but can be influenced by external stimuli. The treatment of cells with differentiation factors (or de-differentiation factors) has been shown to bias cells toward a particular fate[5,18–20].

As such, intrinsic noise and external stimuli, individually or in coordination, provide the means for the cell to transition from one attractor to another. These signals may bring the cell to a point in gene expression space at the edge of the attractor well – a position in which it is

susceptible to falling into a separate well[4]. Alternatively, these signals may directly alter the landscape itself, destabilizing the prior attractor state and facilitating the transition into another[21,22]. This is known as a critical transition: an attractor in multi-stable state space destabilizes until the cell passes the tipping point. Past the tipping point, the cell has access to a new set of stable attractors that it will rapidly descend into. The critical transition model was supported by experiments performed on multipotent hematopoietic precursor cells[21]. Mojtahedi and colleagues treated these cells with differentiation factors (erythropoietin, granulocyte macrophage colony-stimulating factor and interleukin 3, or a combination of the two treatments) and measured the changes in gene expression over the differentiation timeline across 17 genes using single cell qPCR. The study detected theoretical signatures of attractor destabilization, elevating critical transitions as the leading model for cell state changes.

Critical transitions have been studied in a variety of different fields to better understand the events that lead to and follow catastrophic phenomena in nature and social situations[23–37]. Ecologists study tipping points to forecast how and when changes in environmental conditions, often due to human activity, will lead to major ecological shifts or collapse [28,33,34]; including studies to predict the point in which deforestation of the Amazon rainforest will lead to the destabilization of the hydrological cycle[24], the conditions which lead to the irreversible desertification of previously fertile biomes[23], and the point where rapidly changing climate conditions will lead to loss of ocean ecosystems[37]. Climatologists study tipping points to predict when catastrophic environmental conditions, such as the loss or reversal of ocean currents[26] or abrupt shifts in global climate[29–32] will occur. Economists study tipping points to determine conditions that lead to economic collapse[35] and epidemiologists study the conditions that lead to epidemics and disease re-emergence[36].

Though these systems are quite different conceptually, the early warning signs of their approach to the tipping point are often consistent[38]. First is the observation of increasing variance of quantitative observations as the alternative attractor state becomes more accessible. Second is a "critical slowing down" in which the system's rate of return to the prior state following a perturbation progressively decreases as the transition is approached. Near the tipping point, the attractor begins to flatten (i.e., the restoring force weakens), causing recovery from perturbations to take an increasingly long time. In single cell transcriptomics, where measurements are collected from a static timepoint, these early warning signs manifest as increased heterogeneity of gene expression among cells, and, within cells, an increased correlation among a subset of genes[21]. Increased heterogeneity among cells is the intuitive result from both the flattening of the initial attractor state (allowing cells to spread away from one another) and the emergence of cells either transitioning to or already in the newly accessible state. Less intuitive is the concurrent rise in gene-gene correlations, which emerges from the interplay between the high dimensionality of the data and the fact that observations are drawn from a single timepoint. The transition from one stable state to another involves the coordinated action of many genes. These genes are co-activated, and their concerted activity guides the cell from the initial attractor to another. This set of genes, the transition driver genes, will generally exhibit temporally correlated expression with one another.

Because the transcriptome can only be measured as a snapshot of a particular moment in time, estimating the dynamic activity of a GRN from such data requires assuming approximate ergodicity: that cells undergo equivalent deterministic processes but are captured at different points along them. Measuring transcriptomes across many cells at several timepoints along a transition trajectory, we would expect to observe this increase in correlated expression as we

approach the tipping point. The assumption of ergodicity is widely applied in single cell RNA-sequencing (scRNAseq) analyses to infer GRN activity (see Chapter 1). Of course, ergodicity is never perfectly satisfied in real data, and the consequences of this violation must be accounted for when applying it. The transition driver genes would not only correlate with each other, but also with marker genes for the newly accessible cell state. This is because, within a single timepoint, cells will be in different points along the transition trajectory – including within the new attractor. This heterogeneity in gene expression will manifest as positive correlation among genes.

Measuring the change in correlations across a transition trajectory (defined either using time-series collection or through trajectory inference) would, in theory, allow for the identification of transition driver genes; we would expect that these genes would "peak" in correlation strength with another subset of genes immediately preceding the tipping point. Beyond the increase in understanding of cell state transitions they would provide, identification of transition driver genes has interesting potential applications in medicine. Particularly compelling is in the treatment of degenerative diseases such as Alzheimer's and dementia or those which must overcome known barriers to progression such as cancer. Currently, in cancer this can only be treated through the removal or culling of cells while Alzheimer's has no known curative treatment. In these scenarios, knowledge of the tipping point between healthy and diseased cells may allow us to develop preventative therapies which stabilize the healthy attractor states. Additionally, identification of tipping points may help us to better understand development, particularly early developmental stages where many coinciding cell fate decisions are made. The mechanisms that incur abrupt changes in cellular function, such as the conditions leading to apoptosis, may also be better understood. As evidenced in Chapter 2, however,

conventional correlation methods perform poorly on scRNAseq data, making it necessary to develop a statistically principled framework for quantifying correlation magnitude and its associated null distributions.

This thesis begins with a historical overview of methods for inferring dynamic gene regulatory information from scRNAseq data, critically examining the caveats and advantages of their implementation and highlighting the need for dedicated correlation analysis tools.

I then present BigSur, a methodology which redefines common statistics, such as the Pearson correlation coefficient, to account for the structure of scRNAseq data. I first detail how this methodology allows for the calculation and statistical validation of gene-gene correlations. I show that statistical thresholding with BigSur removes a substantial number of spurious correlations and recovers modules of known co-regulated genes that the Pearson correlation coefficient alone cannot. I follow this with a discussion on how the same statistical principles can be used to improve feature selection in clustering analyses. I demonstrate that using BigSur eliminates substantial noise from selected gene lists, allowing for the recovery of rare cell types that common methods would miss.

Finally, I propose a method which leverages correlations to identify modules of candidate transition driver genes from time series scRNAseq data. I first demonstrate that genes which have many peaks in correlation or participate in strongly peaking gene-gene pairs are enriched for known drivers in an epithelial-mesenchymal transition system. I then establish that modules of co-peaking genes can separate exclusive regulatory programs by uncovering a set of genes which separate the branches of neuromesodermal progenitor cell differentiation. Finally, I apply this method in a system whose transition drivers are less understood: the *SHH*-expressing

frontonasal ectodermal zone of the developing chick. There, I uncover novel driver genes that are confirmed through experimental validation.

References

1. Waddington CH. The strategy of the genes. A discussion of some aspects of theoretical biology. With an appendix by H. Kacser. London: George Allen & Unwin, Ltd.; 1957.

2. Jacob F, Monod J. Genetic regulatory mechanisms in the synthesis of proteins. J Mol Biol. 1961;3:318–56. https://doi.org/10.1016/S0022-2836(61)80072-7

3. Kauffman SA. The Origins of Order: Self-Organization and Selection in Evolution. Oxford University Press; 2023. https://doi.org/10.1093/oso/9780195079517.001.0001

4. Kauffman S. Homeostasis and Differentiation in Random Genetic Control Networks. Nature. 1969;224:177–8. https://doi.org/10.1038/224177a0

5. Huang S, Eichler G, Bar-Yam Y, Ingber DE. Cell fates as high-dimensional attractor states of a complex gene regulatory network. Phys Rev Lett. 2005;94:128701. https://doi.org/10.1103/PhysRevLett.94.128701

6. McAdams HH, Arkin A. Stochastic mechanisms in gene expression. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 1997;94:814–9. https://doi.org/10.1073/pnas.94.3.814

7. Swain PS, Elowitz MB, Siggia ED. Intrinsic and extrinsic contributions to stochasticity in gene expression. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 2002;99:12795–800. https://doi.org/10.1073/pnas.162041399

8. Bahar Halpern K, Tanami S, Landen S, Chapal M, Szlak L, Hutzler A, et al. Bursty Gene Expression in the Intact Mammalian Liver. Mol Cell. 2015;58:147–56. https://doi.org/https://doi.org/10.1016/j.molcel.2015.01.027

9. Beal J. Biochemical complexity drives log-normal variation in genetic expression. Eng Biol. John Wiley & Sons, Ltd; 2017;1:55–60. https://doi.org/https://doi.org/10.1049/enb.2017.0004

10. Peccoud J, Ycart B. Markovian Modeling of Gene-Product Synthesis. Theor Popul Biol. 1995;48:222–34. https://doi.org/10.1006/tpbi.1995.1027

11. Munsky B, Neuert G, van Oudenaarden A. Using Gene Expression Noise to Understand Gene Regulation. Science. American Association for the Advancement of Science; 2012;336:183–7. https://doi.org/10.1126/science.1216379

12. Paulsson J. Models of stochastic gene expression. Phys Life Rev. 2005;2:157–75. https://doi.org/10.1016/j.plrev.2005.03.003

13. Till JE, McCulloch EA, Siminovitch L. A stochastic model of stem cell proliferation, based on the growth of spleen colony-forming cells*. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 1964;51:29–36. https://doi.org/10.1073/pnas.51.1.29

14. Suda T, Suda J, Ogawa M. Disparate differentiation in mouse hemopoietic colonies derived from paired progenitors. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 1984;81:2520–4. https://doi.org/10.1073/pnas.81.8.2520

15. Chang HH, Hemberg M, Barahona M, Ingber DE, Huang S. Transcriptome-wide noise controls lineage choice in mammalian progenitor cells. Nature. 2008;453:544–7. https://doi.org/10.1038/nature06965

16. Piras V, Tomita M, Selvarajoo K. Transcriptome-wide Variability in Single Embryonic Development Cells. Sci Rep. 2014;4:7137. https://doi.org/10.1038/srep07137

17. Pina C, Fugazza C, Tipping AJ, Brown J, Soneji S, Teles J, et al. Inferring rules of lineage commitment in haematopoiesis. Nat Cell Biol. 2012;14:287–94. https://doi.org/10.1038/ncb2442

18. Krantz SB. Erythropoietin. Blood. 1991;77:419–34. https://doi.org/10.1182/blood.V77.3.419.419

19. Yamanaka S. Induction of pluripotent stem cells from mouse fibroblasts by four transcription factors. Cell Prolif. 2008;41 Suppl 1:51–6. https://doi.org/10.1111/j.1365-2184.2008.00493.x

20. Xu J, Lamouille S, Derynck R. TGF-β-induced epithelial to mesenchymal transition. Cell Res. Nature Publishing Group; 2009;19:156–72. https://doi.org/10.1038/cr.2009.5

21. Mojtahedi M, Skupin A, Zhou J, Castaño IG, Leong-Quong RYY, Chang H, et al. Cell Fate Decision as High-Dimensional Critical State Transition. PLOS Biol. Public Library of Science; 2016;14:e2000640. https://doi.org/10.1371/journal.pbio.2000640

22. Huang S, Guo Y-P, May G, Enver T. Bifurcation dynamics in lineage-commitment in bipotent progenitor cells. Dev Biol. 2007;305:695–713. https://doi.org/10.1016/j.ydbio.2007.02.036

23. Reynolds JF, Smith DMS, Lambin EF, Turner BL, Mortimore M, Batterbury SPJ, et al. Global Desertification: Building a Science for Dryland Development. Science. American Association for the Advancement of Science; 2007;316:847–51. https://doi.org/10.1126/science.1131634

24. Lovejoy TE, Nobre C. Amazon Tipping Point. Sci Adv. American Association for the Advancement of Science; 4:eaat2340. https://doi.org/10.1126/sciadv.aat2340

25. Centola D, Becker J, Brackbill D, Baronchelli A. Experimental evidence for tipping points in social convention. Science. American Association for the Advancement of Science; 2018;360:1116–9. https://doi.org/10.1126/science.aas8827

26. Held H, Kleinen T. Detection of climate system bifurcations by degenerate fingerprinting. Geophys Res Lett. John Wiley & Sons, Ltd; 2004;31. https://doi.org/https://doi.org/10.1029/2004GL020972

27. Winkelmann R, Donges JF, Smith EK, Milkoreit M, Eder C, Heitzig J, et al. Social tipping processes towards climate action: A conceptual framework. Ecol Econ. 2022;192:107242. https://doi.org/https://doi.org/10.1016/j.ecolecon.2021.107242

28. Dakos V, Matthews B, Hendry AP, Levine J, Loeuille N, Norberg J, et al. Ecosystem tipping points in an evolving world. Nat Ecol Evol. 2019;3:355–62. https://doi.org/10.1038/s41559-019-0797-2

29. Dakos V, Scheffer M, van Nes EH, Brovkin V, Petoukhov V, Held H. Slowing down as an early warning signal for abrupt climate change. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 2008;105:14308–12. https://doi.org/10.1073/pnas.0802430105

30. Lenton TM. Early warning of climate tipping points. Nat Clim Change. 2011;1:201–9. https://doi.org/10.1038/nclimate1143

31. Lenton TM, Held H, Kriegler E, Hall JW, Lucht W, Rahmstorf S, et al. Tipping elements in the Earth's climate system. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 2008;105:1786–93. https://doi.org/10.1073/pnas.0705414105

32. Lenton TM, Livina VN, Dakos V, van Nes EH, Scheffer M. Early warning of climate tipping points from critical slowing down: comparing methods to improve robustness. Philos Trans R Soc Math Phys Eng Sci. Royal Society; 2012;370:1185–204. https://doi.org/10.1098/rsta.2011.0304

33. Folke C, Carpenter S, Walker B, Scheffer M, Elmqvist T, Gunderson L, et al. Regime Shifts, Resilience, and Biodiversity in Ecosystem Management. Annu Rev Ecol Evol Syst. 2004;35:557–81. https://doi.org/https://doi.org/10.1146/annurev.ecolsys.35.021103.105711

34. Scheffer M, Carpenter S, Foley JA, Folke C, Walker B. Catastrophic shifts in ecosystems. Nature. 2001;413:591–6. https://doi.org/10.1038/35098000

35. Smug D, Ashwin P, Sornette D. Predicting financial market crashes using ghost singularities. PLoS One. Public Library of Science; 2018;13:e0195265. https://doi.org/10.1371/journal.pone.0195265

36. O'Regan SM, O'Dea EB, Rohani P, Drake JM. Transient indicators of tipping points in infectious diseases. J R Soc Interface. 2020;17:20200094. https://doi.org/doi:10.1098/rsif.2020.0094

37. Heinze C, Blenckner T, Martins H, Rusiecka D, Döscher R, Gehlen M, et al. The quiet crossing of ocean tipping points. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 2021;118:e2008478118. https://doi.org/10.1073/pnas.2008478118

38. Scheffer M, Carpenter SR, Lenton TM, Bascompte J, Brock W, Dakos V, et al. Anticipating Critical Transitions. Science. American Association for the Advancement of Science; 2012;338:344–8. https://doi.org/10.1126/science.1225244

CHAPTER 1: AN OVERVIEW AND HISTORY OF TECHNIQUES TO INFER DYNAMIC GENE REGULATORY INFORMATION FROM SINGLE CELL RNA-SEQUENCING DATA

1.1 Single cell RNA-sequencing

The advent of single cell RNA-sequencing has allowed us to measure the transcriptional profiles of cells as individuals. A sequenced tissue will contain cells in many different transcriptional states. Each cell sequenced can be thought of as occupying a particular point in gene expression space. We might expect the majority of cells to occupy positions within the confines of an attractor well, surrounded by cells with closely related cell states, and many fewer to exist in transitional positions between major attractor states. This dynamic has been leveraged to create scRNAseq data analysis platforms of three primary types: cell state characterization, transition trajectory inference, and transition cell identification. In the pursuit of tipping point identification, the power and limitations of the methods in each of these categories is important to understand. Below is a brief overview of the history and development of these methods.

1.2 Cell State Characterization

A cell's state, defined by its gene expression profile, is governed by the dynamic activity of its gene regulatory network (GRN) over time. GRNs are typically modeled as continuous functions, yet we observe discrete cell types—populations of cells that are largely phenotypically identical. This apparent contradiction can be resolved by considering that specific GRN parameters give rise to multiple stable attractor states, where cells persist despite continuous fluctuations in gene expression. Transitions between cell types occur when changes in these parameters—such as the activation or deactivation of certain regulatory regions—destabilize an

existing attractor state. If we could precisely model the GRN and its parameters, we could predict potential cell states, determine the effects of parameter shifts on the system, and identify the tipping points between attractor states. However, complete reconstruction of a GRN would require high throughput time series transcriptomics data from a single cell – a type of data with no known method of collection. Instead, we must rely on inference methods which attempt to produce putative GRN connections from static timepoint data. Pairwise correlation in the expression of genes between cells has long been proposed to hold information about gene regulatory network information [1]. Because of this, methods to infer GRNs typically aim to capture these correlated relationships.

Correlation coefficient-based methods

In the early 2000s, correlation between the expression levels of genes was directly calculated from bulk expression data comparing cells under varying experimental conditions. These correlations were captured using a common correlation coefficient, often the Pearson correlation coefficient. From these correlations, graphical representations of the inferred GRN could be used to visualize relationships between different pairs of genes. Genes would serve as the nodes of the graph and edges would be drawn between pairs of genes which had a correlation coefficient above an arbitrary threshold. Several "gene co-expression network" or "relevance network" inference methods were developed under this framework to detect gene regulatory connections revealed by varying experimental conditions [2, 3]. Commonly, these thresholds were determined by calculating a statistical significance level for the correlation through permutation testing or by using the Fisher formula [4], a means of converting correlation coefficients to *p*-values.

The resulting networks from these methods were defined only by adjacency matrices. Therefore, edges between genes are unweighted and the information that was contained in the magnitude of the correlation between them was lost during graph construction. In response, Zhang and Horvath developed weighted correlation network analysis (WGCNA) [5, 6]. WGCNA simply adds edge weights to the networks built from pairwise gene correlations that have values proportional to the magnitude of the Pearson correlation coefficient. These weighted edges allow for a more fine-grained analysis of these networks by retaining information about the strength of correlation between genes.

From bulk expression data, we can only infer the mean gene regulatory networks activated in sequenced cells and their activation can only be reasonably correlated to the varying experimental conditions. For this reason, bulk data is inappropriate to use when inferring cell state-defining GRNs.

Random transcriptional noise from genes in a cell state-defining GRN are expected to induce weak correlations in expression with one another in homogeneous cell populations [7-13]. If we sample many cells that exist in the same stable attractor well, differences in expression between them can be thought to be primarily due to this transcriptional noise. For this reason, scRNAseq data provides a potentially better platform to characterize GRNs.

In principle, WGCNA techniques can be applied to scRNAseq data to infer gene regulatory relationships. Because each cell can be treated as its own separate condition, WGCNA performed on scRNAseq data could theoretically be used to infer gene regulatory networks from random transcriptional noise - provided the set of cells is homogeneous. However, because scRNAseq data is noisy, sparse, and non-normally distributed, the statistical significance of a given correlation is difficult to determine. As shown through our work, this causes the

aforementioned Fisher formula to fail to produce accurate p-values and leads to massive overestimation in the number of significant correlations [14]. Additionally, the intuition that permutation tests could be used to determine statistical significance for correlations empirically has been shown to be incorrect [15]. Several groups have attempted to reduce sparsity and increase normality of the data by pooling expression information from like cells into "meta-cells" before calculating correlations [16-18]. However, we also find this to be invalid and a major source of erroneous conclusions [14]. Thus, a method to judge the statistical significance of correlations calculated from scRNAseq data must be developed before they can be meaningfully applied (See Results).

Non-negative matrix factorization (NMF)

NMF provides an alternative, nonparametric method to identify correlative relationships in scRNAseq data. First introduced by Paatero and Tapper [19] and further refined by Lee and Seung [20], NMF is a dimensionality reduction technique that approximates a matrix $M \in \mathbb{R}^{m \times n}$ with a low-rank matrix defined by the product of two matrices: $U \in \mathbb{R}^{m \times r}$ and $V \in \mathbb{R}^{r \times n}$.

$$M \approx UV$$

What distinguishes NMF from other linear dimensionality reduction methods, such as principal component analysis (PCA), is the non-negativity constraint on the input data and output factors. This constraint ensures that the components represent additive contributions, facilitating intuitive interpretation. Each value in the resulting matrices reflects the "influence" a specific feature has on the observed data, providing a meaningful decomposition without requiring assumptions about the data structure [21]. Non-negativity is the only requirement of the data. Importantly, correlative relationships between vectors are maintained in the produced factors.

NMF was first used to analyze scRNAseq data by [22] where it was used as a tool to cluster cells. The authors applied NMF to normalized counts matrices to obtain the standard NMF components described above: $U \in \mathbb{R}^{m \times r}$ and $V \in \mathbb{R}^{r \times n}$ where m is the number of cells, n is the number of genes, and r is the estimated number of different cell states present in the dataset. The authors further normalize U so that the components r sum to one across the rows. The values in U are interpreted to be the relative probabilities of each cell belonging to each state. A cell is then labeled as its most probable state. The number of cell states r is decided by a brute force method, where each number between two and ten is tested several hundred times. An information gain score is calculated for each based on the number of cells that cannot be confidently classified into any singular type (i.e., cells that have no value r above a predefined cutoff). The number of states which yields the highest information gain is chosen.

NMF methods for GRN inference are similar in concept but instead utilize the V matrix. cNMF was the first of these programs to be developed [23]. In the V matrix, n is the number of genes and r is the estimated number of GRNs. Every entry in V is then the contribution of each gene to each GRN. Once these relationships are established, cNMF attempts to determine which of these programs are cell state identifying and which are related to general cellular function. This is done by analyzing the associated U matrix. Because r is the same in both U and V, the values in U can be interpreted as an "activation level" of a GRN in a cell.

Notably, NMF techniques have also been used to predict which cells are currently undergoing cell state transition. scRCMF [24] identifies transition cells as those in U which appear to not belong specifically to any one cell state - similar to those which were marked as unclassifiable in [22]. The direction of the transition is predicted using an entropy term (See Transition Cell Identification) which identifies the two states that the transitioning cell is most

similar to and determining the most probable transition direction between the two according to Shannon entropy.

While fringe applications of NMF for scRNAseq analysis have since been developed for a variety of applications [25-29], NMF has failed to become widely adopted as a method for GRN inference or cell state characterization. There are several theoretical considerations to evaluate before utilizing an NMF-based analysis. First, and perhaps most importantly, because correlations are captured within the generated *r* vectors, spurious correlations due to the noisiness and non-normality of scRNAseq data are likely emphasized by the output. False positive correlative relationships vastly outnumber truly correlated expression between genes in scRNAseq data [14]. Second, the magnitude of values in NMF matrices is proportional to the size of the value in the original matrix. Therefore, data must be normalized before analysis to compare the relative contributions of each gene. However, because scRNAseq data is usually extremely sparse, many data normalization techniques alter the underlying distributions of the data. This leads to the introduction of spurious correlations and the loss of true relationships [14]. Third, because of the non-negativity requirement, NMF cannot identify the negative correlations associated with repression between genes [23]. Finally, while not unique to NMF, user selected parameters and random initial seedings can significantly alter the calculation of and relationships between *r* vectors. Users must manually select the number of *r* vectors to calculate. Because seeding can also affect results, algorithmic approaches to picking the correct size of *r* can involve calculating consensus quality scores across hundreds of trial runs [22]. This becomes increasingly concerning as the size of scRNAseq datasets continues to increase, as NMF is an NP-hard computational problem [30].

Conclusion

While scRNAseq data in principle provide a platform with which we can infer cell state-defining GRNs, in practice statistical limitations prevent us from systematically separating the true correlative relationships from the many false positives. Methods to control for false discoveries are necessary before we can properly use scRNAseq data to identify correlated expression – including for the direct identification of tipping points (See Transition Cell Identification).

1.3 Transition Trajectory Inference

Pseudotemporal Ordering (Pseudotime)

Because cells are destroyed in the process, single cell RNA sequencing can only collect data from a single moment in time. However, because each individual cell of a particular type can be considered as a distinct transcriptional state, it is thought that a cell's dynamic shifts in gene regulatory network state can often be inferred from data obtained from many cells. This idea was first developed through the pseudotemporal alignment of cells to infer dynamics in development [31, 32].

Monocle was the first "pseudotime" algorithm, releasing in March of 2014 [31]. The method is based on the concept that each cell in a scRNAseq dataset was sequenced at a distinct point along a timeline of cellular progression. Monocle orders the cells one-dimensionally in a reduced dimensional space by their similarity to one another. Genes included in the analysis are initially preselected using prior knowledge, and independent component analysis (ICA) is used to reduce the dimensions further. Cells which do not fit in the longest determined path are added as branching points along the way.

One month after Monocle's debut, Wanderlust was released [32]. Wanderlust similarly orders the cells into a one-dimensional path but differs from Monocle in how it represents the

data. Rather than using an ICA embedding, Wanderlust creates the path on an l-k-NN graph, a tool for graph iteration where, starting from a k-nearest neighbor graph (k-NN), a node keeps a random set of edges of size *l* from its *k* nearest neighbors. Wanderlust then creates an ensemble of those l-k-NN graphs and takes the consensus best fit line through all points between them. This ensures that there are no branching paths, and that every cell is placed along a single one-dimensional continuum.

While these first unsupervised programs made landmark advancements in our ability to infer dynamics from single timepoint measurements, they are limited by their strict assumption that a single linear path between all cells can accurately capture the cell transitional dynamics in any one dataset. This both is tough to argue as biologically sound, as many transition events are bifurcative, and the results are highly variable and unstable [33]. Additionally, without prior knowledge, they cannot determine the direction of the transition across the determined path [33].

SCUBA, released in December of 2014, provided a supervised approach to the pseudotime problem [34]. SCUBA requires time-course collected data at multiple points along development. At each time point, SCUBA will cluster the cells to track the iterative, relative changes in cell's position to one another and uses that information to infer attractor state bifurcation events. Depending on the number of timepoints collected, SCUBA promises the ability to create more complex lineages where individual attractor states are defined by information from many cells and relatively complex branching bifurcation events can be inferred. However, the reliance on multiple experiments leads to separate issues such as batch effects [35] and prohibitively high costs, especially in situations where the collected cell types may not differentiate into new, obvious attractor states at all.

In 2016, TSCAN was introduced as a solution to address the high variability resulting from unsupervised methods [33]. Instead of directly finding a path between all cells, TSCAN first clusters cells and a path is inferred between the centers of each cluster. After this framing path has been established, the position of each cell in the embedding is used to assign it a pseudotime value. Because of this, the pseudotime of each cell can be interpreted as its position along the transition between large, stable attractor states rather than through each minor state. The results from TSCAN, then, are less variable than those obtained using prior methods and, given that we are typically interested in the transition between major cell states, are often more interpretable and relevant for study. However, TSCAN makes many *ad hoc* decisions in how data is preprocessed and analyzed which can lead to errors in results. For example, before cells are clustered, the genes are hierarchically clustered by correlative expression. The counts in each gene cluster are averaged to produce a single value. Given the large number of false-positive correlative relationships present in scRNAseq data, it is likely that the clusters are composed of many unrelated genes – and even if they were related, taking the mean between all the genes in a cluster dampens the true biological variation in the dataset. Like the prior unsupervised methods, the direction of transition can only be determined using prior knowledge [33].

Each of these pseudotime methods organize cells based on their similarity to one another. Similarity metrics are advantageous because they are non-parametric, allowing algorithm developers to operate without assuming a specific structure for the underlying data. However, this flexibility comes at a cost: the curse of dimensionality. As dimensionality increases, the distance between a point and its nearest neighbor becomes nearly identical to the distance to its furthest neighbor [36]. To address this, pseudotime algorithms calculate distances in a reduced-dimensional space. This introduces a new assumption—that dimensionality reduction techniques

accurately capture the biologically relevant differences separating cell states of interest. As previously discussed, the results of those techniques are heavily influenced by numerous user-defined parameters which can drastically affect the inferred trajectories.

GPfates, introduced in March 2017, attempts to minimize the effects of curse of dimensionality by moving away from a similarity score [37]. Instead, GPfates assumes that cell gene expression during a cell state transition is a Gaussian process. That is, they assume that the gene expression for any one cell can be described by a multivariate normal distribution and that changes to the initial expression over the course of the differentiation can be described using a continuous function. This is equivalent to adding multivariate normally distributed noise to each point along a continuous function. Gaussian process regression is used to assign a pseudotime value to each cell based on this assumption, and this regression can be performed on the high-dimensional data, and multiple curves can be fit to identify bifurcative lineages. In practice, however, these curves are fit in a lower dimensional space and mapped back to the higher ones.

To the best of my knowledge, this was the first pseudotime method to assume and leverage an underlying distribution for transcript counts, which is a notable step forward for the field. However, the Gaussian process assumption is weak based on observations of the distribution of the data [38, 39], to the authors' own admission. This is compounded by the fact that the method, by applying the assumed underlying distribution of counts to the observed counts from sequencing, essentially assumes that sequencing perfectly captures the underlying counts. Additionally, the method runs the high risk of overfitting to non-relevant noise in the data, necessitating a proper dimensionality reduction method, such as feature selection, regardless of the assumed underlying model.

Monocle has since received two major updates in the forms of Monocle2 and Moncole3 [40, 41]. Monocle2 was developed with the goal of improving the Monocle software to detect bifurcative developmental trajectories – something that all methods from SCUBA on have attempted to do. To do this, the developers introduced both a dedicated feature selection tool and a reworked algorithm for ordering cells. The feature selection tool, which they call dpFeature, attempts to identify and leverage the genes that are differently expressed across different clusters detected in a t-distributed stochastic neighbor embedding (tSNE) using density peak clustering. The idea being that noise from irrelevant genes can be removed by using only the expression values from genes that separate clusters of cells to assign pseudotime. To assign pseudotime values, Monocle2 uses a machine learning technique known as RGE. This is an unsupervised method which attempts to learn a graphic representation of the data in low dimensional space and maps it back into the higher dimensions – akin to GPfates's curve fitting.

The primary novelty of Monocle2 comes in the form of its built-in feature selection tool, which theoretically lowers the risk of learned fits being influenced by noise, as compared to methods like GPfates. While the development of a feature selection tool for this kind of task is an advancement towards meaningful dimensionality reduction in the field, Monocle2's implementation has theoretical problems because it learns the differentially expressed genes from a pre-clustered set of data. If all the genes are used to define the tSNE, even if PCA is performed beforehand, one is at high risk for noise-driven placement of cells around the representation. However, if instead one performs some sort of feature selection beforehand, regardless of whether one uses prior knowledge or an algorithmic approach, the results will be biased towards that set of genes – effectively "double dipping." Further, tSNE has been argued to

poorly represent biological relationships between different cells without significant tuning of the data [42, 43].

By this point, the problem of predicting bifurcative lineages had been studied and attempted by several different groups. However, none of these previous iterations had addressed or are able to predict convergent lineages – lineages in which multiple distinct cell states converge into a single state. Monocle3 was developed to tackle this issue [41]. Monocle3 first projects a subset of the data (the first 50 principal components obtained after pruning the data to only the top 5000 most highly dispersed genes) into UMAP space. Following this, a nearest neighbor graph is built on the UMAP projection and Louvain clustering is performed on the graph. The graph is further analyzed to separate clusters into smaller groups if they do not meet a certain connection criterion. Each of these groups is used as a separate entry on which to build a principal graph, similar to Monocle2, but allowing for convergent lineages. Monocle3 is the first of the pseudotime programs to utilize many of the standard practices that are commonly used today in scRNAseq analysis – a feature selection algorithm based on a measurement of dispersion or variance, projection into UMAP space rather than TSNE, and an algorithmic approach of further separating clusters on a graph by thresholding the number of edges between nodes. While initially UMAP was thought to be superior to TSNE, it is subject to the same issues and it is unclear if it performs better [42, 43]. Additionally, dispersion-based feature selection methods using a high number of genes can lead to highly variable and poor clustering results [44].

While pseudotime methods have improved significantly since their inception, the assumptions they make are strong and limit their applicability. Even modern pseudotime methods assume that the scRNAseq dataset is composed of only cells that can differentiate into

one another in some way and are therefore inappropriate to use in experiments where the cell state composition is unknown. Additionally, as they are reliant on projections of scRNAseq data into lower dimensions, they remain highly dependent on our ability to capture meaningful biological states within the lower dimensional space – which in and of itself is a multi-step process where changes at any level can produce variable results.

RNA Velocity

As discussed, because they require some amount of knowledge about the states present in the data, pseudotime methods rely on the output of several prior analytical steps (i.e., normalization, low dimensional projections, pre-clustered cells) to make their predictions which can lead to a high amount of variability in the resulting pseudotemporal assignments. In a setting where the present states are unknown or are fairly homogeneous, it would be ideal to use a method that was independent of the counts matrix itself and, therefore, not dependent on variable analytic results. RNA velocity methods promise to do just that [45, 46].

RNA velocity is a technique used to assign a transitional direction and magnitude to cells in the dataset. It achieves this by using the observation that the ratio of spliced to unspliced reads for any given gene differs from cell to cell. The splicing of introns does not occur instantaneously after transcription, but in a process that occurs over a timescale of up to many minutes [47]. Because the process of splicing transcripts takes time, the ratio of spliced to unspliced transcripts of a gene gives information on how far a cell has progressed after that gene has turned on. These ratios are calculated before any data processing has occurred, separating the transitional dynamics from data transformations used during analysis. If an RNA velocity is calculated to be positive, it can be inferred that the gene is being upregulated in the cell. Likewise, a negative RNA velocity is indicative of down regulation.

La Manno et al were the first to leverage this to predict transitional dynamics in single cells [45]. To do this, they utilize the following simple model:

$$\frac{du}{dt} = \alpha(t) - \beta(t)\, u(t)$$

$$\frac{ds}{dt} = \beta(t)\, u(t) - \gamma(t)\, s(t)$$

Where *u(t)* is the number of unspliced transcripts, *s(t)* is the number of spliced transcripts, $\alpha(t)$ is the rate of transcription, $\beta(t)$ is the rate of splicing, and $\gamma(t)$ is the rate of mRNA degradation. When the rates of transcription and degradation are assumed to be constant, these equations can be solved analytically, and the velocity can be calculated gene by gene. A single parameter, γ, must be estimated empirically from the data for each gene. This estimation is carried out using a linear regression fit. This implementation allowed for the inference of gene expression trajectories without relying on the counts matrix – effectively separating trajectory inference from analytic techniques. However, these predictions are only viable under two specific conditions which limit their utility. First, the equations assume that the expression of a gene is independent of the others. This is not generally the case in biology, where gene expression is the result of many codependent processes. From a technical standpoint as well, if too many genes that are correlated in expression are included as genes used in calculated RNA velocity, it likely biases the results heavily in the direction of certain gene regulatory programs, unintentionally. Second, the estimation of γ is made under the assumption that all measured expression is from genes at steady state and that all genes have an equivalent splicing rate [46]. These assumptions are violated on datasets with heterogeneous cell populations – the type of dataset necessary to analyze transition and meta-stable states. It also should not be applied to very lowly expressed genes. These genes, due to the burstiness of gene expression, will have very sparse counts

vectors and, therefore, any cell which does happen to express that gene will be far outside of the statistical steady state.

Because this initial method could not accurately calculate an RNA velocity for cells in transition states, it provided limited utility in the analysis of cell state transitions. Bergen et al developed the method scVelo to tackle this issue [46]. scVelo utilizes the same simple model as the previous method, but no longer assumes that the rate of transcription is constant. Instead, it is a function of two cell specific latent variables: a discrete transcriptional state and a cell specific point in continuous time. These parameters are learned iteratively using expectation-maximization and can fully resolve the values of γ for each cell and gene pair once they are learned. This removes the dependency on linear regression estimation for the calculation of RNA velocity and removes the restraint of only working on cells at steady state.

RNA velocity is a great step forward in our ability to infer cell state trajectories from scRNAseq data and, unlike pseudotime techniques, does not rely on any transformations to or analyses of the counts data to make predictions on trajectories. However, because velocity can be calculated for every gene, users need to be careful in how they apply the results to their own analyses. Regulatory or cyclically expressed genes (e.g., cell cycle-related genes) included in calculating an overall trajectory can lead to results that, while not exactly wrong, are uninteresting in nature. Additionally, because RNA velocity calculations all assume that gene counts are independently distributed, it is easy to bias the results towards specific regulatory networks if many participatory genes are included. RNA velocity is best applied when the user can make informed decisions about which genes to use in trajectory inference and would be best paired with a principled feature selection tool and an accurate gene-gene correlation calculation.

Conclusion

Because scRNAseq data often contains information about cells at many different related states, trajectory inference between these states is a viable, but tenuous, replacement for time-series sequencing data. Users must be cautious to only use tools which make assumptions that apply to their particular dataset. Notably, these techniques only infer trajectories one-dimensionally and contain no information about the underlying potential landscape. To identify tipping points, a trajectory inference method would need to be paired with a method which can characterize cell state.

1.4 Transition Cell Identification

The speed of a cell state transition is limited by the time required for many cellular processes (e.g., transcription, translation, chromatin binding, etc.). Because of this, once a cell passes a tipping point, they are not expected to immediately exhibit new phenotype-affirming GRN behavior. Instead, as cells approach and pass a tipping point, they are thought to transition smoothly and continuously from one stable attractor to the next. These cells are said to exist in a transition state [48, 49]. Transition state cells have been observed in a variety of systems to exhibit a high level of heterogeneity in expression relative to the rest of the population [50-54] which suggests that high heterogeneity may be generalizable to all cell state transitions [55]. Because of this, methods to identify transition state cells often try to quantify each cell's heterogeneity.

Entropy

Entropy techniques are based around the observation that, even in a homogeneous cell population, gene expression in an individual cell is probabilistic while the aggregate of all cells in a population exhibits stable expression [56]. In 2013, MacArthur and Lemischka likened this

to systems studied using statistical mechanics and suggested entropy as an alternative method of quantifying heterogeneity in gene expression [57].

Entropy is a measure of the disorder or uncertainty in a system, like a measurement of variance. Most commonly, entropy is calculated using the Shannon entropy formula:

$$H(q) = -k \sum_{x} q(x) \log q(x)$$

where *q(x)* is a discrete probability mass function and *k* is typically Boltzmann's constant. The entropy of a population of cells is equivalent to the uncertainty in the gene expression state of a cell drawn at random. Regulatory relationships between genes introduce correlations and lower the entropy of a population [57]. This suggests that cells at equilibrium, those with stable gene regulatory networks, will have lower entropy than those with weaker constraints. Because of this, MacArthur and Lemischka argue that cells with high potential for undergoing cell state transitions, such as pluripotent stem cells, will exhibit higher entropy overall and in more diverse patterns than stably differentiated cells.

With the advent of single cell RNA sequencing, Teschendorff and Enver developed SCENT to estimate the differentiation potential and pseudotime position of a cell from its gene expression entropy [58]. SCENT first assigns edge weights to the edges of a graph described by the adjacency matrix of a pre-selected protein-protein interaction (PPI) network. Using the assumption that genes that are more highly expressed are more likely to interact, for a single cell, the probability gene *i* interacting with gene *j* in the sample is

$$q_{ij} = \frac{x_j}{(Ax)_i}$$

where *x* is the number of transcripts and *A* is zero if the genes are not shown to interact in the chosen PPI network and one if they do. Single cell entropy is then calculated as

$$H(q) = -\sum_{i=1}^{n} \pi_i \sum_{j \in N} q_{ij} \log q_{ij}$$

The number of underlying meta-stable states (*K*) for a given population is calculated by fitting a Gaussian mixture model. A probability distribution q_k for each meta-stable state *k* can be fit from the q_{ij} of all cells in state *k*. The entropy of the population of cells can then be defined as

$$H(q) = -\frac{1}{\log K} \sum_{k=1}^{K} q_k \log q_k$$

Populations of cells with higher values of *H(q)* are said to have higher differentiation potential. Lineages can be reconstructed by clustering the cells and determining "landmark" genes and comparing individual entropy states across those specific genes.

A few months after the release of SCENT, Guo *et al* published their own entropy-based algorithm for the determination of cell differentiation potential and lineages: SLICE [59]. SLICE begins by calculating an entropy state for each cell which the authors term scEntropy and from it approximates a differentiation potential using the same principles as SCENT. Unlike SCENT, entropy is not calculated using a known protein-protein interaction network but, instead, calculated for a particular functional group of genes based on gene ontology. Entropy values for these different functional groups are then used to approximate a set of stable states among the different cells in a particular pre-determined cluster and constructs a minimum spanning tree to create putative lineages.

The premise of calculating the entropy over a predefined set of genes and using them to predict differentiation potentials has since been expanded into several more programs, each making changes to methods of calculating entropy across the set of genes and/or the method of determining relevant genes to utilize [60-62]. Each of these methods requires a predefined set of

genes as input which biases the results. However, there have been multiple programs developed which attempt to calculate and utilize entropy in a more unbiased manner [63-67].

The first of these to be published was StemID in late 2016 [64]. StemID seeks to identify stem cells hidden among a pre-clustered population of cells using a combination of graph theory and entropy measurements. Cells which have the highest relative entropy and "link score" – a quantification similar to the degree of a node on a graph – are marked to be the stem cells in the population. StemID's method of approximating a lineage graph enforces the assumption that a cell in any cluster has the possibility of differentiating into any other cluster state.

To achieve a more fine-grained assessment of differentiation potential, Setty et al. introduced their program, Palantir, in 2019 [65]. Palantir models differentiation as a Markov process, where each node in a nearest-neighbor graph represents a cell state. Starting from a chosen initial node, the program calculates the probabilistic fate of the cell by modeling a Markov chain that progresses through the remaining states. This approach assumes that cells in the sample are differentiating, fate decisions are unidirectional, and the probability of any step is independent of the states previously visited. Random walks are employed to identify absorbing states, after which edges leading to these states are removed from the graph. Entropy is then calculated for each node based on the probabilities of traversing its connected edges. By combining the entropy of a cell state with the probabilities of its potential fate decisions, this approach provides greater insight into the likelihood of a cell differentiating in a specific direction compared to using entropy alone. However, this method sacrifices information about the specific genes involved in driving these differentiation processes.

MuTrans, developed by Zhou et al., was created with the goal of directly identifying cells currently undergoing transitions within the dataset in a multiscale fashion [67]. MuTrans

assumes that the stochastic dynamics during cell-fate decisions can be modeled as random walks by individuals across a random walk transition probability matrix (rwTPM). MuTrans estimates rwTPM for transition processes across three different scales: cell to cell transitions, cluster to cluster transitions, and cell to cluster transitions. Entropy is then calculated using the transition probabilities from any individual to another. The individuals with the largest entropies are labeled as transition state cells.

While entropy-based tools have not been directly applied to the detection of tipping points, they offer an intuitive method to do so. If an entropy measurement can accurately capture gene expression heterogeneity, we would expect the entropy to rise until it reaches the tipping point and fall as the cell enters the new stable state. Given enough cells, a tipping point could be determined as the point of maximum entropy.

However, entropy methods are highly dependent on the quality of input data due to their reliance on clustering cells, typically using a nearest neighbor graph, and the labeling of cell states prior to analysis. In the entropy framework, the nodes of the graph each represent a meta-stable state of the cells, and the entropy of each state can be calculated accordingly. Because of this, the predictive power of entropy in identifying transitioning cells depends entirely on the clustering algorithm's ability to position cells in a biologically meaningful manner.

The construction of biologically meaningful graphs, and thus the overall effectiveness of entropy-based methods, depends on several interrelated factors. The data quality and relevance of the genes included in the analysis play a critical role, as does the choice of dimensionality reduction techniques and their associated parameters. Additionally, the specific metrics and thresholds used to construct the nearest-neighbor graph influence the outcomes. Another consideration is the potential variability introduced by random seeding during centroid

initialization [44]. These dependencies can pose challenges for the application of entropy-based methods in situations where cell states and their relationships are poorly understood (i.e., those where we are reliant on unsupervised techniques to identify cell states) making them difficult to apply broadly. Additionally, entropy techniques can only be used to compare the likelihood of an individual cell state to transition in a relative sense to the other states present on the graph. A high relative entropy does not necessarily indicate the presence of transition cells on its own.

Correlation coefficient-based methods

Heterogeneity in gene expression will lead to differences in the correlations between genes in a transition cell compared to a cell in a stable state. If these differences have a predictable pattern, they could in principle be used to identify transition cells.

Mojtahedi et al derived from phenomenological models of dynamical systems that this pattern should present as a concurrent decrease in cell-cell correlations and an increase in gene-gene correlations [68]. A decrease in cell-cell correlations is obvious as the stable attractor state becomes destabilized prior to a noise-induced cell state transition. Less obvious is a rationale for an increase in gene-gene correlations. The authors argue that the gene-gene correlations observed from cells in a stable state are primarily the weak, noise-induced correlations. As a cell begins a state transition, these correlations are overwhelmed by stronger correlations induced by coordinated changes in GRN structure. Based on these theoretical considerations, the authors proposed an index for critical transitions (I_C) as a metric to detect critical transitions:

$$I_C(t) = \frac{\langle |R(g_i, g_j)| \rangle}{\langle R(S^k, S^l) \rangle}$$

where g are gene vectors, S are the cell vectors at a sampling time t, and R denotes the function to calculate the Pearson's correlation coefficient. I_C is expected to be at its maximum at the time t where a critical transition occurs. They verified these claims by studying the transition of

progenitor blood cells into myeloid or erythroid cells using single-cell qPCR across multiple time points during differentiation.

Based on the I_C score, Yang et al proposed BioTIP as a method to detect transition cells in pre-clustered scRNAseq data [69]. From a selected set of highly variable genes, BioTIP first attempts to identify specific genes that may be more informative for identifying transition states cells. It does this by comparing the standard deviation in expression of each gene in a specific cluster to the aggregate standard deviation in the remaining clusters. These genes are then used to create correlation networks for each cluster where edges are drawn between correlated genes according to the Pearson correlation coefficient. The retrieved networks are then used to identify putative transition cell clusters using a dynamical network biomarker (DNB) score [70] that has been modified for use in single cell experiments:

$$DNB.score(X^r, m) = Avg_{i \in m}\left(sd(g_i)\right) \frac{Avg_{i \in m, j \in m}\left(\left|PCC\left({g_i}^r, {g_j}^r\right)\right|\right)}{Avg_{i \in m, j \in m^C}\left(\left|PCC\left({g_i}^r, {g_j}^r\right)\right|\right)}$$

where X^r is the matrix of normalized expression counts for cluster r, m is the set of genes comprising the correlation network inferred earlier for cluster r, m^C is a set genes genes complement to m of equal size (bootstrapped 1000 times), and g is any gene. In other words, putative transition cell clusters are identified by finding clusters that have higher than expected correlation between genes within the identified network of informative genes as compared to their correlations to genes outside of that network.

These putative transition clusters are then tested for significance using a modified I_C score, the $I_C.shrink$. The $I_C.shrink$ replaces the regular Pearson correlation coefficient (PCC) with a shrinkage estimator [71]. The shrinkage estimator is a linear combination of the PCC matrix, and a reduced dimensional form of the PCC matrix labeled as T. T is known as a target matrix, and an algorithmic approach is taken to shrink the PCC matrix towards T. The contribution of

each matrix is determined by the number of cells in any specific cluster. Clusters that have many cells will have a shrinkage estimator dominated by the original PCC matrix, while smaller clusters will have more contribution by the target matrix. This has been shown to reduce variability in results in scenarios where there are low numbers of cells with some tradeoff in accuracy [71]. The authors choose to use this shrinkage estimator because clusters can be of widely varying sizes. The smaller clusters will have higher variability in the PCC matrix estimates which can lead to more extreme correlation values, possibly biasing the I_C metric to be higher. Clusters that are found to be significant by both the *DNB.score* and I_C*.shrink* are identified as transition state cells.

BioTIP proposes a conceptual pipeline that one can use to identify transition state cells and introduces useful metrics such as PCC shrinkage estimates to more readily compare clusters of varying size on a more equal footing. However, their specific implementation is rife with problems that severely limit the accuracy of its results. First, the initial feature selection used is not generally accurate and the choice to use the static number of 3000 genes is unprincipled which can lead to the selection of many irrelevant genes while passing over many that actually vary by cell state [44]. Second, statistical significance for the PCC is determined by the standard Fisher formula which we have shown to be extremely poor in assessing significance [14]. Because false positive correlations are expected to vastly outnumber the true positives, the *DNB.score* and I_C*.shrink* score will be highly inaccurate. The problem is compounded by the fact that these false correlations can have very high PCC values. Next, BioTIP is reliant on pre-clustered data. As previously discussed, many factors can lead to variation in the quality of clustering – including something as simple as the random initially chosen cell [44]. Finally, BioTIP is limited to the identification of transition states on the scale of clusters. This requires a

cluster in the processed data to be primarily composed of transition state cells, which, if lacking ground truth knowledge, can only be determined by using a similar transition cell identification tool.

Conclusion

scRNAseq data provides a promising platform for the detection of transition state cells by measuring heterogeneity in gene expression. However, the techniques that have been developed until now are limited due to the requirement of working on pre-clustered data. Beyond the variability in clustering results, in principle this limitation silos these techniques to only work when analyzing datasets where transitions are already known to be occurring. If the reliance on clustering results can be avoided, both entropy and correlation-based identification of transition states would provide promising tools to generally characterize tipping points in cell state transitions.

1.5 The Utility of Correlation-Based Transition Driver Gene Identification

While many techniques have been developed to identify transitioning or transition state cells, but identifying the genes that drive those transitions has received far less attention. Correlation-based methods are no exception: they have been developed primarily to detect transition-state clusters rather than their drivers. Metrics such as the *DNB.score* and I_{C}*.shrink* do compute gene-gene correlations internally, but because they do not account for the structure of scRNAseq data, the positive correlations are prone to inflation in both number and magnitude. Two things were therefore needed: a method that both measures and statistically evaluates gene-gene correlations in scRNAseq data, and one that uses those correlations to identify transition driver genes based on critical transition theory.

1.6 References

1. Eisen MB, Spellman PT, Brown PO, Botstein D: Cluster analysis and display of genome-wide expression patterns. Proceedings of the National Academy of Sciences 1998, 95:14863-14868.

2. Butte AJ, Kohane IS: Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. Pac Symp Biocomput 2000:418-429.

3. Carter SL, Brechbühler CM, Griffin M, Bond AT: Gene co-expression network topology provides a framework for molecular characterization of cellular state. Bioinformatics 2004, 20:2242-2250.

4. Fisher RA: Statistical methods for research workers. In Breakthroughs in statistics: Methodology and distribution. Springer; 1970: 66-70

5. Zhang B, Horvath S: A general framework for weighted gene co-expression network analysis. Stat Appl Genet Mol Biol 2005, 4:Article17.

6. Langfelder P, Horvath S: WGCNA: an R package for weighted correlation network analysis. BMC Bioinformatics 2008, 9:559.

7. Pedraza JM, van Oudenaarden A: Noise Propagation in Gene Networks. Science 2005, 307:1965-1969.

8. Becskei A, Kaufmann BB, van Oudenaarden A: Contributions of low molecule number and chromosomal positioning to stochastic gene expression. Nature Genetics 2005, 37:937-944.

9. Stewart-Ornstein J, Weissman Jonathan S, El-Samad H: Cellular Noise Regulons Underlie Fluctuations in Saccharomyces cerevisiae. Molecular Cell 2012, 45:483-493.

10. Padovan-Merhar O, Raj A: Using variability in gene expression as a tool for studying gene regulation. WIREs Systems Biology and Medicine 2013, 5:751-759.

11. Munsky B, Neuert G, van Oudenaarden A: Using Gene Expression Noise to Understand Gene Regulation. Science 2012, 336:183-187.

12. Warmflash A, Dinner AR: Signatures of combinatorial regulation in intrinsic biological noise. Proceedings of the National Academy of Sciences 2008, 105:17262-17267.

13. Gupta A, Martin-Rufino JD, Jones TR, Subramanian V, Qiu X, Grody EI, Bloemendal A, Weng C, Niu S-Y, Min KH, et al: Inferring gene regulation from stochastic transcriptional variation across single cells at steady state. Proceedings of the National Academy of Sciences 2022, 119:e2207392119.

14. Silkwood K, Dollinger E, Gervin J, Atwood S, Nie Q, Lander AD: Leveraging gene correlations in single cell transcriptomic data. BMC Bioinformatics 2024, 25:305.
15. DiCiccio CJ, Romano JP: Robust Permutation Tests For Correlation And Regression Coefficients. Journal of the American Statistical Association 2017, 112:1211-1220.

16. Morabito S, Miyoshi E, Michael N, Shahin S, Martini AC, Head E, Silva J, Leavy K, Perez-Rosendahl M, Swarup V: Single-nucleus chromatin accessibility and transcriptomic characterization of Alzheimer's disease. Nature Genetics 2021, 53:1143-1155.

17. Morabito S, Reese F, Rahimzadeh N, Miyoshi E, Swarup V: hdWGCNA identifies co-expression networks in high-dimensional transcriptomics data. Cell Reports Methods 2023, 3.

18. Xu H, Hu Y, Zhang X, Aouizerat BE, Yan C, Xu K: A novel graph-based k-partitioning approach improves the detection of gene-gene correlations by single-cell RNA sequencing. BMC Genomics 2022, 23:35.

19. Paatero P, Tapper U: Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. Environmetrics 1994, 5:111-126.

20. Lee DD, Seung HS: Learning the parts of objects by non-negative matrix factorization. Nature 1999, 401:788-791.

21. Babji S, Tangirala AK: Source separation in systems with correlated sources using NMF. Digital Signal Processing 2010, 20:417-432.

22. Shao C, Höfer T: Robust classification of single-cell transcriptome data by nonnegative matrix factorization. Bioinformatics 2017, 33:235-242.

23. Kotliar D, Veres A, Nagy MA, Tabrizi S, Hodis E, Melton DA, Sabeti PC: Identifying gene expression programs of cell-type identity and cellular activity with single-cell RNA-Seq. eLife 2019, 8:e43803.

24. Zheng X, Jin S, Nie Q, Zou X: scRCMF: Identification of Cell Subpopulations and Transition States From Single-Cell Transcriptomes. IEEE Transactions on Biomedical Engineering 2020, 67:1418-1428.

25. Hozumi Y, Wei G-W: Analyzing single cell RNA sequencing with topological nonnegative matrix factorization. Journal of Computational and Applied Mathematics 2024, 445:115842.

26. Qiu Y, Yan C, Zhao P, Zou Q: SSNMDI: a novel joint learning model of semi-supervised non-negative matrix factorization and data imputation for clustering of single-cell RNA-seq data. Briefings in Bioinformatics 2023, 24.

27. Townes FW, Engelhardt BE: Nonnegative spatial factorization applied to spatial genomics. Nature Methods 2023, 20:229-238.

28. Qian Y, Zou Q, Zhao M, Liu Y, Guo F, Ding Y: scRNMF: An imputation method for single-cell RNA-seq data by robust and non-negative matrix factorization. PLOS Computational Biology 2024, 20:e1012339.

29. Wu W, Ma X: Network-Based Structural Learning Nonnegative Matrix Factorization Algorithm for Clustering of scRNAseq Data. IEEE/ACM Transactions on Computational Biology and Bioinformatics 2023, 20:566-575.

30. Wang YX, Zhang YJ: Nonnegative Matrix Factorization: A Comprehensive Review. IEEE Transactions on Knowledge and Data Engineering 2013, 25:1336-1353.

31. Trapnell C, Cacchiarelli D, Grimsby J, Pokharel P, Li S, Morse M, Lennon NJ, Livak KJ, Mikkelsen TS, Rinn JL: The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. Nature Biotechnology 2014, 32:381-386.

32. Bendall Sean C, Davis Kara L, Amir E-ad D, Tadmor Michelle D, Simonds Erin F, Chen Tiffany J, Shenfeld Daniel K, Nolan Garry P, Pe'er D: Single-Cell Trajectory Detection Uncovers Progression and Regulatory Coordination in Human B Cell Development. Cell 2014, 157:714-725.

33. Ji Z, Ji H: TSCAN: Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis. Nucleic Acids Research 2016, 44:e117-e117.

34. Marco E, Karp RL, Guo G, Robson P, Hart AH, Trippa L, Yuan G-C: Bifurcation analysis of single-cell gene expression data reveals epigenetic landscape. Proceedings of the National Academy of Sciences 2014, 111:E5643-E5650.

35. Haghverdi L, Lun ATL, Morgan MD, Marioni JC: Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. Nature Biotechnology 2018, 36:421-427.

36. Beyer K, Goldstein J, Ramakrishnan R, Shaft U: When Is "Nearest Neighbor" Meaningful? In Database Theory — ICDT'99; 1999//; Berlin, Heidelberg. Edited by Beeri C, Buneman P. Springer Berlin Heidelberg; 1999: 217-235.

37. Lönnberg T, Svensson V, James KR, Fernandez-Ruiz D, Sebina I, Montandon R, Soon MSF, Fogg LG, Nair AS, Liligeto UN, et al: Single-cell RNA-seq and computational analysis using temporal mixture modeling resolves TH1/TFH fate bifurcation in malaria. Science Immunology 2017, 2:eaal2192.

38. Bahar Halpern K, Tanami S, Landen S, Chapal M, Szlak L, Hutzler A, Nizhberg A, Itzkovitz S: Bursty Gene Expression in the Intact Mammalian Liver. Molecular Cell 2015, 58:147-156.

39. Beal J: Biochemical complexity drives log-normal variation in genetic expression. Engineering Biology 2017, 1:55-60.

40. Qiu X, Mao Q, Tang Y, Wang L, Chawla R, Pliner HA, Trapnell C: Reversed graph embedding resolves complex single-cell trajectories. Nature Methods 2017, 14:979-982.

41. Cao J, Spielmann M, Qiu X, Huang X, Ibrahim DM, Hill AJ, Zhang F, Mundlos S, Christiansen L, Steemers FJ, et al: The single-cell transcriptional landscape of mammalian organogenesis. Nature 2019, 566:496-502.

42. Kobak D, Berens P: The art of using t-SNE for single-cell transcriptomics. Nature Communications 2019, 10:5416.

43. Chari T, Pachter L: The specious art of single-cell genomics. PLOS Computational Biology 2023, 19:e1011288.

44. Dollinger E, Silkwood K, Atwood S, Nie Q, Lander AD: Statistically principled feature selection for single cell transcriptomics. bioRxiv 2024:2024.2010.2011.617709.

45. La Manno G, Soldatov R, Zeisel A, Braun E, Hochgerner H, Petukhov V, Lidschreiber K, Kastriti ME, Lönnerberg P, Furlan A, et al: RNA velocity of single cells. Nature 2018, 560:494-498.

46. Bergen V, Lange M, Peidli S, Wolf FA, Theis FJ: Generalizing RNA velocity to transient cell states through dynamical modeling. Nature Biotechnology 2020, 38:1408-1414.

47. Alpert T, Herzel L, Neugebauer KM: Perfect timing: splicing and transcription rates in living cells. WIREs RNA 2017, 8:e1401.

48. Brackston RD, Lakatos E, Stumpf MPH: Transition state characteristics during cell differentiation. PLOS Computational Biology 2018, 14:e1006405.

49. Moris N, Pina C, Arias AM: Transition states and cell fate decisions in epigenetic landscapes. Nature Reviews Genetics 2016, 17:693-703.

50. Hu M, Krause D, Greaves M, Sharkis S, Dexter M, Heyworth C, Enver T: Multilineage gene expression precedes commitment in the hemopoietic system. Genes Dev 1997, 11:774-785.

51. Goolam M, Scialdone A, Graham SJL, Macaulay IC, Jedrusik A, Hupalowska A, Voet T, Marioni JC, Zernicka-Goetz M: Heterogeneity in Oct4 and Sox2 Targets Biases Cell Fate in 4-Cell Mouse Embryos. Cell 2016, 165:61-74.

52. Brunskill EW, Park JS, Chung E, Chen F, Magella B, Potter SS: Single cell dissection of early kidney development: multilineage priming. Development 2014, 141:3093-3101.

53. Miyamoto T, Iwasaki H, Reizis B, Ye M, Graf T, Weissman IL, Akashi K: Myeloid or lymphoid promiscuity as a critical step in hematopoietic lineage commitment. Dev Cell 2002, 3:137-147.

54. Laslo P, Spooner CJ, Warmflash A, Lancki DW, Lee H-J, Sciammas R, Gantner BN, Dinner AR, Singh H: Multilineage Transcriptional Priming and Determination of Alternate Hematopoietic Cell Fates. Cell 2006, 126:755-766.

55. Martinez Arias A, Brickman JM: Gene expression heterogeneities in embryonic stem cell populations: origin and function. Curr Opin Cell Biol 2011, 23:650-656.

56. Hayashi K, de Sousa Lopes SMC, Tang F, Lao K, Surani MA: Dynamic equilibrium and heterogeneity of mouse pluripotent stem cells with distinct functional and epigenetic states. Cell Stem Cell 2008, 3:391-401.

57. MacArthur BD, Lemischka IR: Statistical mechanics of pluripotency. Cell 2013, 154:484-489.

58. Teschendorff AE, Enver T: Single-cell entropy for accurate estimation of differentiation potency from a cell's transcriptome. Nature Communications 2017, 8:15599.

59. Guo M, Bao EL, Wagner M, Whitsett JA, Xu Y: SLICE: determining cell differentiation and lineage based on single cell entropy. Nucleic Acids Res 2017, 45:e54.

60. Liu J, Song Y, Lei J: Single-Cell Entropy to Quantify the Cellular Order Parameter from Single-Cell RNA-Seq Data. Biophysical Reviews and Letters 2020, 15:35-49.

61. Shi J, Teschendorff AE, Chen W, Chen L, Li T: Quantifying Waddington's epigenetic landscape: a comparison of single-cell potency measures. Briefings in Bioinformatics 2020, 21:248-261.

62. Ye Y, Yang Z, Zhu M, Lei J: Using single-cell entropy to describe the dynamics of reprogramming and differentiation of induced pluripotent stem cells. International Journal of Modern Physics B 2020, 34:2050288.

63. Gong W, Rasmussen TL, Singh BN, Koyano-Nakagawa N, Pan W, Garry DJ: Dpath software reveals hierarchical haemato-endothelial lineages of Etv2 progenitors based on single-cell transcriptome analysis. Nature Communications 2017, 8:14362.

64. Grün D, Muraro Mauro J, Boisset J-C, Wiebrands K, Lyubimova A, Dharmadhikari G, van den Born M, van Es J, Jansen E, Clevers H, et al: De Novo Prediction of Stem Cell Identity using Single-Cell Transcriptome Data. Cell Stem Cell 2016, 19:266-277.

65. Setty M, Kiseliovas V, Levine J, Gayoso A, Mazutis L, Pe'er D: Characterization of cell fate probabilities in single-cell data with Palantir. Nature Biotechnology 2019, 37:451-460.

66. Wang S, Drummond ML, Guerrero-Juarez CF, Tarapore E, MacLean AL, Stabell AR, Wu SC, Gutierrez G, That BT, Benavente CA, et al: Single cell transcriptomics of human epidermis identifies basal stem cell transition states. Nature Communications 2020, 11:4239.
67. Zhou P, Wang S, Li T, Nie Q: Dissecting transition cells from single-cell transcriptome data through multiscale stochastic dynamics. Nature Communications 2021, 12:5609.

68. Mojtahedi M, Skupin A, Zhou J, Castaño IG, Leong-Quong RYY, Chang H, Trachana K, Giuliani A, Huang S: Cell Fate Decision as High-Dimensional Critical State Transition. PLOS Biology 2016, 14:e2000640.

69. Yang XH, Goldstein A, Sun Y, Wang Z, Wei M, Moskowitz Ivan P, Cunningham John M: Detecting critical transition signals from single-cell transcriptomes to infer lineage-determining transcription factors. Nucleic Acids Research 2022, 50:e91-e91.

70. Chen L, Liu R, Liu Z-P, Li M, Aihara K: Detecting early-warning signals for sudden deterioration of complex diseases by dynamical network biomarkers. Scientific Reports 2012, 2:342.

71. Schäfer J, Strimmer K: A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. Stat Appl Genet Mol Biol 2005, 4:Article32.

CHAPTER 2: LEVERAGING GENE CORRELATIONS IN SINGLE CELL TRANSCRIPTOMIC DATA

2.1 Abstract

BACKGROUND: Many approaches have been developed to overcome technical noise in single cell RNA-sequencing (scRNAseq). As researchers dig deeper into data—looking for rare cell types, subtleties of cell states, and details of gene regulatory networks—there is a growing need for algorithms with controllable accuracy and fewer *ad hoc* parameters and thresholds. Impeding this goal is the fact that an appropriate null distribution for scRNAseq cannot simply be extracted from data in which ground truth about biological variation is unknown (i.e., usually).

RESULTS: We approach this problem analytically, assuming that scRNAseq data reflect only cell heterogeneity (what we seek to characterize), transcriptional noise (temporal fluctuations randomly distributed across cells), and sampling error (i.e., Poisson noise). We analyze scRNAseq data without normalization—a step that skews distributions, particularly for sparse data—and calculate *p*-values associated with key statistics. We develop an improved method for selecting features for cell clustering and identifying gene-gene correlations, both positive and negative. Using simulated data, we show that this method, which we call BigSur (Basic Informatics and Gene Statistics from Unnormalized Reads), captures even weak yet significant correlation structures in scRNAseq data. Applying BigSur to data from a clonal human melanoma cell line, we identify thousands of correlations that, when clustered without supervision into gene communities, align with known cellular components and biological processes, and highlight potentially novel cell biological relationships.

CONCLUSIONS: New insights into functionally relevant gene regulatory networks can be obtained using a statistically grounded approach to the identification of gene-gene correlations.

Keywords: single cell RNA sequencing; gene-gene correlation; gene regulatory network; gene co-expression network; melanoma.

2.2 Background

Single cell RNA-sequencing (scRNAseq), along with the related method of single nucleus RNA-sequencing, now offer researchers unparalleled opportunities to interrogate cells as individuals. Methods have been developed to classify cell types; identify gene expression markers; infer lineages; learn gene regulatory relationships, and examine the effects of experimental manipulations on both levels of gene expression and cell type abundances [1-7]. Because scRNAseq data are noisy, reliable inference requires leveraging information across many cells, trading off sensitivity for statistical power. How to handle that tradeoff should depend, ideally, on one's goal. Unsupervised clustering of large numbers of transcriptionally very different cells ("cell types") into small numbers of groups of similar size allows for a great deal of latitude in aggregating information across cells; it is thus not surprising that many different clustering approaches perform well. Other tasks, such as ordering cells along a continuum of gene expression change, or picking out rare cell populations within much larger groups, are less forgiving, and a plethora of different approaches currently compete for investigators' attention [8-23]. Assessing the performance of such methods is frequently hindered by a lack of knowledge of ground truth.

A particularly challenging application of scRNAseq is the identification of patterns of gene co-expression. The identification of large-scale blocks of co-expressed genes—co-expression "modules"—can provide an alternative method for classifying cells when traditional clustering fails [24]. In contrast, smaller-sized blocks of gene co-expression have the potential to reflect true gene-regulatory networks that relate to specific functions [25]. This is because random

transcriptional noise in gene circuits should induce weak but real correlations among regulatory genes and their targets. Indeed, it has long been proposed that gene regulatory links could be discovered solely from the weak gene expression correlations that one might encounter when studying otherwise homogenous populations of cells [26-32].

Unfortunately, identifying small yet significant gene expression correlations in single cell data requires a degree of statistical power that scRNAseq applications rarely strive for (and, to be fair, rarely need to). Yet, as greater numbers of scRNAseq datasets accumulate, with a growing trend toward increasing numbers of cells per dataset, we wondered whether substantial amounts of novel information about gene co-regulation might be accessible simply through a more in-depth examination of pairwise gene expression correlations.

One of the main challenges in pursuing such a program is the absence of an accepted statistical model for pairwise correlations in scRNAseq data. Only with a model can one define a null hypothesis by which to judge whether observations are significant. Unfortunately, with scRNAseq, there is not good consensus regarding the model to use for the data distributions of individual genes, much less their correlations. The common approach of fitting individual gene data to *ad hoc* analytical distributions (e.g. “zero-inflated negative binomial” [33]; reviewed by [34]), has met with frequent criticism that is difficult to dismiss [35-38]. One may seek to circumvent such concerns by attempting to learn empirical distributions on a case-by-case basis from data, but this typically requires making assumptions about the amount and distribution of actual biological variation in the data, which are frequently unknown. Furthermore, pitfalls in implementing empirical methods can be hard to avoid, particularly with high-order statistical information, such as correlations. For example, the seemingly reasonable intuition that one might be able to construct the distribution of the correlation coefficient under the null hypothesis simply

by randomly permuting elements is actually incorrect [39]. So, as we will show below, is the intuition that one can improve the detection of gene correlations by averaging over groups of similar cells (i.e., creating "metacells") prior to calculation of correlation coefficients.

One of the major obstacles to defining an appropriate data distribution for scRNAseq data is the fact that underlying sources of technical variation are not fully understood, nor is the range of biological variation in biologically "equivalent" cells fully known. Here we begin by re-considering these factors, and leveraging the work of others, in pursuit of an analytical model of null correlation distributions that makes the fewest ad hoc assumptions and minimizes adjustable parameters. We show that the approach that emerges has the power to identify subtle yet real correlations, both positive and negative, in scRNAseq data, even among genes in modest numbers of cells that are relatively sparsely sequenced. Ultimately this method should be applicable not only to the identification of gene regulatory interactions, but to more complex tasks based on gene-gene correlations—such as the identification of cellular trajectories [40] and "tipping points" [41]—as well as providing a means to achieve a more principled approach to basic, early steps in scRNAseq analysis—such as normalization, batch correction, feature selection and clustering.

2.3 Results

2.3.1 The significance of gene-gene correlations

The statistical significance of correlations is rarely discussed because, for many common kinds of data—those that are continuous and at least approximately normal in distribution—the magnitude of correlation and its significance are related in a simple way that depends only on the number of measurements, and not the data distributions. Owing to Fisher [42], for any Pearson correlation coefficient (PCC), the *p*-value (probability of observing $|PCC| \geq x$ by chance) may be

estimated as $Erfc\left[\sqrt{(n-3)/2}\, arctanh(|x|)\right]$, where *n* is the number of samples, and *Erfc* is the complement of the Error function (we refer to this expression henceforth as the "Fisher formula").

Because scRNAseq data are both discrete and generally not normally distributed, *p*-values obtained using the Fisher formula cannot be accepted as accurate, but just how far off will they be? Figure 1 explores that question through simulation. Assuming Poisson-distributed data, and gene expression vectors of 500 cells in length, the formula does quite poorly for vectors of mean < 1, i.e., where the expected proportion of zeros exceeds 37%, assigning *p*-values that are too low for positive correlations, and too high for negative ones (Fig. 1A). For distributions with mean ≥1 (fewer than 37% zeros on average), the formula does reasonably well down to *p*-values as low as 10^{-4} but deviates progressively thereafter. This degree of accuracy would be a problem for any genome-wide analysis of correlations: To analyze pair-wise correlations among *m* genes one must test $m(m-1)/2$ hypotheses. With values of *m* often ≥ 12,000, this amounts to >7 x 10^8 simultaneous tests, such that statistical significance of any single observation could potentially require *p* as low as 10^{-9}, a value for which we may estimate, by extrapolation, that the Fisher formula is highly inaccurate even for genes with mean expression = 1.

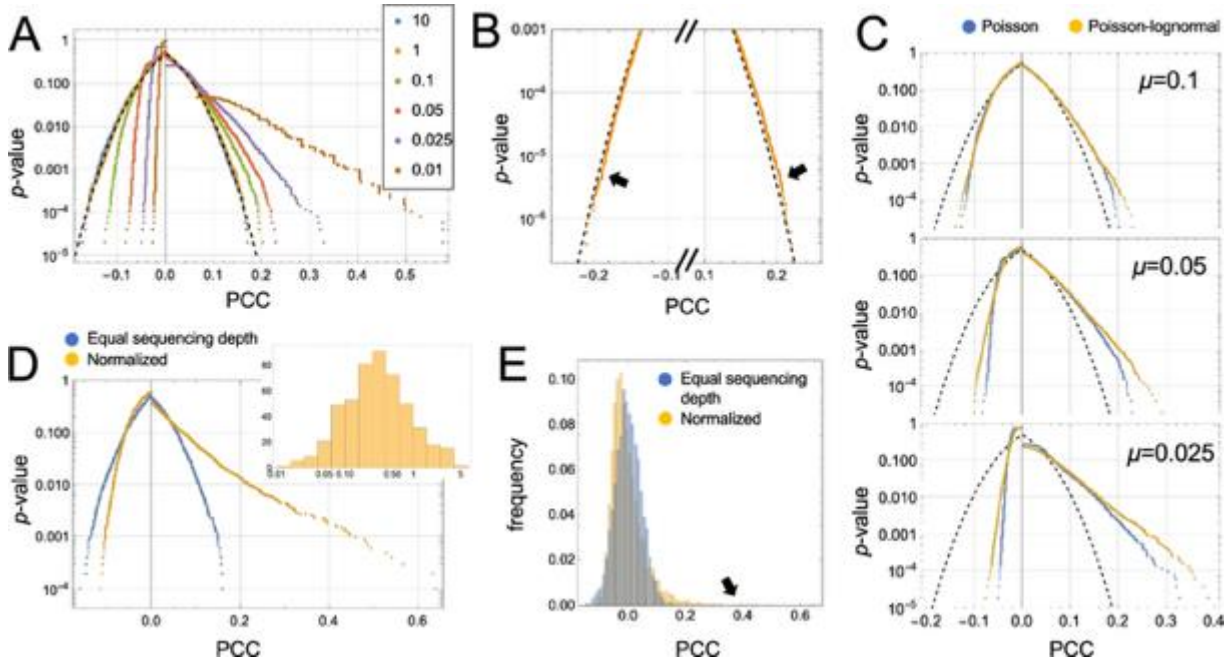


Figure 1. Relationship between Pearson correlation coefficient, vector sparsity and p-value.
The panels compare p-values determined empirically by correlating 50,000 pairs of random, independent vectors of length 500 with p-values predicted by the Fisher formula. A. Data were independent random variates from Poisson distributions with means as indicated. The dashed line shows the output of the Fisher formula. B. Even for vectors drawn from a distribution with mean = 1, the Fisher formula significantly mis-estimated p-values smaller than 10-4. C. Poor performance of the Fisher formula is worsened when data are drawn from a Poisson-log-normal distribution, rather than a Poisson distribution (in this case the underlying log-normal distribution had a coefficient of variation of 0.5). D-E. Data were simulated under the scenario that gene expression is the same in every cell, but due to differences in sequencing depth, observed gene expression varies according to the depths shown in the inset to panel D. In panel D, true gene expression was adjusted so that observed gene expression after normalization would have a mean of 1, and the Pearson correlation coefficients (PCCs) obtained by correlating randomly chosen vectors are shown. Panel D plots empirically derived p-values as a function of PCC, whereas panel E displays histograms of PCCs. Compared with data that do not require normalization, associated p-values from normalized data are even more removed from the predictions of the Fisher formula.

The simulations in Fig 1A-B assume Poisson-distributed data but, as is often pointed out, scRNAseq data are usually over-dispersed relative to the Poisson distribution (more on this below). As shown in Fig. 1C, adding in such additional variance causes simulated data to deviate even further from the Fisher formula.

Further problems arise when considering that scRNAseq data always come from collections of cells with widely varying total numbers of UMI. Depending on the platform, such

"sequencing depth" can vary over orders of magnitude, which is why normalization is usually considered a necessary early step in data analysis. Without normalization, it is obvious that many spurious gene-gene correlations would be detected, as any difference in sequencing depth between cells would, if not corrected for, induce positive correlation across all expressed genes.

As one might expect, normalizing individual reads by scaling them to each cell's sequencing depth eliminates this bias, restoring the expected value of PCC under the null hypothesis to zero. Yet normalization does not restore the *distribution* of PCCs to what it would have been had all cells been sequenced equally. The consequences can be dramatic, as shown in Fig. 1D-E, where we simulate a case in which "true" gene expression is the same in each of 500 cells, but observed gene expression is a Poisson random variate from a mean that was scaled by a factor chosen from a distribution of cell-specific sequencing depths similar to what one might observe in a typical scRNAseq experiment, using the 10X Chromium platform (inset, Fig. 1D). Despite mean gene expression being ~1, the relationship between PCC and *p*-value more closely resembles the case (in the absence of sequencing depth variation) where the mean is 0.01 (Fig. 1A). This is surprising, given that the average fraction of zeros in the normalized vectors was only 0.68 (which, for Poisson-distributed data, would be expected to occur at a mean expression level of 0.39). In short, for gene expression data resembling what is typically obtained in scRNAseq, the Fisher formula is highly unsuitable, for most pairs of genes, for estimating the significance of correlations.

2.3.2 An analytical model for the distribution of correlations

The data in Figure 1 indicate that the relationship between PCCs and *p*-values is highly sensitive both to data structure and procedures intended to "correct" for technical variation. Because of this, we were concerned that a suitable null model for the distribution of correlations

might be difficult to estimate empirically, especially when the true biological variation in most datasets is unknown. We therefore turned to constructing a null model analytically, attempting to account for known sources of variation (beside meaningful biological variation). The three sources considered were (1) variation introduced by imperfect normalization; (2) technical variation due to random sampling of transcripts during library preparation and sequencing; and (3) variation due to stochasticity of gene expression. The last of these cannot properly be called technical variation—fluctuating gene expression is a biological phenomenon—but like technical noise, gene expression fluctuation is usually an unwanted source of variation, and its effects need somehow to be suppressed.

With regard to the first source, we follow [43] in correcting not the gene expression data points themselves, but rather their Pearson residuals. Traditionally, the Pearson residual P_{ij} for cell *i* and gene *j*, is defined as

$$P_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\mu_j}}$$

where x_{ij} is the gene expression value for cell *i* and gene *j*, and μ_j is the mean expression of gene *j* averaged over all cells. As the Pearson residual is mean-centered, its expectation value, $E[P_{ij}]$, is zero; thus, the average of Pearson residuals for a large number of x_{ij} drawn from a single distribution should approach zero. The average of squares of Pearson residuals can be seen, by inspection, to approach the variance divided by the mean of the distribution from which the x_{ij} derive. Variance divided by mean is also known as the Fano factor and is often used to assess whether data are consistent with a Poisson distribution since, for any Poisson distribution regardless of mean, $E[{P_{ij}}^2] = 1$.

Pearson residuals may also be used to construct PCCs, which are commonly defined in terms of variances and covariances, but may be equivalently expressed as:

$$\mathrm{PCC}_{a\times b} = \frac{\sum_{i=1}^{n} P_{ia}P_{ib}}{\sqrt{\sum_{i=1}^{n} {P_{ia}}^2 \sum_{i=1}^{n} {P_{ib}}^2}} = \frac{1}{(n-1)\sqrt{\phi_a \phi_b}} \sum_{i=1}^{n} P_{ia}P_{ib}$$

where $\mathrm{PCC}_{a\times b}$ is the Pearson correlation coefficient between genes *a* and *b*, *n* the number of cells, and $\phi_a$and ϕ_b represent the Fano factors for genes *a* and *b*, respectively, i.e.

$$\phi_j = \frac{1}{n-1} \sum_{i=1}^{n} {P_{ij}}^2$$

Since $E[P_{ij}] = 0$ for any gene, it follows that $E[\mathrm{PCC}_{a\times b}] = 0$, as long as all the expression values for gene *a* are drawn from a single distribution, and those for *b* are independently drawn from a single distribution.

However, in scRNAseq, unequal sequencing depth means that the expression values for any given gene are generally *not* drawn from a common distribution, but rather from one that is different for each cell. Interestingly, dividing x_{ij} in a Pearson residual by an appropriate scaling constant—i.e. normalizing the data—will restore $E[P_{ij}]$ to 0, but will not restore the higher moments of P_{ij}, e.g., $E[{P_{ij}}^2] \neq 1$. To capture the correct second moment, one must scale the value of μ_j inside each Pearson residual, rather than scaling x_{ij}. As [43] have pointed out, we can define a separate μ_{ij} for each cell and gene:

$$\mu_{\mathrm{ij}} = \frac{\sum_j x_{\mathrm{ij}} \sum_i x_{\mathrm{ij}}}{\sum_{ij} x_{\mathrm{ij}}}$$

and consequently, define a corrected Pearson residual as

$$P_{ij} = \frac{x_{\mathrm{ij}} - \mu_{\mathrm{ij}}}{\sqrt{\mu_{\mathrm{ij}}}}$$

Although this transformation does not recover moments of the Pearson residual beyond the first two, it provides a principled alternative to traditional normalization. Moreover, by permitting calculation of a corrected Fano factor that has the appropriate expectation value under the assumption of Poisson distributed data, it can be used to test that assumption, in real data.

Source #2 refers to technical variation due to random, independent sampling of discrete numbers of transcripts. Although the sampling process in scRNAseq involves several discrete steps, including cell lysis, library preparation and DNA sequencing, several groups have argued that, at least at modest to low expression levels, simple Poisson "noise" can reasonably model the variation derived from these processes [35-38]. We accept this assumption here, but note that, in the following derivations, the Poisson distribution could just as easily be replaced by another known distribution, if it were adequately justified.

Finally, source #3 refers to the fact that "equivalent" cells are usually only equivalent in a time-averaged sense, i.e., transcript numbers will fluctuate around some mean value. Both theory and observation support the conclusion that these fluctuations can be large [30, 44-46]. The actual magnitude seems to differ for different categories of genes, but data from single-molecule transcript counting [44] suggest that, for most genes, the distribution of transcripts typically is approximately log-normal (consistent with the theoretical work of [47]), with a coefficient of variation in the range of 0.2-0.6.

Thus, even if library synthesis and sequencing performed identically across cells, one should not expect to observe Poisson-distributed reads. Under the reasonable assumption that gene expression fluctuations and sampling are independent, the variance of the combined process should

be the sum of the variances of the composing processes. Since both processes have the same mean, we can re-state this thusly: the Fano factor of the combined process should be the sum of the Fano factors of the composing processes. One can then use this fact to further adjust the Pearson residual, and subsequently the Fano factor. Specifically, we define a "modified corrected Pearson residual" as:

$$P_{\mathrm{ij}}{}' = \frac{x_{ij}-\mu_{ij}}{\sqrt{\mu_{ij}(1+c_j{}^2\mu_{ij})}} \qquad \text{(Eq. 1)}$$

where c represents the coefficient of variation of gene expression for gene j. Accordingly, the expectation value for $(P_{\mathrm{ij}}{}')^2$ becomes

$$\frac{< \frac{(x_{ij}-\mu_{ij})^2}{\mu_{\mathrm{ij}}} >}{(1+c_j{}^2\mu_{\mathrm{ij}})}$$

which resembles a Fano factor divided by $(1+c_j{}^2\mu_{\mathrm{ij}})$. Since $c^2\mu_{\mathrm{ij}}$ can alternatively be written as $\frac{\sigma_{\mathrm{ij}}}{\mu_{\mathrm{ij}}}$, where s is the standard deviation of gene expression fluctuations, the term $(1+c_j{}^2\mu_{\mathrm{ij}})$ is simply the sum of the Fano factors for Poisson sampling (unity) and gene expression fluctuation $(c_j{}^2\mu_{\mathrm{ij}})$. Dividing by that sum essentially removes the additional variance due to gene expression noise from the expectation value of $(P_{\mathrm{ij}}{}')^2$, restoring that value to 1.

In this way, one may define a "modified corrected Fano factor" ϕ' equal to the expectation value of $(P_{\mathrm{ij}}{}')^2$. Likewise, we may use ϕ' to define a modified corrected Pearson correlation coefficient, PCC′:

$$\mathrm{PCC}'_{a\times b} = \frac{1}{(n-1)\sqrt{\phi'_a\phi'_b}}\sum_{i=1}^{n} P'_{ia}P'_{ib} \qquad \text{(Eq. 2)}$$

Notice that ϕ' provides a measure of the degree to which a gene's expression is more variable than expected by chance, and PCC′ provides a metric by which gene-pairs are judged more positively or negatively correlated than expected by chance, correcting in both cases for unequal sequencing across cells (without normalizing data) and the expected noisiness of gene expression.

To use these statistics in practice, one needs to know not only their expectation values but their full distributions under the null hypothesis. Constructing those analytically requires not only the coefficient of variation of gene expression noise, c, but the full distribution of that noise, which in the absence of information to the contrary, we will take to be log-normal [47], noting that any other distribution could as easily be substituted in the following discussion.

We thus treat x_{ij} as a Poisson random variable from a distribution whose mean is a log-normal random variable with a coefficient of variation of c (we refer to this compound distribution as Poisson-log-normal). Although an analytical form for the probability mass function of the Poisson-log-normal distribution is not known, we may derive analytical forms for an arbitrary number of its moments, as a function of μ (the mean) and c (see Methods). Thus, one can calculate for every cell i and gene j, given the observed values of μ_{ij}, the moments of the expected distributions of the ${P_{\mathrm{ij}}}'$ under the null hypothesis, and subsequently those for the $({P_{\mathrm{ij}}}')^2$. From there one can calculate the moments of any number of products and sums of ${P_{\mathrm{ij}}}'$ and $({P_{\mathrm{ij}}}')^2$, such that, eventually, the moments of ϕ' and PCC′ under the null hypothesis are ultimately obtained (see Appendix). Given a finite number of moments, one can estimate the tails of the distributions of these statistics (see Methods), allowing one to calculate the probability of extreme values of ϕ' and PCC′ arising by chance (p-values).

This method, which we refer to as BigSur (Basic Informatics and Gene Statistics from Unnormalized Reads), provides an approach for discovering genes that are significantly variable across cells (based on ϕ'), and gene pairs that are significantly positively or negatively correlated (based on PCC′), automatically accounting for the widely varying distributions of these statistics as a function of gene expression level and vector length (number of cells). The one free parameter in the method, *c*, is relatively constrained, as its average value (over all genes) can be estimated from a plot of ϕ' versus mean expression (see below). In this manner, one can avoid the use of arbitrary thresholds or cutoffs in deciding which genes are significantly "highly" variable (e.g., for dimensionality reduction and cell classification) and which genes are significantly positively and negatively correlated (e.g., to discover gene expression modules and construct regulatory networks).

2.3.3 Performance on simulated data

In Figure 2 we simulate gene expression data for 1000 genes and 999 cells, under the null model described above (i.e., complete independence), varying "true" mean expression widely and uniformly over the genes, such that the most highly expressed genes average 3467 transcripts/cell and the most lowly expressed 0.0351 per cell. "Observed" gene expression values are then obtained by randomly sampling from a Poisson log-normal distribution with c=0.5, in which the gene-specific mean is first scaled in each cell according to a pre-defined distribution of sequencing depth factors (chosen to mimic typical ranges of sequencing depth when using the 10X Chromium platform). The result is a set of gene expression vectors of length 999, with means varying between 0.001 and 231.

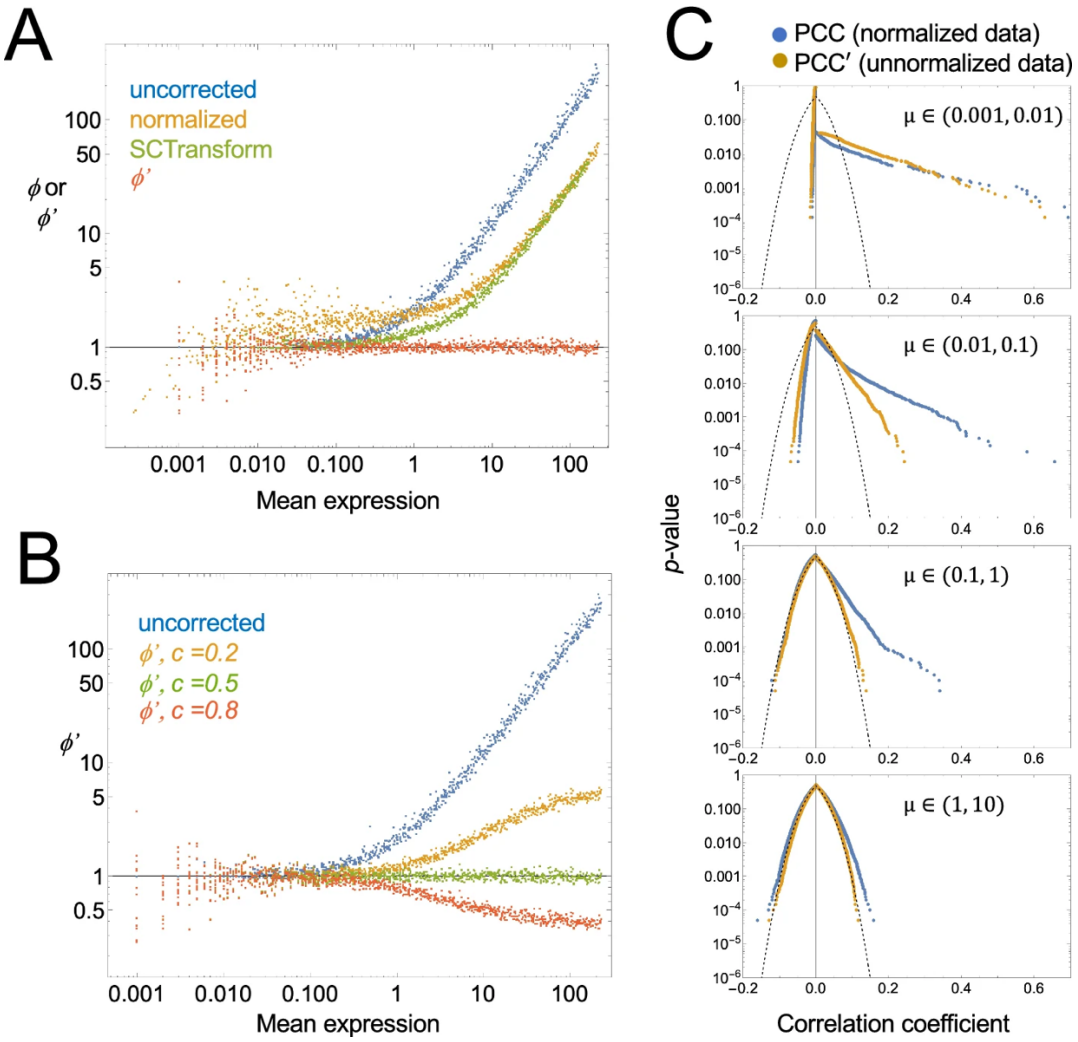


Figure 2. Comparing uncorrected and modified corrected Fano factors and correlation coefficients. Random, independent, uncorrelated gene expression data was generated for 1000 genes in 999 cells, under the assumption that observations are random Poisson variates from a per-cell expression level that is itself a random variate of a log-normal distribution, scaled by a sequencing depth factor that is different for each cell (see methods). A. Uncorrected (ϕ) or modified corrected (ϕ') Fano factors are plotted as a function of mean expression level for each gene. Uncorrected factors were calculated either without normalization, or with default normalization (scaling observations by sequencing depth factors, learned by summing the gene expression in each cell). Uncorrected Fano factors were also calculated using SCTransform [48] as an alternative to default normalization. Modified corrected Fano factors were obtained by applying BigSur to unnormalized data, using a coefficient of variation parameter of c=0.5. B. Modified corrected Fano factors (ϕ') were calculated as in A, but using different values of c. The data suggest that an optimal choice of c can usually be found by examining a plot of ϕ' versus mean expression. C. Empirical p-values associated with uncorrected (PCC) or modified corrected (PCC') Pearson correlation coefficients were calculated for pairwise combinations of genes in bins of different mean gene expression level (μ); examples are shown for four representative bins (both genes derived from the same bin). With increasing gene expression levels, the p-value vs. PCC relationship begins to approach the Fisher formula (dashed curve), but it does so much sooner for PCC' than PCC.

As shown in Figure 2A, for most genes with mean expression greater than ~0.1 reads/cell, uncorrected Fano factors exceed 1, and rise linearly with expression level. Normalizing the data—scaling expression in each cell to the relative sequencing depth of that cell—reduces high-expression skewing somewhat, but also elevates the Fano factors for genes with low expression, driving them closer to 2. These results, in which the majority of genes display Fano factors greater than 1, which rise further for highly expressed genes, agree with the pattern most commonly seen in actual scRNAseq data. Values of the Fano factor for lowly expressed genes may be restored to near 1 by normalizing using SCTransform, an algorithm designed to correct for some of the variance-inflating aspects of normalization-by-scaling [48], but the presence of high Fano factors among the highly expressed genes persists.

In contrast, if we calculate the modified corrected Fano factor, ϕ', for each gene, using c=0.5, we see that values are now centered around 1 at all expression levels (Fig. 2A). Note that choosing different values of c produces consistent positive or negative skewing at mean gene expression values above 1 (Fig. 2B). Under the assumption that most genes in real data should not vary significantly across cells, one may therefore estimate the optimal choice of c for any data set by simply finding the value that minimizes such high-expression skewing of ϕ'.

Because it uses the moments of the Pearson residuals to calculate p-values, BigSur assigns statistical significance to every gene's ϕ'. As expected, given that the data in Fig. 2A were random samples, no values of ϕ' were found to be statistically significant at a Bonferroni-corrected p-value threshold of 0.05, or a Benjamini-Hochberg [49] false discovery rate of 0.05. Indeed, the lowest uncorrected p-value associated with any of the 1000 genes in Fig. 2A was 0.001.

Similarly, when the same synthetic data are analyzed for gene-gene correlations, one may compare the PCC values produced directly from normalized expression data with the PCC′ values

produced (from unnormalized data) by BigSur. As expected, both procedures return a distribution of values with zero mean, but PCC values are more broadly distributed than PCC′. Fig. 2C shows the frequency at which various values of the correlation coefficient arise when vectors with different mean gene expression are correlated with each other. Although significant skewing from the Fisher formula is apparent, especially at low values of gene expression, it is much greater for PCC than PCC′. Indeed, for moderately expressed genes (e.g., 1-10 unique molecular identifiers [UMI] per cell), only PCC′ returns values whose distribution is relatively insensitive to expression level.

2.3.4 Performance on real data

To characterize the performance of BigSur on real data, we used the droplet-based sequencing data of Torre et al., 2018 [50], obtained from a clonal isolate of a human melanoma cell line grown in culture. In this data set, the number of cells is large (8,640), data were validated on a second sequencing platform as well as by single molecule FISH, and the broad distribution of sequencing depths was typical of droplet-based sequencing.

We deliberately chose a clonal cell line because tissues always contain multiple cell types, i.e., groups of cells that express large sets of genes in cell-type specific ways. In such heterogeneous samples, genes that are associated with cell type identity will necessarily be strongly and densely correlated with each other; making the identification of correlations, in a sense, too easy—i.e., not a particularly good test of a method's performance—and not particularly informative (one may expect to identify as correlated more or less the same genes one would find by clustering cells by gene expression and testing for differential expression between clusters).

In contrast, the use of (ostensibly) homogeneous cells forces BigSur to operate on more subtle connections—for example, those involving fluctuations in function-specific gene regulatory networks—that cell clustering and differential expression would not easily detect.

Accordingly, scRNAseq data from these cells were subjected to minimal pre-processing prior to analysis (see Methods), such that expression values were analyzed for 14,933 genes. Total numbers of UMI per cell varied greatly, ranging from 67 to 90,494, with 90% of cells containing between 666 and 9004 UMI.

First, we compared (uncorrected) PCCs for all gene-gene pairs, calculated using default-normalized expression data (i.e., data scaled to total UMI per cell), with PCC′ values returned by BigSur (Figure 3A-B). In panel B, frequencies are scaled logarithmically, to better display the distribution of large absolute values. BigSur associated a false discovery rate (FDR) of 2% to *p*-values less than 1.15×10^{-4}, at which threshold it detected 639,789 correlations, 350,466 of which were positive. For uncorrected PCCs, the same *p*-value cutoff would translate, using the Fisher formula, to $|PCC| > 0.041$, which is displayed as a dotted line in Fig. 3B. Using this threshold, 1,484,156 correlations would be considered significant. Comparison of histogram shapes shows that use of uncorrected PCCs particularly inflates positive correlations, especially large ones, and under-counts negative ones, which is consistent with the results obtained using simulated data (Fig. 2C).

To see how the discovery of correlations varied with gene expression level, we divided genes into 6 bins of different mean expression, and separately analyzed correlations between genes in all 21 possible combinations of bins. The full data are presented in Fig. S1-S2 (additional files 3-4), with two representative panels in Fig. 3C-D. Each point represents a gene-gene pair. The value on the abscissa is either the uncorrected PCC (Fig. 3C and S1) or modified corrected PCC′

(Fig. 3D and S2), and the value on the ordinate is the -log_{10} *p*-value, as determined by BigSur (i.e., the larger the number, the lower the *p*-value). Shaded territories mark those data points that were judged statistically significant (FDR<0.02) either by BigSur (orange and gray), or when *p*-values were calculated by the Fisher formula (blue and orange; for further details see figure legend). The data confirm that the Fisher formula returns an excess of correlations, compared to BigSur, albeit less severely for PCC′ than for PCC. Examination of the full dataset (Fig. S1-S2) shows that many more truly significant correlations are found among highly expressed genes; and the Fisher formula performs worst when at least one of the pairs in a correlation is a lowly-expressed gene. Indeed, as gene expression becomes high (e.g., mean value >1 UMI per cell for both genes in a pair), the distribution of *p*-values calculated by BigSur for PCC′ begins to approximate the Fisher formula reasonably well (Fig. S2), with deviation apparent only for very low *p*-values (-$log_{10}p$ > 10). This observation validates the accuracy of the method used by BigSur to recover *p*-value distributions from the first five moments of the expected distributions of modified, corrected Pearson residuals (see Methods).

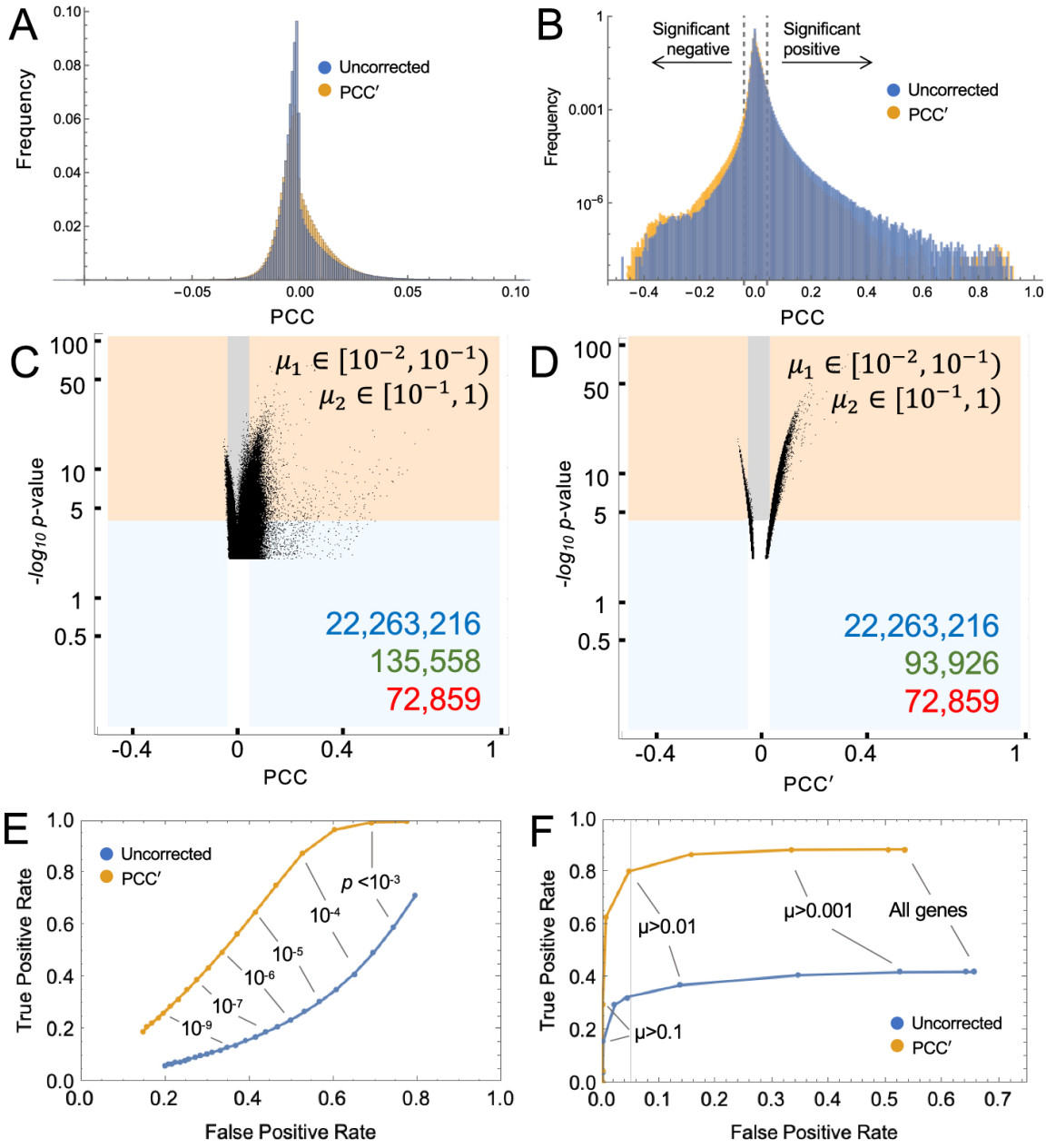


Figure 3. Statistical significance of pairwise gene correlations in data from a clonal cell line.
A-B. Using scRNAseq data from a human melanoma cell line [50] (8640 cells x 14933 genes), pairwise values of PCC were calculated from normalized data, and PCC' from raw data. Histograms display the frequency of observed values (the logarithmic axis in B emphasizes low-frequency events). Notice in B how positive skewing, also seen in simulated data (Fig. 2), is less for PCC' than PCC. Dashed lines in B show thresholds at which Fisher formula-derived p-values would fall below 1.1 x 10-4. C-D. Scatterplots showing p-values assigned by BigSur to pairs of genes within two representative sets of bins of gene expression (for all pairwise combinations see Fig. S1-S2/additional files 3-4). The abscissa shows PCC (panel C) and PCC' (panel D). The ordinate gives the negative log10 of p-values determined by BigSur, i.e., larger values mean greater statistical significance. Orange and gray shading indicate gene pairs judged significant by BigSur (FDR<0.02). Blue and orange show gene pairs that would have been judged statistically significant by applying the Fisher formula to the PCC or PCC', using the same p-value

threshold as used by BigSur. The blue region contains gene pairs judged significant by the Fisher formula only, while the unshaded region shows gene pairs not significant by either method. Numbers in the lower right corner are the total numbers of possible correlations (blue), statistically significant correlations according to the Fisher formula (green), and statistically significant correlations according to BigSur (red). E-F. ROC curves assessing whether the overall performance of the Fisher formula—applied either to PCC or PCC'—can be adequately improved either by using a more stringent p-value cutoff (E) or limiting pairwise gene-correlations to those involving only genes with mean expression above a threshold level, μ (F).

Assuming, for the sake of illustration, that the correlations judged significant by BigSur represent ground truth, we may then calculate levels of true- and false-positivity when *p*-values are calculated by feeding either PCC or PCC′ into the Fisher formula. This enabled us to ask whether applying a more stringent *p*-value cutoff, or thresholding gene expression (i.e., excluding genes below a certain expression level), might enable this simpler, formulaic approach to achieve an acceptable level of sensitivity and specificity. As shown by the receiver-operating characteristic (ROC) curves in Fig. 3E, the performance of uncorrected PCCs is exceedingly poor regardless of *p*-value threshold, with false positives exceeding true positives at all values. PCC′ does better, but to control FDR to <10%, one still loses the ability to detect >85% of true positives.

Arbitrarily thresholding gene expression performs somewhat better (Fig. 3F). For uncorrected PCCs, one must exclude all genes with average expression < 0.065 UMI/cell to control the FDR at 5%, which for this dataset means discarding 67% of all gene expression data, and in return recovering only 33% of true positives. PCC′ does much better here: we may recover 81% of true positives at an FDR of 5% by discarding the ~52% of genes with the lowest expression. While the exact numbers are likely to vary for different data sets, these observations suggest that, if one is willing to sacrifice the power to identify a substantial minority of correlations, feeding PCC′ (but not PCC) into the Fisher formula can represent an acceptable and computationally fast alternative to *p*-value identification by BigSur.

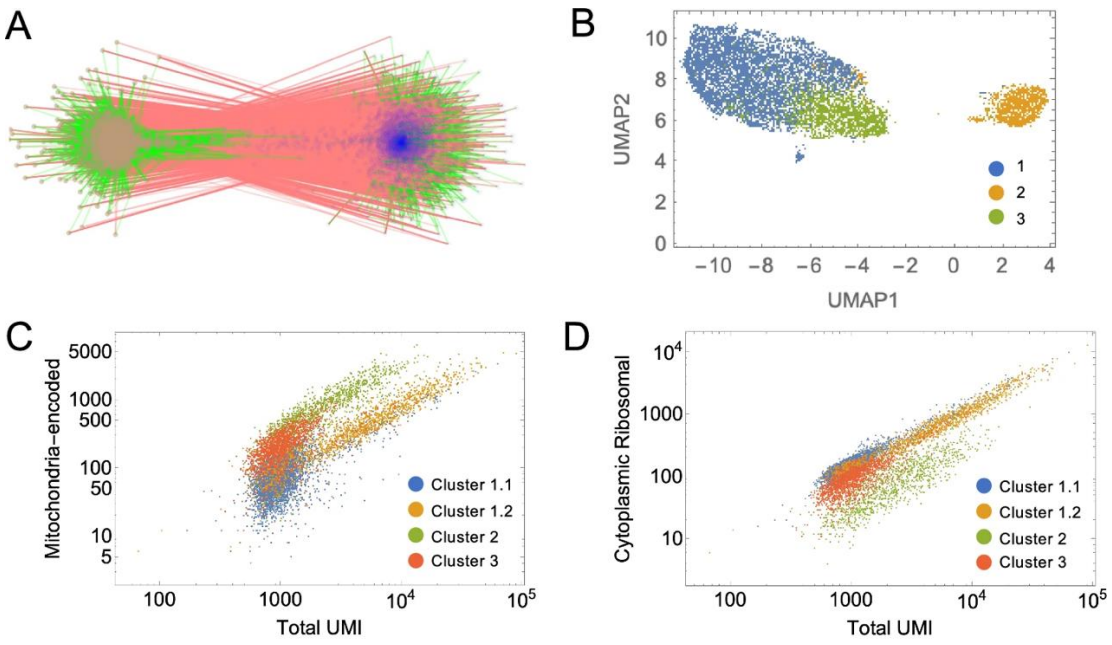
2.3.5 Clustering using correlations

Although BigSur found 639,789 statistically significant correlations in this dataset (about 0.57% of all possible pairwise correlations) the vast majority had quite small values of PCC′ (Fig. 3A-B), indicating that most statistically significant correlations were weak. To obtain a measure of correlation strength that could be compared across samples of different lengths (numbers of cells), we expressed each correlation in terms of an "equivalent" PCC, which is simply the PCC value that, for continuous, normally-distributed data of the same length, would have produced the same *p*-value (by the Fisher formula). As shown in Fig. S3 (additional file 5), only 4335 gene pairs displayed equivalent PCCs greater than 0.2 or less than -0.2.

Yet, despite the weak strength of most correlations, there are good reasons to believe them to be biologically relevant. One of the simplest comes from examining the frequency at which correlations arise among paralogous genes and genes that encode proteins that physically interact. It is known that gene paralogs are often co-regulated [51] leading us to expect paralog pairs to be enriched among *bona fide* correlations. It is also reasonable to expect that transcripts encoding proteins that interact will be co-expressed at least some of the time. As it happens, among the correlations identified by BigSur we observe ~12 fold enrichment for paralogs and ~7.5 fold enrichment for genes encoding physically-interacting proteins [see Methods].

To divide the full set of correlations into potentially interpretable groups, we used a random-walk algorithm [52] to identify subnetworks more highly connected internally than to other genes; we refer to these as gene communities. Most communities were of modest size, with 94 of the 96 largest containing between 4 and 392 genes each. However, the largest two contained 2160 and 1570 genes respectively, were very densely connected internally, and strongly anti-correlated with each other (Figure 4A). These factors strongly suggest that these cells, despite being a clonal line, are heterogenous, falling into (at least) two distinct groups. Interestingly, the

largest gene community contained virtually the entire set of mitochondrially-encoded mitochondrial genes, and the second largest contained virtually all protein-coding ribosomal genes (for both cytoplasmic and mitochondrial ribosomes). Using the top 50 most-highly connected genes (those with the largest positive equivalent PCCs) in each of these communities as features, we performed Leiden clustering on the modified corrected Pearson residuals of all 8,640 cells, and easily subdivided them into three clusters of 5,812, 1,180 and 1,638 cells, which we labeled as clusters 1,2 and 3, respectively (Fig. 4B; see Methods).



E

Cluster	Cell number	Significant intra-cluster correlations Positive	Negative	Total	Genes expressed	Genes involved in Significant correlations
all cells	8640	350,466	289,323	639,789	14,933	11,961
1	5,812	21,203	8,589	29,792	14,933	4,405
1.1	4,226	1,382	536	1,918	14,786	1,281
1.2	1,582	18,353	4,702	23,055	14,915	2,519
2	1,180	20,390	4,992	25,382	14,698	2,672
3	1,638	1,869	366	2,235	14,232	918

Figure 4. Clustering cells based on mitochondrial and ribosomal communities.
A. Correlations among the two largest gene communities detected by BigSur are shown. Vertices are genes, and edges—green and red—represent significant positive and negative correlations, respectively.

Blue vertices represent members of the mitochondrially encoded gene community and brown vertices the ribosomal protein gene community (Labels have been omitted due to the large numbers of gene involved). B. Using the top 50 most highly positively connected vertices in the two communities as features, cells were subjected to PCA and Leiden clustering; the three clusters recovered are displayed by UMAP. C-D. After a second round of clustering of Cluster 1, the resulting four cell groups were analyzed for the distribution of expression of ribosomal protein-coding and mitochondrially-encoded genes, as a function of total UMI in each cell. The results show that the four clusters form distinct groups based on their relative abundances of ribosomal and mitochondrial genes. E. Results of applying BigSur to each cluster. Note the large decrease in statistically significant correlations—especially negative correlations—in any of the clusters when compared with results obtained using all of the cells together (>600,000 total correlations). This is consistent with heterogeneity in the original sample, causing large blocks of genes to correlate and anti-correlate.

We then analyzed each cluster independently by BigSur, re-calculating PCC′ values and statistical significance for each group of cells. Surprisingly, in each cluster BigSur again found two large, strongly anti-correlating communities, one of which contained the mitochondrial-encoded genes and the other the ribosomal protein genes. This led us to further subcluster cluster 1, again using the 50 most-highly connected genes in each of these two communities as features and subdivided it into two groups of 4,226 (cluster 1.1) and 1,582 cells (cluster 1.2).

Variable expression of mitochondrially-encoded genes is a common finding in scRNAseq. Their presence at high levels (e.g. >25% of total UMI) is usually interpreted as an indication of a "low quality" cell—potentially one in which the plasma membrane has ruptured and cytoplasm has been lost—or perhaps a cell in the process of apoptosis [53]. Closer examination of the cell clusters identified in this dataset suggests these phenomena are likely only part of the explanation. In Fig 4C, we plot mitochondrial-encoded UMI versus total UMI for the entire set of cells, coloring them according to the four cell clusters mentioned above. Several distinct behaviors were noted. Cell cluster 2 forms a coherent group with high mitochondrial expression that is linearly proportional to total UMI. Throughout this group, the percentage of total transcripts that is mitochondrial remains in a narrow band, with mean of 34% and coefficient of variation (CV) of 0.41. Cluster 3 displays low total UMI, and a mitochondrial fraction centered around a mean of

21% (CV=0.43). Cluster 1.2 has high mitochondrial expression, but even higher total UMI, such that the mitochondrial fraction averages 9.7% (CV=0.45), while cluster 1.1 has both low total UMI and low mitochondrial UMI (average mitochondrial fraction 8.2%, CV=0.47).

In Fig. 4D, we also plot total UMI derived from ribosomal protein genes against total UMI. Of all the clusters, cluster 2 best fits the expectation for "damaged" cells: Mitochondrial and ribosomal UMI rise proportionately with total UMI (consistent with randomly varying sequencing depths), but the proportion of mitochondrial UMI is almost three times higher, the proportion of ribosomal UMI almost three times lower, and total UMI about 2.5 times lower than in cluster 1.2, the only other cluster that displays a wide range of sequencing depths. Yet it is curious that the mitochondrial proportions in cluster 2 are so narrowly distributed around a mean; one might expect variable degrees of cytoplasm leakage following cell damage to produce a broad distribution beginning near cluster 1.2 and gradually tapering off. The absence of such behavior suggests that such cells are not merely variably damaged during preparation, but represent a distinct, possibly pre-existing state, perhaps associated with some form of cell stress or death (although we see no significant enrichment of gene expression associated with apoptosis [54] in cluster 2).

Even if we remove cluster 2 from further analysis, clusters 1.1, 1.2 and 3 also differ between each other in relative proportions of mitochondrially-encoded, ribosomal and total transcripts; however, the way in which they do so is not suggestive of any simple mechanism of cell damage (and overall mitochondrial content is not in the range that would necessarily lead to exclusion from downstream scRNAseq analyses). As mentioned above, using the genes in the mitochondrially-encoded- and ribosomal-enriched communities as features, analysis of gene correlations using BigSur separately on each of the four clusters revealed anti-correlating communities of mitochondrially-encoded and ribosomal genes in every case (Figure 5A-D; Table

3). In fact, even after another round of subclustering of the subclusters derived from cluster 1 (cluster 1.1 and cluster 1.2) using these genes as features, BigSur still identified distinct, anti-correlating mitochondrially-encoded and ribosomal communities (not shown). These data strongly suggest that, notwithstanding effects of cell damage or cell death, there exists among healthy cells continuous, anti-correlated variation in both the mitochondrially-encoded and ribosomal protein genes, suggestive of some biologically meaningful regulatory relationship (discussed further below).

Community	# of genes	Category	Gene names
A	253	Cell cycle, G2/M	ANLN, ANP32B, ANP32E, ARHGAP11A, ARHGEF39, ARL6IP1, ASPM, AURKA, AURKB, BIRC5, BUB1, BUB1B, BUB3, CALM3, CASC5, CBX1, CCNA2, CCNB1, CCNB2, CCNF, CDC20, CDC25B, CDC27, CDCA2, CDCA3, CDCA8, CDK5RAP2, CDKN1B, CDKN3, CENPA, CENPE, CENPF, CEP55, CEP70, CHEK2, CIT, CKAP2, CKAP2L, CKAP5, CKS1B, CKS2, CNTRL, CTCF, DBF4, DDX39A, DEPDC1, DEPDC1B, DKC1, DLGAP5, DNAJA1, DNAJB1, DTYMK, DYNLL1, ECT2, EGR1, FAM64A, FAM83D, FANCD2, FOXM1, G2E3, GAS2L3, GPSM2, GTSE1, H2AFV, H2AFX, H2AFZ, HJURP, HMG20B, HMGB2, HMGB3, HMGN2, HMMR, HN1, HNRNPD, HP1BP3, HSP90AA1, HSPA8, ILF2, INCENP, KIAA1524, KIF11, KIF14, KIF15, KIF18A, KIF20A, KIF20B, KIF23, KIF2C, KIF4A, KIF5B, KNSTRN, KPNA2, KPNB1, LBR, LIN54, LSM6, MAD2L1,

			MIS18BP1, MKI67, MPHOSPH9, MTF2, MZT1, NDC80, NEIL3, NEK2, NUCKS1, NUF2, NUP37, NUSAP1, OIP5, PBK, PCF11, PDS5B, PIF1, PLK1, PLK4, PPP1R12A, PPP2R5C, PRC1, PRR11, PSMC3, PSRC1, PTTG1, RACGAP1, RAD21, RANGAP1, RBM8A, RCCD1, RPS6KA5, SFPQ, SGOL1, SGOL2, SHCBP1, SMC4, SNRPA1, SNRPD1, SNRPD3, SPAG5, SPDL1, SRSF3, STMN1, TACC3, TICRR, TOP2A, TPX2, TRIP13, TROAP, TTK, TUBA1B, TUBA1C, TUBB2A, TUBB4B, UBE2C, UBE2S, UTP18, WHSC1, XPO1
		Cell cycle, other	ACTB, ACTL6A, BRD7, CACYBP, CENPN, CENPW, CHAMP1, DYNLT1, FXR1, H3F3B, HES1, KIAA0586, LMNB2, LYAR, MAGOHB, NCAPD2, NCAPG, PARPBP, PIM3, POLR2K, PPP4R2, RAN, RHEB, SETD2, SKA2, SNRPB, UBE2D3, YY1, ZWILCH
		Spliceosome	ACIN1, LSM4, LSM5, PPIH, RBMX, SRRM1, SRRT, SRSF7, WBP4

		Not accounted for (58/253) = 23%	*ABHD2, ACAT2, ACTG1, AFF4, AHSA1, ARPC5L, AZIN1, BNIP2, BRIX1, C1orf52, C5orf34, CCDC150, CCDC18, CCDC34, CHORDC1, DDX6, DIAPH3, DLEU2, EIF3J, EIF5, EXOSC3, FAM46A, FUBP1, FZD6, GNL2, HIBCH, HIRIP3, HMGB1, HMGN1, HSPH1, IDI1, IER2, IFI16, KPNA4, LEO1, LZIC, MITF, NAV2, PHAX, PHF19, PPID, PRDX3, PSPC1, PTGES3, RBBP6, RLF, RPL39L, SCLT1, SEC62, SUB1, SYTL2, TAF13, TFAM, TRMT10B, TWISTNB, UBALD2, UGDH, WDR1*
B	213	Cell cycle G1/S, DNA replication	ARL6IP6, ASF1B, ATAD2, ATAD5, BAZ1B, BLM, BRCA1, BRCA2, BRIP1, CASP8AP2, CCNE2, CDC16, CDC45, CDC6, CDCA5, CDCA7, CDK1, CENPQ, CHAF1A, CHAF1B, CHEK1, CLSPN, DDX11, DEK, DHFR, DNA2, DNMT1, DSCC1, DTL, DUT, E2F7, E2F8, EME1, ESCO2, EXO1, EZH2, FAM111A, FBXO5, FEN1, GEN1, GINS1, GINS2, GINS4, GMNN, HELLS, HIST1H4C, ICMT, KIAA0101, LIG1, MASTL, MCM10, MCM2,

			MCM3, MCM4, MCM5, MCM6, MCM7, MCM8, MGME1, MSH2, MSH6, NASP, ORC1, ORC6, PCNA, PKMYT1, POLA1, POLD3, POLE3, POLQ, PRIM1, PRKDC, RAD18, RAD51, RAD51C, RBBP4, RBBP7, RBBP8, RBL1, RECQL, RFC1, RFC2, RFC3, RFC4, RFC5, RMI1, RNASEH2A, RPA2, RPA3, RRM1, SLBP, SMC1A, SMC2, SUPT16H, SVIP, TIMELESS, TIPIN, TK1, TMPO, TOPBP1, TYMS, UNG, USP1, USP37, WDHD1, WDR76, XRCC2, YEATS4, ZGRF1
		Cell cycle, other	CALM2, CCDC14, CDK11B, CENPH, CENPJ, CENPK, CENPM, CENPU, CEP152, CMC2, CSE1L, DNAJC9, DSN1, ERCC6L, EXOSC8, FBXO43, HAUS1, HAUS8, HPS4, ITGB3BP, KLHL23, KNTC1, LMNB1, MELK, MIS18A, MND1, MYBL1, NCAPD3, NCAPG2, NCAPH, NUP107, NUP85, PAICS, PSMC3IP, RANBP1, RTKN2, SPC25, SRSF10, STIL, SYNE2, TRIM37, TUBG1, VRK1

		DNA repair	ANKRD32, CDC5L, FANCB, FANCI, FANCL, FIGNL1, FUS, KIF22, MBD4, NUDT21, PARP2, PMS1, RAD51AP1, RAD54B, RIF1, SMCHD1, TTF2, UBE2T, USP10, XRCC5, ZWINT
		Histones	HIST1H1D, HIST1H1E
		Not accounted for (38/213) = 18%	*ACAA2, BAZ2B, BTG3, C19orf48, C21orf58, C3orf14, CBX5, CCDC15, CDCA4, CTDSPL2, CTNNAL1, DERA, FAM161A, FKBP5, GGCT, GGH, HMOX1, HNRNPAB, HSPB11, LGALS1, LSM8, MTHFD1, NAP1L4, POLR3K, POP7, PPM1G, PSIP1, PTPRG-AS1, RRM2, SDHA, SIVA1, SKA3, SLC43A3, SNRNP25, SRSF2, TEX30, TMEM106C, WDR34*
C	194	Unfolded protein response, ER stress	ASNS, ATF4, ATF6, ATP2A2, C19orf10, CALR, CANX, CAV1, CREB3L2, CRELD1, CRELD2, DDIT3, DDIT4, DNAJB11, DNAJB9, DNAJC10, DNAJC3, EIF4EBP1, ERP44, FAM129A, GFPT1, GLRX2, HERPUD1, HM13, HSP90B1, HSPA13, HSPA5, HSPA9, HYOU1, KDELR3, LMAN1, MANF, MTHFD2, NARS, NUPR1,

			OS9, P4HB, PDIA3, PDIA4, PDIA6, PPIB, PPP1R15A, PSAT1, RCN3, SDF2L1, SEC31A, SEL1L, SERP1, SESN2, SIL1, SPCS3, SRPR, SSR1, STT3B, TARS, TMCO1, TMED2, TRIB3, TXNIP, UGGT1, VEGFA, VIMP, WARS, WIPI1, XBP1, XPOT
		Other endoplasmic reticulum	ARF1, ARF4, ARFGAP3, ASPH, CALU, CLCN3, COPA, COPB1, COPB2, DDOST, FKBP2, GOLGA4, KDELR1, KDELR2, LAMB1, MAGT1, MIA3, MLEC, MTDH, NFE2L1, NUCB2, OSTC, PLOD3, PRNP, RCN1, RHOBTB3, RPN1, RPN2, RRBP1, SEC11C, SEC24D, SEC63, SLC35B1, SLC7A11, SMIM14, SPCS2, SSR2, SSR3, SUCO, TMED10, TMED9, TMF1, TRAM1
		tRNAs	AARS, EPRS, GARS, LARS, MARS, SARS, YARS
		Metal ion transport and homeostasis	ANXA2, ARHGEF2, ATP1A1, BEST1, CYB561, EDNRB, PDE4B, PEG10, SERPINE2, SHMT2, SLC39A14, SLC3A2, SLC5A3, TCEA1, TES

		Not accounted for (63/194) = 32%	*ACBD3, ALDH1L2, ANKRD11, ATP6V1F, BCAT1, BTG1, BUD31, C11orf24, C6orf48, CDH19, CDK2AP2, CITED1, CTC-425F1.4, DDR2, DUSP6, EIF1, FAM114A1, FBXO25, GADD45A, GAS5, GAS7, GDF15, GHITM, GOT1, GPNMB, HDLBP, IFRD1, IL1RAP, ITGA4, LARP1B, LIMA1, LMO4, LRIF1, LURAP1L-AS1, MAP4, MBNL2, MCF2L, MESDC2, MORF4L2, NRSN2-AS1, PHGDH, PHLDA1, PMP22, PPAPDC1B, PSAP, PYCR1, PYGB, RCAN1, S100A11, SELK, SEP15, SLC1A5, SLFN5, SORBS2, SPARC, SUPV3L1, TAX1BP1, TMEM263, TMX4, WDR26, ZEB2, ZFAS1, ZNF106*
D	158	Ribosomal proteins	FAU, RPL10, RPL10A, RPL11, RPL12, RPL13, RPL13A, RPL14, RPL15, RPL18, RPL18A, RPL19, RPL22, RPL23, RPL23A, RPL24, RPL26, RPL27, RPL27A, RPL28, RPL29, RPL3, RPL30, RPL31, RPL32, RPL34, RPL35, RPL35A, RPL36, RPL36AL, RPL37, RPL37A, RPL38, RPL39, RPL4, RPL5, RPL6, RPL7A, RPL8, RPL9, RPLP0, RPLP1, RPLP2, RPS10, RPS11, RPS12,

			RPS13, RPS14, RPS15, RPS15A, RPS16, RPS17, RPS18, RPS19, RPS2, RPS20, RPS21, RPS23, RPS24, RPS25, RPS27, RPS27A, RPS28, RPS29, RPS3, RPS3A, RPS4X, RPS5, RPS6, RPS7, RPS8, RPS9, RPSA, UBA52
		Protein translation	EEF1A1, EEF1B2, EEF1D, EEF2, EIF3E, EIF3F, EIF3H, EIF3K, NACA, NACA2, PABPC1, PRKCSH, SEC11A, SSR4
		Ribosome biogenesis	GLTSCR2, NHP2
		Iron metabolism	FTH1, FTL
		Not accounted for (66/158) = 42%	*ABHD14B, APP, ARL2, ATP5G2, C14orf2, C4orf48, C7orf55, CCDC86, CCDC88C, CD109, CEP350, COMMD6, COX5B, COX7A2L, CST3, CUEDC2, DCBLD2, ETV5, FBXO32, FIBP, HIF1A, HLTF, IFITM3, IMPDH2, ITSN2, LRRC75A, LRRC75A-AS1, METTL12, MIA, MME, MT-ND6, NAA38, NAP1L1, NDUFB7, NDUFS4, NIN, OAZ1, OST4, PABPC3, PGLS, PPFIA1, PRDX5, PRSS23, RB1CC1, ROMO1, RP11-193E15.4, RP11-356J5.12, RP11-669N7.2, SAT2, SERF2, SF3B6, SH3KBP1, SNHG19, SNHG5,*

			SNHG6, SUCLG2, TOMM7, TPT1, TSPO, UBL5, UQCRB, UQCRH, UQCRHL, VPS28, WNK1
E	120	Cellular respiration	ACAT1, ATP5B, ATP5G1, ATP5G3, ATP5J, ATP5J2, ATP5L, ATP5O, BNIP3, BNIP3L, C17orf89, C1QBP, CHCHD2, COX17, COX5A, COX6B1, COX6C, COX7C, COX8A, CYCS, HMBS, LGALS3, MMADHC, MRPL11, MRPL12, MRPL21, MRPL41, MRPS16, NDUFA2, NDUFA4, NDUFA8, NDUFB3, NDUFB4, NDUFB9, NDUFC1, NDUFC2, NDUFS5, NDUFS6, NPM1, PARK7, PFDN2, PFN1, PHB, PHB2, PSMB4, SDHB, SLC25A3, SLC25A36, SLC25A39, SLIRP, TIMM13, TRAP1, UQCR10, UQCRQ, VDAC1
		Glycolysis	ALDOA, ANKZF1, CD44, ENO2, FAM162A, GAPDH, LDHA, ME1, MIF, P4HA1, PGK1, PKM, PPIA, SLC2A3, TPI1, TXN
		Hypoxia, oxidative stress	ADM, ARHGDIA, ATXN2L, C20orf27, EIF5A, GBE1, GLO1, HINT1, HSPB1, ITGAE, KDM3A, NME1, NRN1, POLR2L,

			PSMA4, PSMB2, PSMB6, S100A10, STRA13, TAF1D, TNS1, TRIB2, VIM
		Splicing	SNRPD2, SNRPF, THOC7, YBX1
		Endopeptidase inhibition	CST1, CST4, CSTB
		Not accounted for (18/120) = 15%	*APRT, BTF3, C12orf57, CCL28, CFL1, DARS, EIF4A1, EIF4A2, H1F0, HPCAL1, LMAN2, RP11-58E21.4, SDF4, SLAMF9, TMED4, TRAPPC1, TXNDC17, USP11*
F	110	Pigmentation, melanosome	APOE, ASAH1, ATOX1, B2M, BIRC7, CALM1, CAPG, CD320, CD63, CDK2, CLEC11A, CTSA, CTSB, CTSD, CTSH, DAB2, EMP1, ERP29, GNPTAB, GPR56, GRN, GSTO1, GUSB, GYPC, HMCN1, IDH1, MFGE8, MLANA, MLPH, MRPL44, MZT2B, NDUFB1, NGFRAP1, NPAS2, NPC2, NUMA1, PMEL, PRDX1, PRDX4, PSMA7, RAB27A, RAB38, RAI14, RLBP1, SDCBP, SLC24A5, TMEM98, TYR, UBR5, WSB1
		Myeloid phenotypes (monocyte, macrophage,	ACOT7, AKR1A1, ARPC1B, ATP6AP1, ATP6V0E1, BHLHE41, BTN2A2, C21orf91, CD59, CD9, CHCHD6, CPM, DBI, HEXB,

		basophil, dendritic cell)	HIPK2, ITM2C, LGALS3BP, LITAF, LRPAP1, MDH1, METTL9, NDUFB6, PEBP1, QPCT, RGS10, SPAG9, SUMO2, TIMP1, TMEM147, TMSB4X, TOP1, TRIM63
		Not accounted for (28/110) = 25%	*AC104655.3, AFG3L2, BCAR3, CAPN3, CFAP61, COL4A3BP, CYTL1, G3BP1, G3BP2, GJB1, HTATSF1, KIAA1456, LHFPL3-AS1, LONRF1, MAGED1, MAGI1, MPG, PCCB, PDE3B, PLCH1-AS1, PMP2, POLR2F, RNF19A, RP4-718J7.4, SCML4, SEPT6, STK32A, TIMM50*
G	70	HNRNPs	HNRNPA2B1, HNRNPA3, HNRNPH3, HNRNPK, HNRNPM, HNRNPR, HNRNPU
		Other RNA processing	DDX1, DDX21, DNTTIP2, EIF4A3, EIF4G1, ESF1, LUC7L3, MYH10, NCL, NOLC1, NOP56, NOP58, NSUN2, NUDT1, PA2G4, PNO1, PNPT1, PRPF40A, PUS7, SET, SNRPC, SNRPE, SNRPG, SSB, SYNCRIP, TPRKB, TSR1, WDR3
		Other RNA binding proteins	BZW1, CNBP, EIF3A, EIF3B, GSPT1, LARP4, SERBP1

		Protein chaperones	DNAJC2, DNAJC8, CCT4, CCT5, CCT6A, HSP90AB1, HSPD1, HSPE1, NUDCD1, STIP1, TCP1
		Not accounted for (17/70) = 24%	*ARPC2, ATP6V1C1, CEBPZ, GPATCH4, KDM5A, KIAA0020, MACF1, MRPL19, NOM1, PSMD14, PSME4, PTMA, TLN1, TPM3, UTP11L, YWHAQ, ZC3H15*
H	59	Mitochondrially-encoded genes	MT-ATP6, MT-ATP8, MT-CO1, MT-CO2, MT-CO3, MT-CYB, MT-ND1, MT-ND2, MT-ND3, MT-ND4, MT-ND4L, MT-ND5, MT-RNR1, MT-RNR2, MT-TF, MT-TL1, MT-TP, MT-TS2, MT-TT, MT-TV
		Regulation of expression of mitochondrially-encoded genes	LRPPRC
		Protein ubiquitination	CBLB, DDX3X, DZIP3, HECTD1, MYCBP2, NBR1, TBL1XR1
		Hypoxia/stress response	CALD1, DST, FOS, IGF1R, IGFBP5, JUNB, MXI1, ZMYND8
		Not accounted for (20/59) = 34%	*AC021451.1, ADCY1, AKAP9, BPTF, CAPN2, COX6A1, CSDE1, DDX17, FCHSD2, GOLGB1, KRIT1, MEF2A, PCLO, PLCB4,*

			PPP1R9A, SPEN, SRRM2, TDRD3, TIA1, ZBTB38, ZC3H13, ZKSCAN1, ZNF704
I	45	S100 proteins	S100A1, S100A13, S100A4, S100A6
		Endosome function	ARFIP1, INPP5F, PIK3C3
		Lipoprotein metabolism	APOC1, APOD
		Not accounted for (36/45) = 80%	*ATP6V0B, ATPIF1, BHLHE40, CDKN2A, CKLF, COPS6, CTHRC1, FRMD4A, FXYD3, HOXB2, MGST3, MRPS21, MT2A, NDUFA5, NFKBIZ, PAXIP1-AS1, PLEKHA4, PTPRE, RABAC1, RAMP1, RPS27L, SEMA3B, SH3BGRL3, SLC20A1, STRIP2, TDRKH, TM4SF1, TMEM258, TMSB10, TNFRSF12A, TPD52L1, TSC22D1, TUBA1A, USP53, VGF, ZNHIT1*
J	29	Hypoxia	AKAP12, KLHL24, RND3, SAMD4A, SAT1
		Golgi and lysosome	EXOC1, GOLGA8B, LRRK2, PRELP, RAB30, SMPD1, SPG11
		Not accounted for (17/29) = 59%	*ARID5B, BCL2L11, CPQ, CRYL1, CSAG1, EYA3, HIST1H1C, INTU, KDM1B, LPP, RAB17, RP11-258C19.7, SH3BGRL, STARD13, TRIQK, UPF2, ZNF451*

K	20	Cholesterol, sterol biosynthesis	ACSS2, C14orf1, CYP51A1, DHCR24, DHCR7, EBP, FDFT1, FDPS, HMGCR, HMGCS1, INSIG1, LDLR, MSMO1, SC5D, SCD, SQLE, STARD4, TMEM97, LPIN1
		Not accounted for (1/20) = 5%	*PRRX1*
L	12	Interferon response	CEACAM1, HERC5, IFI44, IFIH1, IFIT2, IFIT3, ISG15, PMAIP1
		Not accounted for (4/12) = 33%	*KIN, MRPL55, P2RX7, PPM1K*
M	7	Regulation of growth factor signaling	FAM98B, LEMD3, PTPN1
		RNA splicing	SAP18
		Not accounted for (3/7) = 43%	*GALNT2, MRPL57, SLC2A11*

Table 1. Gene communities identified using cell subcluster 1.2.
Communities containing at least 7 genes are shown. Genes were assigned to annotations after manually exploring overlaps with all MSigDB datasets (category labels used here often reflect the merger of redundant or semi-redundant gene sets). Genes marked "Not accounted for" failed to overlap significantly with known gene sets (i.e., overlaps involved at most two genes, or accounted for less than 1% of genes in the dataset).

2.3.6 Analysis of gene communities

Figure 4E summarizes the results of using BigSur to identify significant correlations (FDR<0.02) in each of the cell clusters described above (1.1, 1.2, 2 and 3). Because statistical power to detect correlations falls with number of cells analyzed, one might expect to see the fewest significant correlations in the smallest clusters, but this was not the case. Instead, the clusters with

the largest number of significant correlations were those with the highest average sequencing depth (Fig. 4C), suggesting that data sparsity has an especially strong influence on correlation detection.

Schadt and colleagues have recently argued [55] that negative correlations may be considered a strong indicator of cell heterogeneity. These authors argue that minimization of the proportion of negative gene-gene correlations provides a principled metric for determining when to cease sub-clustering cells. Consistent with this view, we observed that, as cells were successively subclustered, the proportion of negative correlations fell from 45% to about 20% (Fig. 4E). The clusters with the lowest proportion of negative correlations were clusters 1.2, 2 and 3. As cluster 2 had very high levels of mitochondrially-encoded genes (33.7% of UMI), and cluster 3 had the lowest average UMI/cell (1,273) of any cluster, we focused our analysis on cluster 1.2, which had both low levels of mitochondrial genes (9.7% of UMI) and high average UMI/cell (6,252). .

Within cluster 1.2, we found that enrichment for correlated paralog pairs and genes whose products interact physically was much higher than when all 8,640 cells had been considered together. Specifically, among the positive correlations, paralog pairs were enriched 55-fold, and links supported by protein-protein interactions 32-fold, over what would have been expected by chance. Indeed, 15% of all positive correlations in cluster 1.2 corresponded to links supported by known protein-protein interactions.

Cell cluster	Community	Number of genes	Gene names
1.1	Mitochondrially-encoded	110	AKAP9, ANKRD37, ARGLU1, ATP1A1, BHLHE40, C21orf58, CD109, CDK5RAP3, CIR1, COPB1, CTSC, DDIT4, DDX17, DDX5, DMTF1, DNAJB1, DST, EIF2A, EIF3D, EIF4A2, EIF4G1, EMP1, EPRS, EWSR1, FAM168A, FNBP4, FUS, GAS5, GOLGA8B, GPNMB, GRN, HERC5, HSP90AB1, HSPA9, HSPD1, IFIT1, IFIT2, IFIT3, IGF2R, ING2, IRF1, ISG15, IVNS1ABP, LAMB1, LGALS3BP, LYST, MACF1, MAN2C1, MCM6, **MT-ATP6, MT-ATP8, MT-CO1, MT-CO2, MT-CO3, MT-CYB, MT-ND1, MT-ND2, MT-ND3, MT-ND4, MT-ND4L, MT-ND5, MT-ND6, MT-RNR1, MT-TD, MT-TF, MT-TL1, MT-TP, MT-TS2, MT-TV,** MXI1, MYC, MYO10, NCOA3, NDUFB9, P4HA1, P4HB, PABPC1L, PDIA3, PDIA6, PMAIP1, PMEL, PNISR, PON2, PPFIA1, PRKDC, PSAP, PTPRM, RASGRP3, RGS1, RRBP1, SAT1, SEC31A, SERINC1, SF3B1, SLC1A3, SLC2A3, SLC35F6, SLC3A2, SNHG12, SORBS2, SPEN, SPTBN1, SRRM2, TAF1D, TLN1, TSR1, TYR, UBR5, VIM, ZFAS1

1.2	Mitochondrially-encoded	59	AC021451.1, ADCY1, AKAP9, BPTF, CALD1, CAPN2, CBLB, COX6A1, CSDE1, DDX17, DDX3X, DST, DZIP3, FCHSD2, FOS, GOLGB1, HECTD1, IGF1R, IGFBP5, JUNB, KRIT1, LRPPRC, MEF2A, **MT-ATP6, MT-ATP8, MT-CO1, MT-CO2, MT-CO3, MT-CYB, MT-ND1, MT-ND2, MT-ND3, MT-ND4, MT-ND4L, MT-ND5, MT-RNR1, MT-RNR2, MT-TF, MT-TL1, MT-TP, MT-TS2, MT-TT, MT-TV,** MXI1, MYCBP2, NBR1, PCLO, PLCB4, PPP1R9A, SPEN, SRRM2, TBL1XR1, TDRD3, TIA1, ZBTB38, ZC3H13, ZKSCAN1, ZMYND8, ZNF704
2	Mitochondrially-encoded	50	APOD, APP, ATP6AP1, BSG, CD109, COX6A1, CPVL, CTSA, CTSD, CTSK, GPNMB, GRN, GUSB, IGSF8, LGALS3BP, LRPAP1, MAGED2, MCUR1, MGST3, **MT-ATP6, MT-ATP8, MT-CO1, MT-CO2, MT-CO3, MT-CYB, MT-ND1, MT-ND2, MT-ND3, MT-ND4, MT-ND4L, MT-RNR1, MT-RNR2, MT-TF, MT-TL1, MT-TP, MT-TS2, MT-TT, MT-TV,** NCSTN, NDUFB9, P4HB, PCNXL2, PLAT, PLTP, PMEL, PSAP, RPN2, SPARC, SRPX, TYR

3	Mitochondrially-encoded	15	COX6A1, GPNMB, **MT-ATP6, MT-ATP8, MT-CO2, MT-CO3, MT-CYB, MT-ND1, MT-ND2, MT-ND3, MT-ND4, MT-ND4L, MT-ND5, MT-TL1,** PMEL
1.1	Ribosomal protein genes	48	FTH1, **RPL10, RPL11, RPL12, RPL13, RPL13A, RPL18, RPL19, RPL24, RPL26, RPL27, RPL27A, RPL28, RPL30, RPL31, RPL34, RPL35, RPL37, RPL37A, RPL38, RPL8, RPLP0, RPLP1, RPLP2, RPS11, RPS12, RPS13, RPS14, RPS15, RPS15A, RPS16, RPS17, RPS18, RPS19, RPS2, RPS20, RPS21, RPS23, RPS27, RPS27A, RPS29, RPS3, RPS5, RPS6, RPS8,** SNHG5, TPT1, UBA52

1.2	Ribosomal protein genes	158	ABHD14B, APP, ARL2, ATP5G2, C14orf2, C4orf48, C7orf55, CCDC86, CCDC88C, CD109, CEP350, COMMD6, COX5B, COX7A2L, CST3, CUEDC2, DCBLD2, EEF1A1, EEF1B2, EEF1D, EEF2, EIF3E, EIF3F, EIF3H, EIF3K, ETV5, **FAU**, FBXO32, FIBP, FTH1, FTL, GLTSCR2, GNB2L1, HIF1A, HLTF, IFITM3, IMPDH2, ITSN2, LRRC75A, LRRC75A-AS1, METTL12, MIA, MME, MT-ND6, NAA38, NACA, NACA2, NAP1L1, NDUFB7, NDUFS4, NHP2, NIN, OAZ1, OST4, PABPC1, PABPC3, PGLS, PPFIA1, PRDX5, PRKCSH, PRSS23, RB1CC1, ROMO1, RP11-193E15.4, RP11-356J5.12, RP11-669N7.2, **RPL10, RPL10A, RPL11, RPL12, RPL13, RPL13A, RPL14, RPL15, RPL18, RPL18A, RPL19, RPL22, RPL23, RPL23A, RPL24, RPL26, RPL27, RPL27A, RPL28, RPL29, RPL3, RPL30, RPL31, RPL32, RPL34, RPL35, RPL35A, RPL36, RPL36AL, RPL37, RPL37A, RPL38, RPL39, RPL4, RPL5, RPL6, RPL7A, RPL8, RPL9, RPLP0, RPLP1, RPLP2, RPS10, RPS11, RPS12, RPS13, RPS14, RPS15, RPS15A, RPS16, RPS17, RPS18, RPS19, RPS2, RPS20,**

			RPS21, RPS23, RPS24, RPS25, RPS27, RPS27A, RPS28, RPS29, RPS3, RPS3A, RPS4X, RPS5, RPS6, RPS7, RPS8, RPS9, RPSA, SAT2, SEC11A, SERF2, SF3B6, SH3KBP1, SNHG19, SNHG5, SNHG6, SSR4, SUCLG2, TOMM7, TPT1, TSPO, UBA52, UBL5, UQCRB, UQCRH, UQCRHL, VPS28, WNK1

2	Ribosomal protein genes	440	ACTB, ACTG1, AKR1B1, ALDOA, ANAPC13, ANP32A, ANP32B, ANXA5, AP2S1, APOA1BP, APRT, ARL6IP5, ARPC1A, ARPC1B, ARPC2, ARPC3, ATOX1, ATP5B, ATP5E, ATP5EP2, ATP5F1, ATP5G1, ATP5G2, ATP5G3, ATP5I, ATP5J, ATP5J2, ATP5L, ATP5O, ATP6V0B, ATP6V1F, ATP6V1G1, ATPIF1, BCAS3, BCCIP, BOLA3, BTF3, BUD31, C11orf31, C12orf57, C14orf2, C17orf89, C19orf10, C19orf53, C1QBP, C20orf27, C8orf33, CACYBP, CALM1, CALM2, CAPG, CAPZB, CBX1, CBX3, CCNI, CCT7, CD59, CD63, CDC42, CDKN2A, CETN2, CFL1, CHCHD2, CHCHD6, CHCHD7, CHMP2A, CHMP4B, CKS2, CNBP, CNIH1, COA4, COX14, COX17, COX4I1, COX5A, COX5B, COX6B1, COX6C, COX7A2, COX7B, COX7C, COX8A, CST3, CSTB, CTHRC1, CTSH, CWC15, CYC1, CYCS, CYTL1, DBI, DDOST, DGUOK, DNAJB1, DNAJC8, DRG1, DSTN, DTD1, DUT, DYNLL1, DYNLRB1, ECHS1, EDF1, EEF1A1, EEF1B2, EEF1D, EIF1, EIF2S2, EIF3H, EIF3I, EIF4EBP1, EIF5A, EMC4, ENO1, ENY2, ERH, ESD, **FAU**, FBL, FIS1, FKBP1A, FKBP2, FTH1, FTL,

			FXYD5, GADD45GIP1, GAPDH, GHITM, GLO1, GMNN, GNAS, GNB2L1, GNG12, GPX1, GPX4, GSTO1, GSTP1, GUK1, GYPC, H2AFZ, H3F3B, HDDC2, HINT1, HIST1H4C, HMGB1, HMGN2, HNRNPC, HSBP1, HSPB1, HSPE1, INSIG1, ITGB1BP1, LAGE3, LAMTOR1, LAMTOR4, LAMTOR5, LAPTM4B, LDHA, LDHB, LGALS1, LGALS3, LINC00998, LRRC75A, LRRC75A-AS1, LSM4, LSM7, METTL9, MIA, MIF, MKKS, MLANA, MMADHC, MRPL12, MRPL13, MRPL21, MRPL33, MRPL48, MRPL51, MRPL57, MRPS33, MRPS34, MRPS6, MT2A, MTPN, MYEOV2, MYL6, MZT2B, NAA38, NACA, NAP1L1, NDUFA1, NDUFA12, NDUFA13, NDUFA4, NDUFA5, NDUFA8, NDUFAB1, NDUFB1, NDUFB10, NDUFB11, NDUFB2, NDUFB3, NDUFB4, NDUFC1, NDUFC2, NDUFS5, NDUFS6, NEDD8, NGFRAP1, NHP2, NHP2L1, NME1, NME4, NOL7, NPC2, NPM1, NSA2, NUDT1, NUTF2, OAZ1, OLA1, OST4, OSTC, PA2G4, PABPC1, PABPC3, PARK7, PDAP1, PDCD5, PEBP1, PFDN2, PFDN4, PFDN5, PFN1, PHB, PHB2, PKM, PMP22,

			POLR2F, POLR2J, POLR2L, POMP, PPDPF, PPIA, PPIB, PPT1, PRDX1, PRDX5, PRDX6, PRELID1, PSMA5, PSMA7, PSMB4, PSMB6, PSMB7, PSME1, PSMG1, PTMA, PTTG1IP, PYURF, RAB2A, RAB38, RAB7A, RAN, RBX1, REXO2, RGS10, RHOA, ROMO1, RP11-669N7.2, RP11-831H9.11, **RPL10, RPL10A, RPL11, RPL12, RPL13, RPL13A, RPL14, RPL15, RPL18, RPL19, RPL22, RPL23, RPL23A, RPL24, RPL26, RPL27, RPL27A, RPL28, RPL29, RPL3, RPL30, RPL31, RPL32, RPL34, RPL35, RPL35A, RPL36, RPL36AL, RPL37, RPL37A, RPL38, RPL39, RPL4, RPL5, RPL6, RPL7A, RPL8, RPL9, RPLP0, RPLP1, RPLP2, RPS10, RPS11, RPS12, RPS13, RPS14, RPS15, RPS15A, RPS16, RPS17, RPS18, RPS19, RPS19BP1, RPS2, RPS20, RPS21, RPS23, RPS24, RPS25, RPS27, RPS27A, RPS27L, RPS28, RPS29, RPS3, RPS4X, RPS5, RPS6, RPS7, RPS8, RPS9, RPSA,** RSL1D1, RSL24D1, RTFDC1, S100A1, S100A10, S100A11, S100A13, S100A6, S100B, SCAND1, SEC11A, SEC61G, SEP15, SERF2, SERP1, SET, SF3B6, SH3BGRL,

			SH3BGRL3, SHFM1, SKP1, SLC25A3, SLC25A5, SLIRP, SLMO2, SMS, SNHG5, SNHG6, SNHG8, SNRPC, SNRPD1, SNRPD2, SNRPD3, SNRPE, SNRPF, SNRPG, SOD1, SPCS1, SRP14, SRP72, SRSF3, SSBP1, SSR3, SSR4, ST13, STMN1, SUB1, SUMO2, SUMO3, TBCA, TCEA1, TCEAL8, TCEB1, TCEB2, THOC7, TIMP1, TMA7, TMBIM6, TMED2, TMEM14A, TMEM14B, TMEM14C, TMSB10, TMSB4X, TOMM20, TOMM6, TOMM7, TPD52, TPI1, TPT1, TRAPPC1, TSPAN3, TUBA1B, TXN, TXN2, TXNDC17, UBA52, UBB, UBE2L3, UBE2N, UBL5, UBXN1, UQCC2, UQCR10, UQCRB, UQCRH, UQCRQ, USMG5, UTP11L, UXT, VDAC1, VIM, YBX1, YWHAE, YWHAZ, ZNF706, ZNHIT1

3	Ribosomal protein genes	63	ACTG1, ATOX1, CD63, CHCHD2, COX7C, **FAU**, FTH1, FTL, H2AFZ, HSPE1, LGALS3, MLANA, PRDX1, PSMA7, **RPL10, RPL11, RPL12, RPL13, RPL13A, RPL15, RPL18, RPL19, RPL23, RPL24, RPL26, RPL27, RPL27A, RPL28, RPL31, RPL32, RPL34, RPL35, RPL35A, RPL37, RPL37A, RPL38, RPL5, RPL8, RPLP0, RPLP2, RPS11, RPS12, RPS13, RPS14, RPS15, RPS15A, RPS18, RPS2, RPS20, RPS23, RPS24, RPS25, RPS27A, RPS29, RPS3, RPS4X, RPS6, RPS8,** TMSB10, TPI1, TPT1, UBA52, UQCRB

Table 2. Mitochondrially-encoded and ribosomal protein gene communities identified following unsupervised clustering of each of the four cell clusters, 1.1, 1.2, 2 and 3.

Table 1-2 and Figures S4-S7 (additional files 6-9) summarize results for 13 of the largest gene communities detected in cluster 1.2, which collectively account for 1,291 of the 2,519 genes that showed any significant positive correlations. Table 1 groups the genes of each community into functional categories according to annotations found in the most recent release of the MSigDB database [56, 57]. Annotations identified using the Database for Annotation, Visualization and Integrated Discovery (DAVID [58]) are also shown in Table 2. In 11 of the 13 communities (accounting for 1,217 of the 1,291 genes), over 50% of the genes could be associated with just one or a few functional annotations (Table 1).

For example, communities A and B consist primarily of genes related to the cell cycle, with more than 62% of the genes in community A known to be preferentially associated with the G2 and M phases of the cell cycle. About 71% of community B is associated with the cell cycle, the majority of these being associated with G1 and S phase. These communities were also correlated with each other, but the community-finding algorithm easily subdivided them.

Most other communities are at best weakly correlated with either of the cell-cycle communities. In community C, 34% of genes encode proteins involved in the unfolded protein response and/or endoplasmic reticulum (ER) stress, and other ER proteins account for another 22% of genes. ER stress is commonly observed in cancer, and the unfolded protein response is specifically and strongly activated in melanoma cells [59-61].

Community D contains almost all genes encoding protein components of cytoplasmic ribosomes, plus a large set of genes that regulate ribosome biogenesis or function. Overall, ribosome-related genes make up 57% of this community. Community E combines genes involved in cellular respiration, glycolysis, and oxidative and hypoxic stress, which collectively account for 78% of this community.

Fifty genes in community F, nearly half the total, are associated with melanin synthesis and melanosome biogenesis, and include traditional melanocyte markers such as *MLANA*, *PMEL*, *RAB27A* and *TYR*. Another subset in community F, consisting of 32 genes, shares annotations related to markers of myeloid cell types (monocytes, macrophages, basophils, etc.) but largely consists of ubiquitously expressed genes involved in processes such as lipid biosynthesis. At least one of these genes, *TRIM63*, is known to be strongly associated with melanoma [62], and is a validated target of MITF [63], the primary transcription factor controlling expression of pigmentation genes.

Community G contains multiple genes involved in RNA processing, including seven genes encoding members of the ubiquitously expressed heterogeneous nuclear ribonucleoprotein (HNRNP) family, as well as genes encoding protein chaperones of the HSP10, HSP40 HSP60, HSP90, and CCT families. Community H contains most of the mitochondrially-encoded genes, as well as genes involved in protein ubiquitination and the hypoxia stress response.

Community I combines four genes encoding S100-family calcium binding proteins, as well as various genes associated with endosome function and lipoprotein metabolism. Unlike other communities, in this community most genes do not fall into large groups with traditional annotations. However, many of them are genes known to be strongly associated with melanoma. These include *APOC1*, *APOD*, *CDKN2A*, *CTHRC1*, *FXYD3*, *INPP5F*, *MT2A*, *S100A4, SEMA3B, TMSB10* and *VGF* [64-67], suggesting this community is detecting genes co-regulated by drivers of a melanoma-specific cell state.

Community J combines genes associated with the response to hypoxia with genes involved in Golgi body and lysosome function. Nearly all the genes in community K are associated with cholesterol/sterol biosynthesis and homeostasis; the only exception is *PRRX1*, which serves as the transcriptional co-regulator of serum response factor. In oligodendrocytes at least, it has been shown that PRRX1 is necessary for the expression of cholesterol biosynthesis genes [68].

Community L consists mainly of genes annotated as related to interferon signaling; these genes are also the primary drivers of the cellular anti-viral response. Finally, community M contains several genes associated with growth factor signaling.

Figures S4-S7 (additional files 6-9) display the gene-gene linkages within these communities graphically. Here, genes are shown as light blue disks, except for transcription factors which appear as yellow squares. The areas of the disks and squares are proportional to the relative

mean expression of the genes (absolute scaling differs between panels, as it was adjusted to enhance the readability of each figure). Green lines denote significant positive correlations (no negative correlations were observed within any of the communities shown). In several cases, a smaller version of each graph, in which links supported by protein-protein interaction data have been overlayed with brown lines, is displayed as an inset; for some communities, only the graphs containing these highlights are shown. Examination of Fig. S4-S7 shows that genes associated with a single functional annotation, or that encode directly-interacting proteins, are often especially densely interconnected, causing them to cluster together (e.g., genes encoding directly physically interacting proteins in A and B, unfolded protein response genes in C, ribosomal subunit genes in D, glycolysis genes in E, S100 genes in I, etc.).

Whereas the clustering of genes into communities had been carried out based on positive gene-gene correlations, plotting pairs of communities together enabled the evaluation of negative correlations between them, the vast majority of which involved the mitochondrially-encoded genes which, as mentioned previously, strongly anti-correlated with ribosomal protein genes (Fig. 5). Mitochondrial genes also anti-correlated with some of the genes in communities A, F, E and G; for example, anticorrelations between mitochondrial genes and genes encoding glycolytic enzymes are shown in Fig. 5E.

Figure 5 and Table 3 also document that the anti-correlation between mitochondrial and ribosomal genes is as apparent in cell clusters 1.1, 2 and 3 as it is in cluster 1.2. This is interesting insofar as the features used to subdivide cells into these clusters included most of mitochondrially-encoded and ribosomal genes. What this suggests is that there are not two separable populations of cells overexpressing mitochondrial versus ribosomal genes. Instead, the data suggest that the observed anti-correlations are not themselves sufficiently correlated with each other to drive cell

clustering. In other words, BigSur seems to able to detect local gene-gene relationships that could not have been identified by comparing differential expression of genes in any possible grouping of cells.

2.3.7 Estimating the accuracy of BigSur

The association of a small number of annotation categories with nearly all of the gene communities that BigSur found, in an unsupervised manner, in scRNAseq data strongly suggest that many of the correlation s detected by BigSur are true positives. Nevertheless, in nearly all communities some genes did not fall into obvious categories (Table 1). Do these just reflect the incompleteness of existing annotations, or did BigSur produce false positive results beyond the ~2% expected from the FDR cutoff that was used?

Rarely in biology does one have ground truth information with which to settle this question, but one can still perform informative tests. For example, one can consider a subset of the genes that form a functional category and ask whether BigSur correctly identifies other members of the category as being correlated with them. An example is shown in Figure 6A, where we started with a 27-gene panel, the "Reactome Cholesterol Biosynthesis" gene set from MSigDB (*ACAT2, ARV1, CYP51A1, DHCR24, DHCR7, EBP, FDFT1, FDPS, GGPS1, HMGCR, HMGCS1, HSD17B7, IDI1, IDI2, LBR, LSS, MSMO1, MVD, MVK, NSDHL, PPAPDC2, PMVK, SC5D, SQLE, SREBF1, SREBF2, TM7SF2)*. That panel includes many, but not all, genes known to be involved in cholesterol metabolism. The graph displays all statistically significant positive correlations involving any of these genes that were detected in cell cluster 1.2 (members of the gene panel are indicated with large font, blue lettering).

All detected members of the gene set formed a single network, with many internal positive linkages. Closely connected to this network were eight additional genes (outlined in red) that are

not part of the Reactome Cholesterol Biosynthesis set, but belong either to the "Hallmark" cholesterol homeostasis set from MSigDB (*ACSS2, ACTG1, LDLR, SCD, STARD4, TMEM97*); are designated as core cholesterol metabolism genes in recent literature (*ACLY* [69]; *C14or1/ERG28* [70]); or encode sterol binding proteins that serve as feedback controllers of cholesterol biosynthesis (*INSIG1* [71]). Highlighted in orange are two additional genes that regulate lipid biosynthesis more generally, *LPIN1* and *PRRX1*.

Four additional genes are highlighted in gray, as it is possible they are also related to cholesterol metabolism: AKR1A1, CALR, PRDX1, and PMEL. *AKR1A1* encodes an enzyme of the aldo/keto reductase superfamily, which reduces a wide variety of carbonyl-containing compounds to their corresponding alcohols; its orthologues *AKR1C1* and *AKR1D1* are well known to play a role in the metabolism of steroid hormones but *AKR1A1* is thought to act on non-steroid compounds (perhaps this assumption should be revisited). *CALR*, encodes calreticulin, a protein that promotes folding, oligomeric assembly and quality control in the endoplasmic reticulum. As cholesterol synthesis takes place in the ER membrane, it is perhaps not surprising that expression of *CALR* and cholesterol synthesis genes would be co-regulated. *PRDX1* encodes peroxiredoxin, an antioxidant enzyme that reduces hydrogen peroxide and alkyl hydroperoxides and plays a key role in maintaining redox balance. In macrophages, PRDX1 has been shown to be critical for cholesterol efflux during autophagy [72]. *PMEL* encodes a major component of the melanosome. Interestingly, it has been reported that cholesterol strongly stimulates melanogenesis in melanocytes and melanoma cells [73].

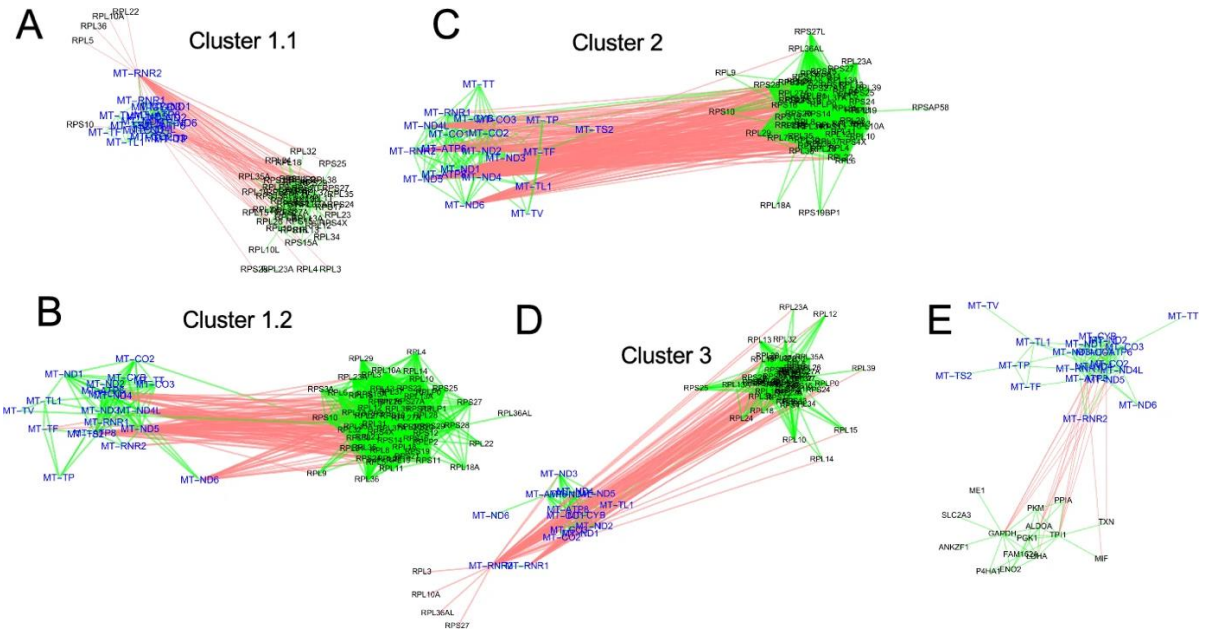


Figure 5. Mitochondrial communities are a source of strong anti-correlations.
Mitochondrially-encoded and ribosomal protein gene communities were identified in all clusters. Anti-correlations identified specifically between mitochondrially-encoded genes and ribosomal protein genes within the different clusters are shown in A (cluster 1.1), B (cluster 1.2), C (cluster 2) and D (cluster 3). Panel E shows additional anticorrelations in cluster 1.2 involving mitochondrially-encoded genes and glycolysis genes. For ease of readability, mitochondrial genes have been highlighted in blue. Red links refer to negative correlations; green to positive.

In addition to these genes, a set of tightly interconnected genes that have no known association with cholesterol metabolism is peripherally connected to this community (outlined by a dotted line in Figure 6A). A large proportion of these genes encode proteins associated with the cell cycle, especially with functions associated with mitosis, such as mitotic checkpoint events, and cyto- and karyokinesis. Strongly connected to these mitotic genes is *LBR*, which encodes the Lamin B receptor. LBR is a multifunctional protein: it plays a key role in cholesterol biosynthesis, catalyzing the reduction of the C14-unsaturated bond of lanosterol, but also anchors the nuclear lamina and heterochromatin to the inner nuclear membrane, and in so doing is strongly regulated by phosphorylation during the cell cycle [74]. A functional link between expression of cholesterol biosynthesis genes and mitosis may arise from that fact that cholesterol synthesis normally rises

dramatically during G1 phase and if prevented from doing so will result in G1 arrest [75]. It thus may make sense to upregulate the production of cholesterol biosynthetic genes as cells go into mitosis, so they are available to act in the G1 phase that immediately follows.

To the right of the graph in Figure 6A the genes of the Reactome Cholesterol Biosynthesis set are listed alongside their relative expression levels (UMI/cell after normalization) in this data set. Highlighted in yellow are those for which BigSur identified significant positive correlations, nearly all of which were toward the higher end of expression levels. These results illustrate an important limitation of any correlation methodology, which is that statistical power depends on the number of non-zero entries in the data, which will, in turn, reflect expression level, sequencing depth, and the number of cells analyzed. Examination of the raw data indicates that the point at which BigSur begins to fail to identify significant correlations occurred, for these genes, when the total number of non-zero entries (among 1582 cells) fell to about 320; this corresponded to a mean expression level of about 0.3 UMI/cell. For more strongly correlated genes one should of course expect fewer entries to be necessary to achieve significance, but based on the results here we suggest that, when seeking to identify the kinds of weak correlations associated with noise-coupled gene-regulatory networks, a minimum of several hundred non-zero entries per gene may be a reasonable threshold. In practice, one should be able to adjust the number of genes that fulfill this criterion either by varying the number of cells analyzed or varying the depth of sequencing (or both).

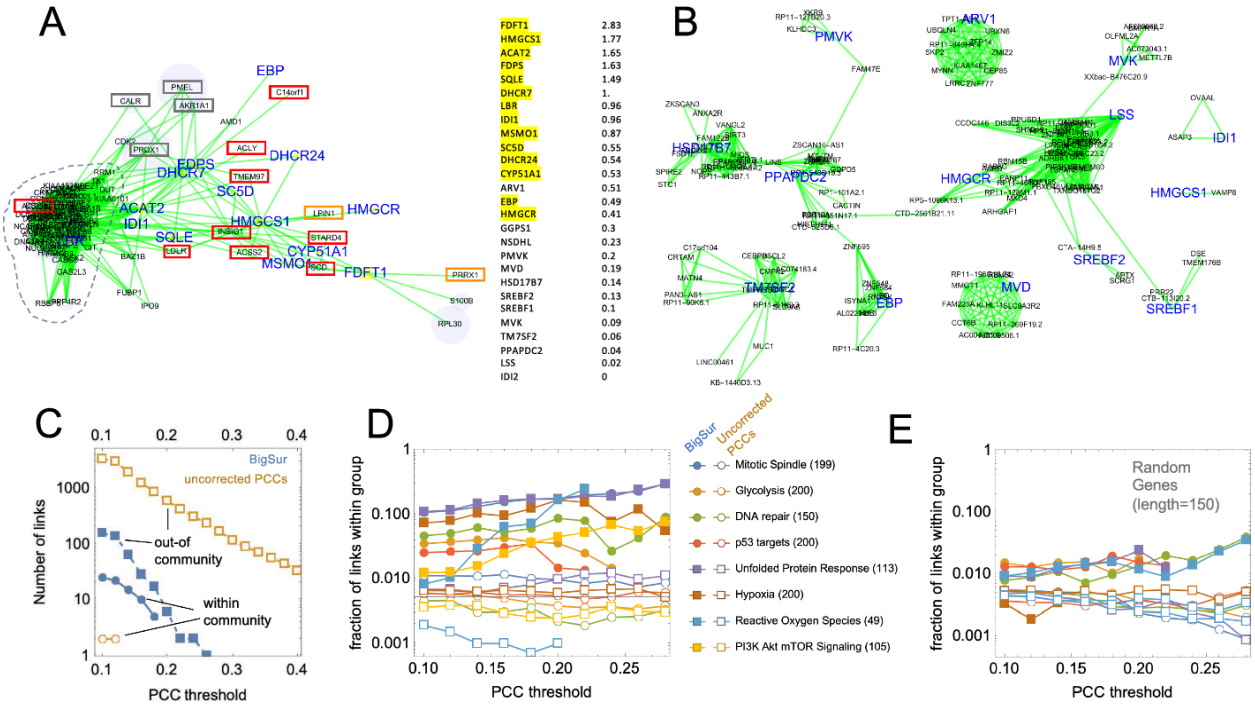


Figure 6. Comparing the ability of BigSur and uncorrected PCCs to identify biologically significant correlations.

Using the data from cell cluster 1.2, BigSur identified positive correlations and their p-values for all genes, converting the p-values to equivalent PCCs. In addition, the same starting data were also normalized and uncorrected PCCs obtained. A-D compare the correlations identified by the two methods. A. Genes identified by BigSur as correlating with the Reactome Cholesterol Metabolism gene set. All significant gene-gene correlations detected by BigSur involving genes in this 27-gene set (the names of which are shown at right) and all other genes in the genome are shown graphically. Genes belonging to the set are highlighted in blue; additional known cholesterol metabolism-related genes are highlighted with rectangles. Green lines show significant positive correlations. Dashing surrounds a group of cell cycle genes that correlate with the dual-function gene LBR. Expression levels (mean UMI per cell after normalization) for each of the genes in the set are shown at right; genes highlighted in yellow are those that displayed any significant correlations. B. Correlation communities for the same gene set, panel identified using uncorrected PCCs, visualized as in panel A. Note that the target genes are scattered among 5 disconnected communities and are connected with genes with no obvious relationship to cholesterol metabolism. C. Positive correlations involving any of the genes belonging to the Reactome Cholesterol Metabolism gene set were divided into “within-community” links and “out-of-community” links. With BigSur (filled symbols), the ratio of within-community to out-of-community links is much higher than with uncorrected PCCs (open symbols), suggesting the latter produce much less enrichment for functionally relevant connections. D. Analyses similar to that in panel C were performed for eight additional MSigDB “Hallmark” gene sets. Plotted are the proportions of correlations that are within-group over a range of PCC thresholds. Numbers of genes in each panel are shown in parentheses. E. Analyses similar to that in panel D were performed with 150-gene sets chosen at random after first removing genes with mean expression below 0.052 (so that median expression for random genes was similar to that of the functional panel genes.

Fig. 6B plots the results of the identical exercise as in Fig. 6A, carried out using uncorrected PCCs obtained directly from normalized gene expression data, rather than using PCC′ and BigSur. A threshold of PCC = 0.29 was selected so that the number of Reactome Cholesterol Biosynthesis genes that exhibited above-threshold correlations was the same as in Fig. 6A. In this case, however, the cholesterol biosynthesis genes did not form a single community, but rather separated into 5 disconnected groups. Most groups contained links to many genes, none of which appeared, on inspection, to bear any relationship to cholesterol metabolism. These data make the important point that *most of the links detected using uncorrected PCCs are likely to be false positives*.

One way to extend this analysis to other functional gene sets is suggested by the observation that, among the correlations in Fig. 6A involving Reactome Cholesterol Biosynthesis genes, 25 occur between member genes ("within-community") while 157 involve other genes ("out-of-community"). In contrast, in Fig. 6B there are 143 out-of-community links and zero in-community ones. Fig. 6C shows how the numbers of within-community and out-of-community links change as the PCC significance threshold is changed, both for BigSur and uncorrected PCCs. These results suggest that the fraction of all correlations that is in-community can be used as a measure of the specificity with which functionally relevant correlations are identified by any given method. In Fig. 6D, we used this approach to contrast the performance of BigSur (filled symbols) with uncorrected PCCs (open symbols) for 8 different gene sets, each of which contained between 49 and 200 members. Using BigSur, the proportion of links that were in-community varied between ~1% and 30% depending on the gene set, usually increasing as the equivalent PCC threshold was made more stringent. Using uncorrected PCCs, the proportion of in-community links varied between 0.2% and 1% and did not change appreciably with PCC threshold. For comparison, Fig. 6E performs a similar analysis using 8 "random" gene sets—sets of 150 genes

randomly selected from genes with similar overall expression to those in the gene sets used in Fig. 6D. The results suggest that the performance of BigSur on functionally related genes is far above chance level, while uncorrected PCCs perform at or around that level.

2.3.8 Induced changes in gene expression are associated with increased gene-gene correlation

If BigSur detects physiologically meaningful gene correlations, one might expect genes induced in response to specific perturbations to be correlated with each other. In other words, among genes that differ *between* treated and untreated conditions, one might also expect to observe groups of the same genes displaying correlations *within* each individual condition—presumably because they represent genes regulated by common upstream signals. To test this, we turned to a scRNAseq study in which differentiated human airway epithelial cells were treated with IL-13, a model of cytokine-induced asthma [76]. We focused on a single subset of cells, those labeled "defense secretory", as similar numbers of them were captured in both treated and untreated conditions (see Methods).

Data from control and IL13-treated cells were then analyzed separately using BigSur (Figure S8/additional file 10). First, we focused on the 419 genes reported to be upregulated by IL13 [76]. Within the treated group we found 313 of these genes were also correlated with each other in a single, large community (Fig. S8A). Interestingly, 45 of these genes were correlated in control cells as well (Fig. S8B), suggesting that a module of co-regulated gene expression is already active prior to treatment.

Next, we asked whether BigSur could identify co-regulated genes that had not been picked up as differentially expressed (perhaps because their expression levels did not change sufficiently

after treatment). We focused on genes that correlated with one particular gene, MUC16, which encodes an IL13-induceable mucin thought to play a prominent role in asthma-associated mucus obstruction of the airway [76]. We found 168 genes to be significantly positively correlated with *MUC16* in IL13-treated cells (Figure S9/additional file 11), only 23 of which were among those differentially expressed in response to IL13 [76]. Among the remaining 145 genes were found *MUC4*, another mucin gene; *SYTL2*, a paralog of *SYT2*, which encodes a limiting factor in mucin secretion [77]; and *THSD7B*, which encodes a direct interactor of *SYT2* and *SYTL2*.

Also among the genes that correlated with *MUC16* in IL13-treated cells were 5 of 11 genes previously identified by GWAS as associated with risk of childhood asthma: *PDE4D*, *PTPRD*, *RBFOX1*, *ROBO2*, and *RYR2* [78]; *MUC16* also correlated with *ROBO1*, which encodes an interaction partner of *ROBO2*. The top-ranked GWAS gene, *RYR2*, encodes a calcium channel thought to play an important role in asthma pathogenesis [78]. Neither RYR2, nor any of the other 11 GWAS associated genes, were detected as differentially expressed after IL13 treatment, although two genes encoding proteins that interact with RYR2 (*CALM1* and *CMK2D*) were [76]. These data suggest that the analysis of gene correlations has the potential to substantially augment biological insights that come from studies of differential gene expression.

2.3.9 Aggregation into "meta-cells" impedes the identification of gene correlations.

The ability of BigSur to detect biologically relevant gene correlations with controllable false discovery reflects its ability to handle the highly skewed distributions of scRNAseq data, distributions that are usually dominated by many zeros. In the scRNAseq literature, several other approaches have been proposed to compensate for data sparsity, such as filling in zeros by imputation (estimating values based on closely related cells) or aggregating groups of cells

(merging measurements in similar cells, or across technical replicates of the same cells). Aggregation essentially replaces groups of cells with a smaller number of "metacells" or "pseudocells", essentially implementing a small-scale version of "pseudobulking" [79-83]. This approach has proved useful in improving the accuracy of identifying differentially expressed genes, and in analyses in which genes are clustered into weighted gene co-expression networks [84, 85].

To the extent that aggregation reduces data sparsity, one might expect it also to improve accuracy in discovering gene-gene correlations. In fact, at least one group [86] has proposed this explicitly, developing a method that starts with normalized, log-transformed scRNAseq data, and averages values among groups of cells that are judged sufficiently similar (based on Leiden clustering). Using this method, they claimed that one can enhance the ability to detect meaningful correlations in scRNAseq data.

We investigated this claim, however, and came to the opposite conclusion. We discovered that grouping cells into metacells markedly inflates false positive correlations, a phenomenon not explored by [86]. This effect is easily demonstrated by applying their procedure to synthetic data in which gene expression values were sampled randomly and independently from Poisson-lognormal distributions (coefficient of variation of underlying lognormal distribution = 0.5), with cells sequenced to different depths, as in Fig. 1-2. Using such data as a starting point, we calculated correlation coefficients six different ways (Figure 7A-C), using the Fischer formula to define the threshold for statistical significance (with PCC′ we also used a Benjamini-Hochberg adjusted FDR threshold of < 2%, as determined by BigSur). In such a data set, an accurate method should detect *no* significant correlations, positive or negative.

As expected, using PCCs calculated directly from raw, unnormalized data, nearly 100 million artifactually significant positive correlations were identified, and these were not eliminated by data normalization (scaling UMI values to sequencing depth). Log-transformation of the data (using a "pseudocount" of 1) reduced the number of false positives to 382,511, probably because it greatly reduced the range of variability between high and low values. In contrast, when log-transformed values were grouped and averaged according to the procedure of [86], clustering the 3,570 cells and grouping them into 100 metacells, we observed a massive inflation of large positive and negative values of PCC (Fig. 7B-C), among which over 2 million were judged significant by the Fischer formula (the formula automatically takes into account the reduced number of cells). This was not just a consequence of failure to compensate for variable sequencing depth, because even when we created data with equal sequencing depth across all cells, we still observed that grouping into metacells produced just as many false correlations (Fig. 7B-C).

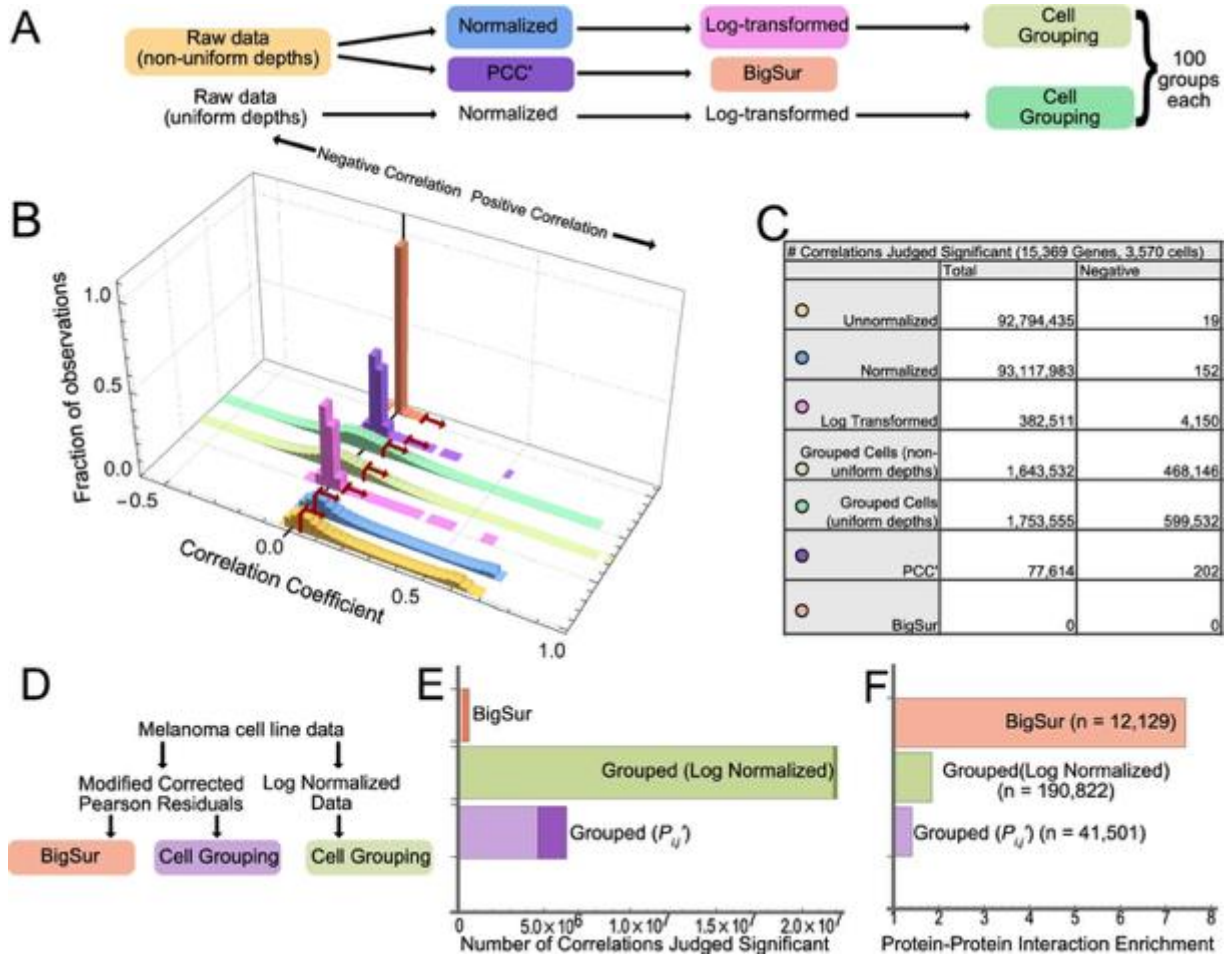


Figure 7. Grouping into "metacells" creates false positive correlations.
A-C. Analysis of synthetic, uncorrelated data. A. Pipelines for data processing. Each box denotes a step at which a Pearson correlation coefficient (PCC or PCC') was calculated, with the color of the box corresponding to the colors used in the following plots. B. Histograms of correlation coefficients obtained from the data sets in panel A. Arrows show the thresholds above which observed correlations were judged statistically significant. C. Numbers of correlations in panel B that were judged to be significant ($p < 0.02$). D-F. Analysis of melanoma cell line data. D. Pipeline for data processing. The colors of each box correspond to the colors in panels E-F. E. Number of correlations judged significant in data sets in D. Darker shading denotes the negative correlations; lighter are positive correlations. F. Enrichment for known protein-protein interactions among the correlations shown in E. The value of n in each case gives the absolute number of protein-protein interactions. Of the 12,129 interactions detected by BigSur, 11,373 were also detected using grouping of log-normalized data and 10,773 were detected using grouping of modified corrected Pearson residuals (10,324 were shared among all three).

In contrast, PCC′, calculated from unnormalized data, did much better, producing fewer than 78,000 false correlations, as judged using the Fischer formula, and no correlations, positive or negative, when the FDR was appropriately calculated using BigSur.

These results indicate that, even if grouping cells increased the likelihood of detecting true correlations, the effect would likely be overwhelmed by the creation of so many false ones. To assess the net effect, we compared the method of [86] with BigSur using the full set of melanoma cells analyzed in Fig. 3. As diagramed in Fig. 7D, we performed grouping both on Log-transformed normalized data, as per [86], and on modified corrected Pearson residuals (P_{ij}'), hoping that the latter might better correct for variable sequencing depth. Consistent with what was seen with synthetic data, grouping produced many more correlations than BigSur, although the use of modified corrected Pearson residuals reduced this to some extent (Fig. 7E).

To assess the recovery of true positives, we quantified enrichment for pairs of genes with known protein-protein interactions (Fig. 7F), as such genes are more likely than random genes to be co-expressed. As described earlier, such interactions were enriched more than seven-fold among the pairs of genes found significantly positively correlated by BigSur. In contrast, even though grouping into metacells identified significant correlations among more total gene pairs with known protein-protein interactions, the enrichment over what would have been expected by chance was less than two-fold, suggesting an accuracy barely better than chance levels.

The reason grouping generates so many false positive correlations appears to be generic to the process of aggregating cells by similarity, and not specific to details of the method of grouping of [86], as we can see similar effects using synthetic data and other, simpler algorithms for grouping cells. These observations nicely illustrate how, when dealing with correlations, even seemingly innocuous empirical fixes can create unexpected problems, and underscore the value of BigSur in provided a principled, theory-grounded approach to assessing both the magnitude and significance of gene-gene correlation.

2.4 Discussion

The above results suggest that *bona fide* communities of co-regulated genes can be identified with high specificity by carefully mining weak correlations within groups of a thousand or fewer relatively homogeneous cells. BigSur achieves the accuracy to do this first by correcting measures of correlation for unequal sequencing depth and the added variance contributed by gene expression noise, and subsequently by estimating an individual *p*-value for each gene pair—thereby overcoming the strong effect of gene expression distribution on the likelihood of correlation arising by chance. In contrast, the use of uncorrected PCCs to identify co-regulated genes performed poorly on both synthetic and real scRNAseq data, as did at least one method of compensating for data sparsity by cell aggregation.

In developing BigSur, we sought to avoid normalization steps and expression thresholds and to minimize user-provided parameters to the greatest extent possible. The major user input to BigSur, besides a UMI matrix, is a coefficient of variation for gene expression noise, c, which can be quickly estimated by fitting a plot of the modified corrected Fano factor against gene expression level (Fig. 2B). In reality, the magnitude of the noise of gene expression may be different for different genes, and for some the Poisson log-normal distribution may not be the best approximation of the noise. Although these factors likely degrade the performance of BigSur, we note that the value of c only significantly impacts the highly expressed genes, for which (due to low sparsity) the detection of significant correlations is a less challenging task.

A more subtle source of potential error comes from fact that, in calculating modified corrected Pearson residuals, the value of μ_{ij} used by BigSur is determined empirically from a finite set of cells, i.e., it is an estimator of μ_{ij}. Furthermore, whereas it is an unbiased estimator when μ is Poisson-distributed, this is not generally the case for more skewed distributions, such as Poisson-

lognormal [43]. It is unknown whether these sources of error have much impact on the determination of correlation coefficient *p*-values by BigSur, and additional work will be necessary to investigate this question.

Despite these concerns, the ability of BigSur to identify gene communities that are closely related in function (Table 1), as well as add new, functionally related genes into known gene sets (Fig. 6, S9), suggests that it already operates at a level of performance that can be useful to cell and tissue biologists. Particularly interesting are the questions it raises about linkages within communities—for example, why do genes encoding cystatin endopeptidase inhibitors (*CST1*, *CST4*, *CSTB*) correlate with genes involved in glycolysis and cellular respiration (community E)? Why do genes encoding protein chaperones correlate with genes involved in RNA processing (community G)? Why do genes involved in Golgi and lysosome function correlate with genes related to hypoxia (community J)?

The data in Table 1 also suggest that new cell biology may be discovered by examining genes labeled "not accounted for", i.e., genes that are associated with a community but do not, as a group, overlap substantially with any mSigDB dataset. For example, 38 of the 213 genes that correlate with cell cycle community B are not currently annotated as cell-cycle related. Among the strongly-coupled unfolded-protein response genes in community C, one also finds genes encoding secreted (*GDF15, PSAP, SPARC*) and cell surface molecules (*DDR2*, *GPNMB*, *ITGA4, IL1RAP, PMP22, SLC1A5*); the potential relationship of these genes to cellular stress responses may deserve heightened attention. Indeed, each of the communities in Table 1 suggests new and unexpected forms of co-regulation of gene expression. It will be interesting to see how many of these are reproduced in other cell types—a task that can be efficiently approached by applying BigSur to the large number of existing scRNAseq data sets.

It is instructive to note that few of the gene-gene relationships detected by BigSur could have been revealed by the typical analytical steps of cell clustering and identification of differentially expressed genes, as most of the gene communities identified here are not associated with sufficient total variation in gene expression to be useful drivers of cell clustering. On the other hand, clustering did play an important role here in reducing the heterogeneity of the sample to which BigSur was applied. Although most of the gene communities that were identified using the 1,582-cell subcluster 1.2 were also observed when BigSur was applied to the entire sample of 8,640 cells (not shown), clusters were more difficult to visualize and analyze in the latter case, thanks to a large background of cell-type (or cell-state)-specific gene expression (which generated numerous additional correlations). This experience suggests that iterative application of BigSur analysis and clustering (potentially using correlated gene communities as features) can provide a useful pipeline for identifying meaningful gene communities of manageable size.

It is interesting that one of the strongest axes of variation we detected in this study involved mitochondrially-encoded genes and genes coding for ribosomal subunits, with both communities strongly anti-correlating with each other (both before and after "damaged" cell removal and subclustering). Because the mitochondrial community consists only of mitochondrially-encoded, and not cytoplasmically-encoded, mitochondrial genes, the simplest interpretation is that this community reflects cell-to-cell variation in the number of mitochondria (or, more accurately, mitochondrial genomes). What then might explain anti-correlation between mitochondrial number and transcripts for ribosomal proteins? Given that protein synthesis requires both specialized machinery (ribosomes) and a source of energy (mitochondria), one might expect to see positive, rather than negative, correlation between the agents that orchestrate these processes. Yet this intuition is correct only in a long-time-averaged sense and does not necessarily apply if demand

for protein synthesis fluctuates. In mammalian cells, ribosomal protein mRNAs are long-lived, with half-lives in the 5-10 hour range [87], whereas mitochondria can replicate on a time scale of 1-2 hours [88]; accordingly, one process may systematically lag the other, producing the kind of anti-correlation we observe here. While this interpretation is speculative, it demonstrates how the analysis of gene-gene correlations can motivate new hypotheses about transcriptome-scale regulation of cell biology.

Under the expectation that most networks of gene correlation reflect shared transcriptional regulation, we might have expected to identify upstream transcription factors more frequently in most of the networks we discovered. In some cases, this clearly did occur: For example, the transcription factors *ATF4*, *AT6*, and *XBP1*, which were detected in community C, are known controllers of the unfolded protein response, the components of which show up in the same community. Transcription factors related to cell cycle progression and DNA damage-repair strongly associated with cell-cycle communities A and B. *ZKSCAN1*, a transcription factor targeted to mitochondria [89] associates with the mitochondrially-encoded gene community H. However, in many cases, expected transcription factors (e.g., *SREBF1* or *SREBF2* in community K) were not observed. This may reflect limitations in statistical power, as transcription factor genes tend to be expressed at a somewhat lower level than other genes—although in this data set expression of the average observed transcription factor was only about half that of the average observed gene. Another likely explanation is that gene regulation is often achieved through the post-translational modification of transcription factors, rather than regulation of their mRNAs. Alternatively, it may reflect the importance of factors that act post-transcriptionally, such as miRNAs (which were not assessed in this data set), in controlling gene expression. In future, it

will be important to extend BigSur to take account not only of miRNAs but also of "multi-omic" features, such as chromatin accessibility.

Although we have focused here primarily on the use of BigSur as a tool for discovery of gene-gene correlations, it is worth pointing out that the intermediate steps in the BigSur pipeline produce useful tools for other analytical procedures, some of which were exploited here. For example, methods for feature selection for cell clustering commonly involve thresholds (e.g., expression levels) and cutoffs (e.g., numbers of features) that are arbitrary, and may not be ideal choices for every data set. Use of the modified corrected Fano factor ϕ', and its associated *p*-values, can provide a less arbitrary approach to feature selection, which can outperform other methods on challenging tasks, such as finding rare cell types, or subclustering cells that differ only modestly [90]. In addition, as we did here when dividing cells into subclusters based on the expression of ribosomal and mitochondrial genes (Fig. 4), one may also use communities of correlated genes themselves as features for clustering—in this way leveraging not just variation but co-variation to drive clustering. Finally, whereas it is common practice to cluster cells based on their normalized expression values, the matrix of modified corrected Pearson residuals that BigSur calculates almost certainly provides a more accurate starting point for clustering, as it avoids artifacts introduced by normalization, and corrects for the inflated variance associated with highly expressed genes.

2.5 Conclusions

It has long been suspected that, by mining gene expression correlations within single cell transcriptomes, it should be possible not only to identify groups of genes that distinguish among cell types, but also discover small-scale gene regulatory networks that operate within cell types.

We have provided evidence here that this goal can be achieved, first by modifying and correcting the metric of correlation and then using an analytical approach to assign statistical significance to every gene pair. When applied to scRNAseq data, this dramatically reduced the number of false positives that would have been identified by other methods, and enabled the identification of biologically relevant gene communities, both known and novel.

2.6 Methods

As the first step in calculating ϕ' and PCC′, we begin with "raw" (neither normalized nor log-transformed) UMI data and calculate, for each gene in each cell, a cell- and gene-specific Pearson Residual, defined according to eq. 1. To do so, we first calculate a cell- and gene-specific expected value (μ_{ij}) which we obtain for each gene by averaging its values over all cells, then scaling that in each cell by to the relative proportion of total UMI that each cell contains. This is essentially the same procedure used in the simplest form of normalization, except that, rather than normalize the data matrix, we are normalizing the term for the mean in the Pearson residual.

Next, each modified Pearson residual is divided by $\sqrt{(1 + c^2\mu_{ij})}$, where c is a constant between 0.2 and 0.6. Ideally, c should be chosen individually for each gene, depending on prior knowledge of the level of gene expression stochasticity, however, in the absence of prior knowledge we typically find c empirically (see Fig. 2B). It should be noted that, for many scRNAseq data sets, most values of μ_{ij} will be < 1, meaning the effect of the choice of c on most of the data is often relatively small. To obtain ϕ' for each gene, the Pearson residuals for each cell are squared, summed, and divided by n-1, where n is the number of cells.

To calculate *p*-values for ϕ' any given gene, we consider the null hypothesis to be the expected number of transcripts in each cell is μ_{ij}, i.e., the total number of transcripts across all the cells partitioned in proportion to the number of total UMI in each cell. As noted above, because BigSur determines the value of μ_{ij} empirically—by summing up genes UMI across all cells and multiplying by the fraction of total UMI in each cell—μ_{ij} is technically an estimator of the cell-specific expectation value for each gene and cell.

To calculate p we need to know how the sums of squared Pearson residuals should be distributed for any given set of μ_{ij} and c. As discussed above, we take μ_{ij} to have a Poisson-log-normal distribution, allowing us to calculate the moments of the distribution of squared Pearson residuals from the moments of the Poisson and log-normal distributions, and from there the moments of the distribution of sums of squared Pearson residuals. In the end we obtain, for each gene *j*, a finite set of moments (typically 4 or 5) for the distribution of ϕ' that would be expected under the null hypothesis for that particular gene, given the values $\mu_{1j}, \mu_{2j}, \mu_{3j} \ldots \mu_{nj}$ and *c*. We then use Cornish-Fisher approximation of the Edgeworth expansion [91] as a reasonably computationally efficient way to approximate the *p*-value associated with any given observation of ϕ', given that distribution.

The procedure for obtaining PCC′ proceeds in the same fashion, starting with the same modified corrected Pearson residuals, but now taking the dot product of the vectors of Pearson residuals for each pair of genes, and dividing by *n*-1 times the geometric mean of the ϕ'values for those genes (eq. 2). Moments of the expected distributions of PCC′ are calculated analytically in exactly the manner described above, with the slight complication that the ϕ'terms in the denominator are not strictly independent of the Pearson residuals in the numerator, but to a good first approximation may be treated as such, as they aggregate information across all the Pearson

residuals. The Cornish-Fisher approximation is again used, as described above, to assign *p*-values to both tails of the resulting distribution. In solving the 4th degree polynomial equations produced by this method (a computationally slow step), we improve speed by sacrificing accuracy specifically for those *p*-values that clearly fall below thresholds of interest.

Because the number of possible gene-gene correlations scales roughly as the square of the number of genes, the need to correct for multiple hypothesis testing is particularly acute. Given that the observations are not independent from one another, Bonferroni correction is clearly too conservative, and thus we use the Benjamini-Hochberg algorithm [49] for controlling the false discovery rate.

2.6.1 Simulating scRNAseq data

In Figure 2, we simulate the expression of 1000 genes across 999 equivalent cells, where by equivalent we mean that, for each gene, a single "target" transcript level was chosen from a log-normal distribution, the mean of which was selected so that the logarithm of its value varied uniformly across the gene set, over the range from 0.035 to 3467 transcripts per cell. The coefficient of variation of each log-normal distribution was taken to be 0.5 for all genes. Next, we generated a set of 999 scaling factors, drawn from a log-normal distribution with mean of 1 and coefficient of variation of 0.75, selected so that the logarithm of the value varied vary smoothly across the range. Finally, we generated a UMI value to each gene in each cell that was a random variate from a Poisson distribution with a mean equal to the target for that gene times the scaling factor for that cell. The result was a set of gene expression vectors of length 999, with mean values varying between 0.001 and 231.

In Figure 7A-C, synthetic data were generated as in Figure 2, except that the number of genes was increased to 15,369 and the number of cells increased to 3,570. Grouping cells into 100

metacells was carried out as described by [86]. In Figure 7D-F, the full melanoma cell dataset was used (17,451 genes x 8,640 cells; see below), and grouping was again performed to create 100 metacells.

2.6.2 Analysis of melanoma cell line data

Data from droplet-based sequencing of subcloned WM989 melanoma cells (GEO accession GSE99330), which had been pre-processed to remove UMI judged not to be associated with true cells, were imported and further pre-processed in the following way: First we removed all known pseudogenes (comprehensive lists of human pseudogenes were obtained from HGNC and BioMART). Pseudogenes derived by gene duplication or retrotransposition are often highly homologous to their parent genes, creating ambiguity in the mapping of the short-read sequences used in scRNAseq. When pseudogenes were not removed from analysis, we frequently detected strong correlations between pseudogenes and parent genes that very likely represented the effects of ambiguous mapping, rather than true correlation. After pseudogene removal, the number of detected genes was 27,526. As both theory (Fig. 1) and experience indicated that statistically significant correlations were usually unobservable for genes with UMI in fewer than 0.001% of cells, we further eliminated genes expressed in fewer than 8 of the 8640 cells; this reduced the size of the gene set to 17,451, and the number of possible unique correlations to 152,259,975 (while not strictly necessary, this step reduces computational time by ~2.5 fold, since the number of correlations to test varies approximately quadratically with the number of genes).

2.6.3 Analysis of cultured airway epithelial cell data

Data from a study comparing untreated and acutely IL13-treated cultured human airway epithelial cells were downloaded from GSE145013 [76]. To confine analysis of correlations to a relatively homogeneous cell type, we subsetted only those cells clustered into groups composed

primarily of the "secretory" cell type; specifically, related clusters c5 and c6 contained most of the secretory cells of the treated and untreated samples, respectively, and analysis was confined to cells of these clusters. This resulted in a dataset of 407 IL13-treated 303 untreated cells. After removal of pseudogenes, and gene with fewer than 80 total UMI, each dataset contained the same set of 15,268 genes.

2.6.4 Calculating paralog pair and protein-interaction enrichment scores.

A curated list of 3,132 paralogous pairs of human genes was downloaded from [92]. A list of physical human protein-protein interactions was downloaded from BioGrid (https://downloads.thebiogrid.org/File/BioGRID/Release-Archive/BIOGRID-4.4.218/BIOGRID-MV-Physical-4.4.218.tab3.zip) and supplemented with additional data from HIPPIE v2.3 (http://cbdm-01.zdv.uni-mainz.de/~mschaefer/hippie/), to produce a list of 790,008 gene pairs. To calculate enrichment, we first calculated the fraction of statistically significantly positively correlated gene pairs identified by BigSur that overlapped with either the paralog-pair or protein-protein interaction pair database. Next, we removed from the databases all gene pairs involving genes not detected in the scRNAseq data and divided the remaining number by the total number of possible gene-gene correlations (i.e. $m(m\text{-}1)/2$, where m is the number of genes in the scRNAseq data) to yield the expected frequency of paralogous or interacting pairs under the hypothesis they are randomly distributed among all possible pairwise correlations. The ratio between the observed frequency and expected frequency was considered to be the fold enrichment.

2.6.5 Extracting (and pooling) gene communities

BigSur generates a matrix in which rows and columns are genes, and entries are signed equivalent PCCs—which are derived by using the inverse of the Fisher formula on the p-values returned by BigSur, together with the signs of the values of PCC′. Although it is a derived quantity,

the equivalent PCC is a useful form in which to store correlation data, not only because it is signed, but also because it adjusts for differences in data length (number of cells), so that similar "strengths" of correlation would be expected to translate into similar equivalent PCCs, even across samples with very different numbers of cells (cell number has a large effect on the relationship between correlation strength and *p*-value).

Only equivalent PCCs that were judged statistically significant according to a user-supplied threshold (e.g., a Benjamini-Hochberg FDR) were included, all others being set to zero. This matrix was converted to an unweighted adjacency matrix (all nonzero values replaced with 1) and the *walktrap* algorithm (with a default setting of 4 steps) was used to identify initial communities [52]. Because this produces communities connected by both positive and negative links, each community was then subjected to a second round of community-finding, after first setting negative links to 0, thus allowing subcommunities that negatively correlate with each other to be separated.

To identify instances in which walktrap had subdivided communities too finely, we manually examined the number of positive correlations between genes in each community and each other community, recursively merging communities in which the number of inter-community correlations was particularly large (compared with the number of possible links between communities). In addition, in rare cases in which communities returned were very large (e.g., in the thousands), we subdivided them by applying walktrap an additional time.

2.6.6 Cell clustering based on correlated features

Feature selection refers to the process of identifying genes that capture important dimensions of variation on which cells may be clustered. A variety of approaches have been proposed for identifying such genes, and many work equivalently under most circumstances (with tens of thousands of genes, clustering is often a highly over-determined problem). Under challenging

circumstances (e.g. when the number of true difference separating clusters is small, or the number of cells in a state is small), we have shown [90] that ϕ' is a measure of variability at least as good as others, and because BigSur returns both ϕ' and *p*-values, one may avoid selecting too large a set of features (which can defeat clustering algorithms).

It has also been pointed out, however, that not only the statistical features of individual genes, but also their interdependencies (i.e., correlations) should ideally be used to inform clustering [24]. We recognized that the communities identified by BigSur represent ideal sources of features, particularly if we emphasize those community members that are the most highly connected to each other. We also recognized that the modified corrected Pearson residuals generated by BigSur provide a more sensible set of vectors to use as the input to clustering than either raw or normalized UMIs (for the same reasons that ϕ' and PCC$'$ are improvements over their unmodified, uncorrected forms). Briefly, we used the modified corrected Pearson residuals to construct shared nearest neighbor graphs based on the top 50 principal components, with k=20 and a Jaccard similarity threshold of 1/15; this was followed by Leiden clustering [93] and UMAP visualization, essentially as in the Seurat analysis package [94].

With this approach, we found that well-defined clusters can often be reliably obtained using sets of as few as 50-75 highly connected genes. This was used to repetitively subcluster the scRNAseq data from WM989 melanoma cells, at each step running BigSur and using the 50 most connected genes of the clusters containing the majority of ribosomal genes, together with the 50 most connected genes of those containing the majority of mitochondrially encoded genes, as features.

2.6.7 Graphical display of correlations

Matrices representing statistically significant correlations were plotted using the *GraphPlot* function of Mathematica software, in which vertices were arrayed either by Spring Embedding or

Spring Electrical Embedding. Edges were colored green when positive and red when negative. Vertex locations were first determined according to the graph produced after deleting negative edges, after which vertices connected only by negative edges were added in. Symbols used to represent vertices were scaled so that their areas were proportional to the mean expression level of the gene represented. Correlation strengths (the absolute values of the equivalent PCCs) are not represented on these images.

2.7 Code Availability

R and Mathematica implementations are available under the Lander lab profile on GitHub (https://github.com/landerlabcode/).

2.8 Abbreviations

BigSur, Basic Informatics and Gene Statistics from Unnormalized Reads; CV, Coefficient of variation; ER, Endoplasmic reticulum; FDR, False discovery rate; PCA, Principal component analysis; PCC, Pearson correlation coefficient; PCC', Modified, corrected Pearson correlation Coefficient; ROC, Receiver operating characteristic; scRNAseq, Single cell RNA sequencing; UMAP, Uniform manifold approximation and projection; UMI, Unique molecular identifier.

2.9 Declarations

Ethics approval and consent to participate: Not applicable.

Consent for publication: Not applicable.

Availability of data and materials: The datasets generated and code used for analysis during the current study are available in the BigSurM Github repository,

https://github.com/landerlabcode/BigSurM/tree/0c128246bed1969b371e12bdbd094e49436e2e34/Datasets

Competing interests: The authors declare that they have no competing interests

Funding: This work was supported by NIH grants CA217378, AR075047, DE019638 and the NSF-Simons Center for Multiscale Cell Fate Research (NSF 1763272). K.S. and E.D. acknowledge support from NIH training grant GM136624.

Author contributions: KS conceived of the work, developed and implemented code, was responsible for posting code and data to github, and contributed to writing of the manuscript. ED conceived of the work and contributed to writing of the manuscript. JG helped conceive of the study and contributed to the development of code. SA provided financial support and edited the manuscript. QN provided financial support, advice on code development and edited the manuscript. AL conceived of the work, developed and implemented code, and contributed to writing of the manuscript. All authors read and approved the final manuscript.

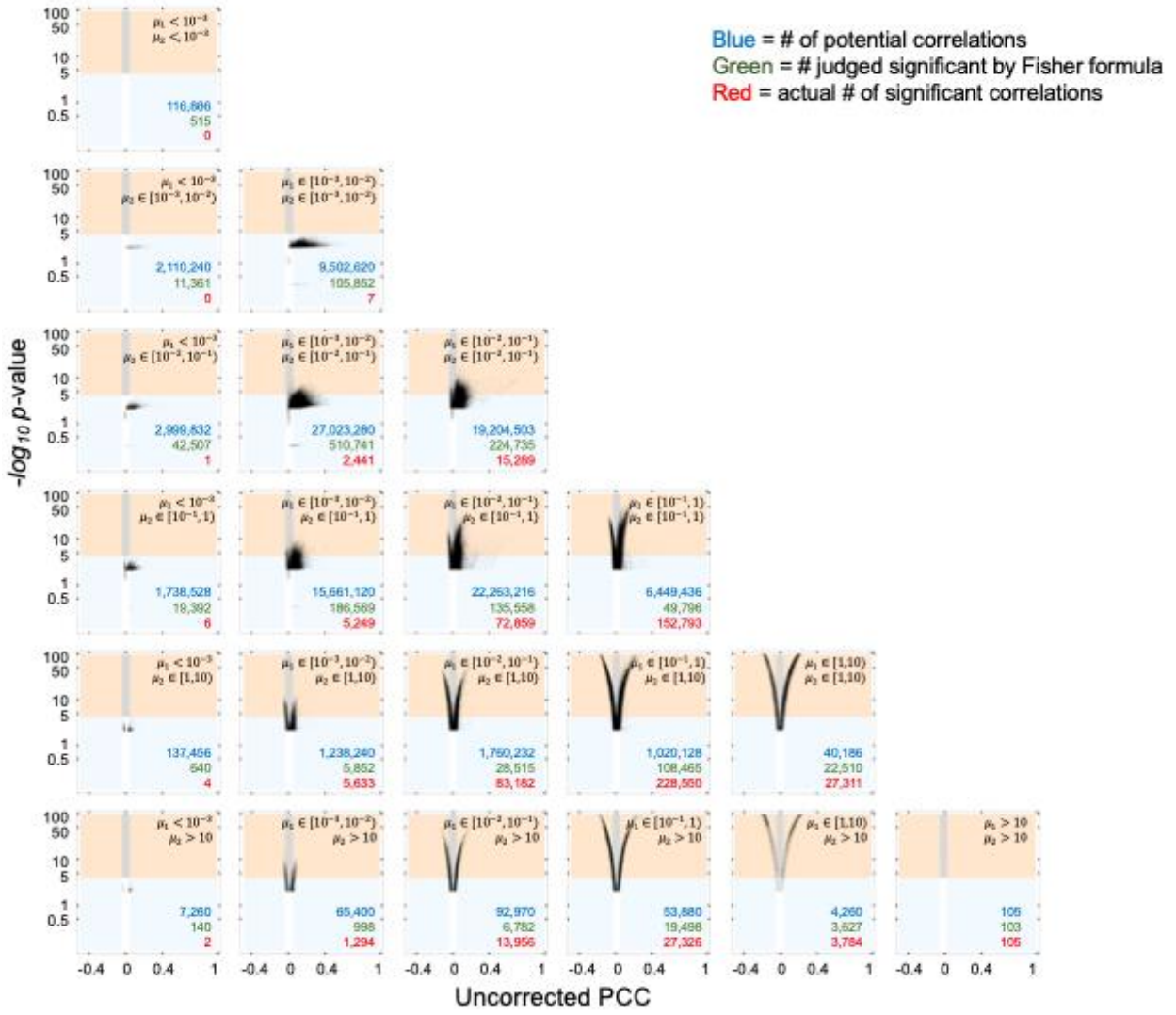
Acknowledgements: We thank Weining Shen (UCI) for advice on statistical analysis. We acknowledge Mika Caldwell and Yilun Zhu for helpful feedback and testing of code.

2.10 Description of Supplementary Files

Additional File 1: Supplementary Table 1. Analysis of gene communities using the Database for Annotation, Visualization and Integrated Discovery (DAVID)**.** Gene communities in Table 1 were analyzed using DAVID functional analysis [58] and the following databases: UP_KW_BIOLOGICAL_PROCESS, UP_KW_CELLULAR_COMPONENT, UP_KW_MOLECULAR_FUNCTION, UP_KW_PTM, UP_SEQ_FEATURE, GOTERM_BP_DIRECT, GOTERM_CC_DIRECT, GOTERM_MF_DIRECT, BBID,

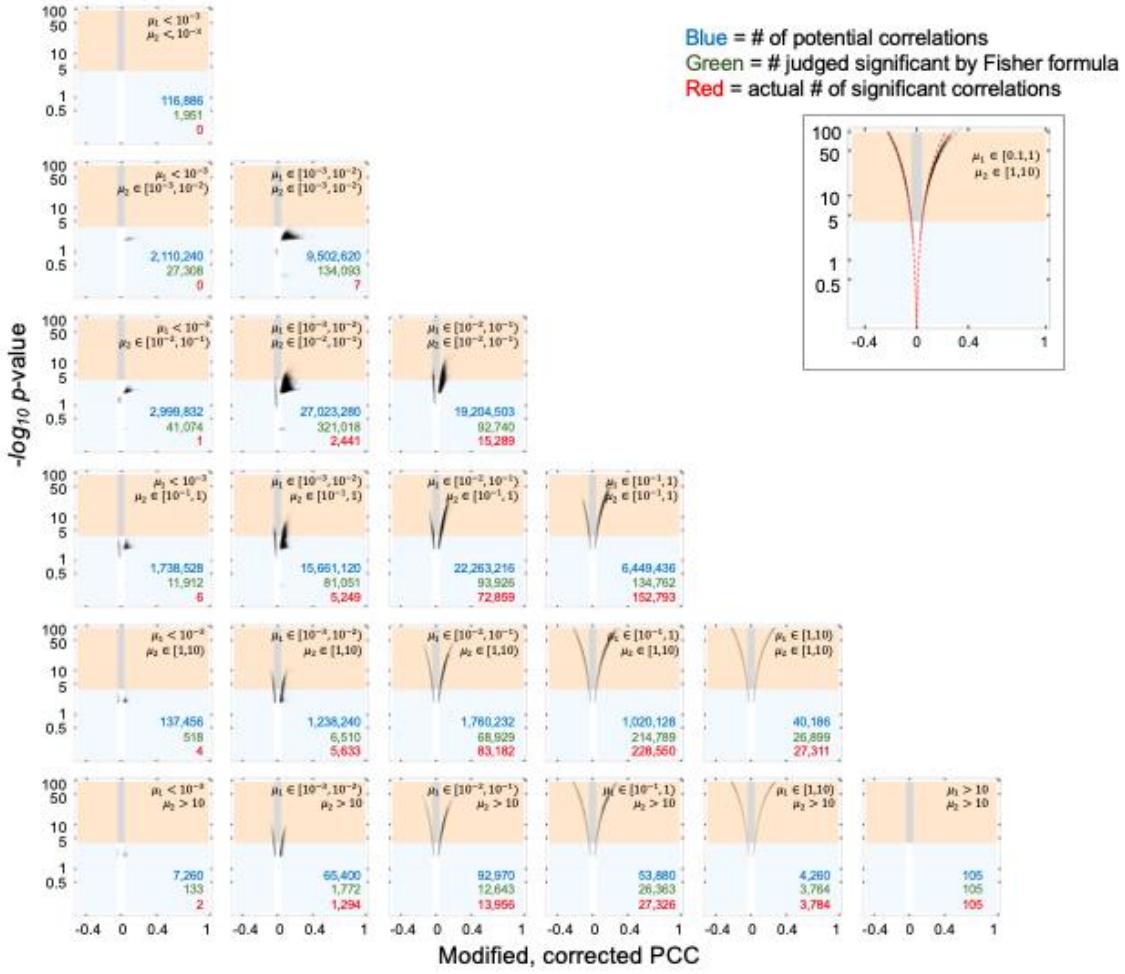
BIOCARTA, and KEGG_PATHWAY. Results shown are those that met default threshold requirements to display in the Functional Annotation Chart (at least 2 genes in each category and an EASE score of 0.1 or lower). Results for each community are given on a separate spreadsheet. Additional files are found with the original published manuscript.

2.11 Supplementary Figures

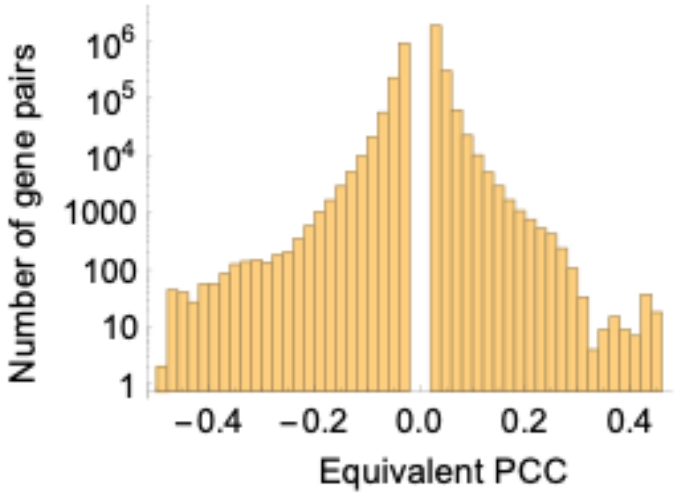


Supplementary Figure 1. Significance of uncorrected Pearson correlation coefficients (PCC), as calculated by BigSur versus the Fisher formula, binned by gene expression. scRNAseq data were as described in Figure 3. Data points representing pairs of genes were divided into 21 bins based on the mean expression levels of each gene, and the results for each bin were plotted as described in Fig. 3C. The abscissa shows PCC (calculated from default-normalized data) while the ordinate gives the negative log10 of p-values

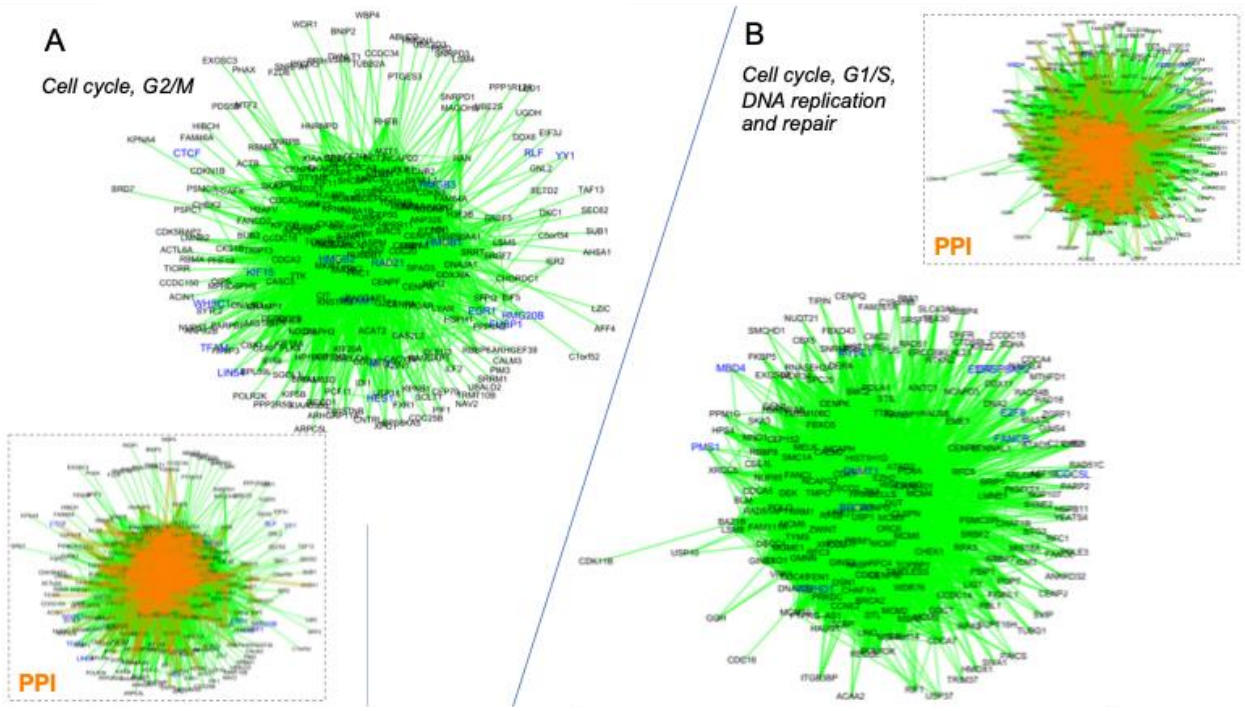
determined by BigSur, i.e., larger values mean greater statistical significance. Orange and gray shading indicate gene pairs judged significant by BigSur (FDR<0.02). Blue and orange show gene pairs that would have been judged statistically significant by applying the Fisher formula to the PCC, using the same p-value threshold as used by BigSur. The blue region contains gene pairs judged significant by the Fisher formula only, while the unshaded region shows gene pairs not significant by either method. Numbers in the lower right corner of each panel are the total numbers of possible correlations (blue), statistically significant correlations according to the Fisher formula (green), and statistically significant correlations according to BigSur (red).



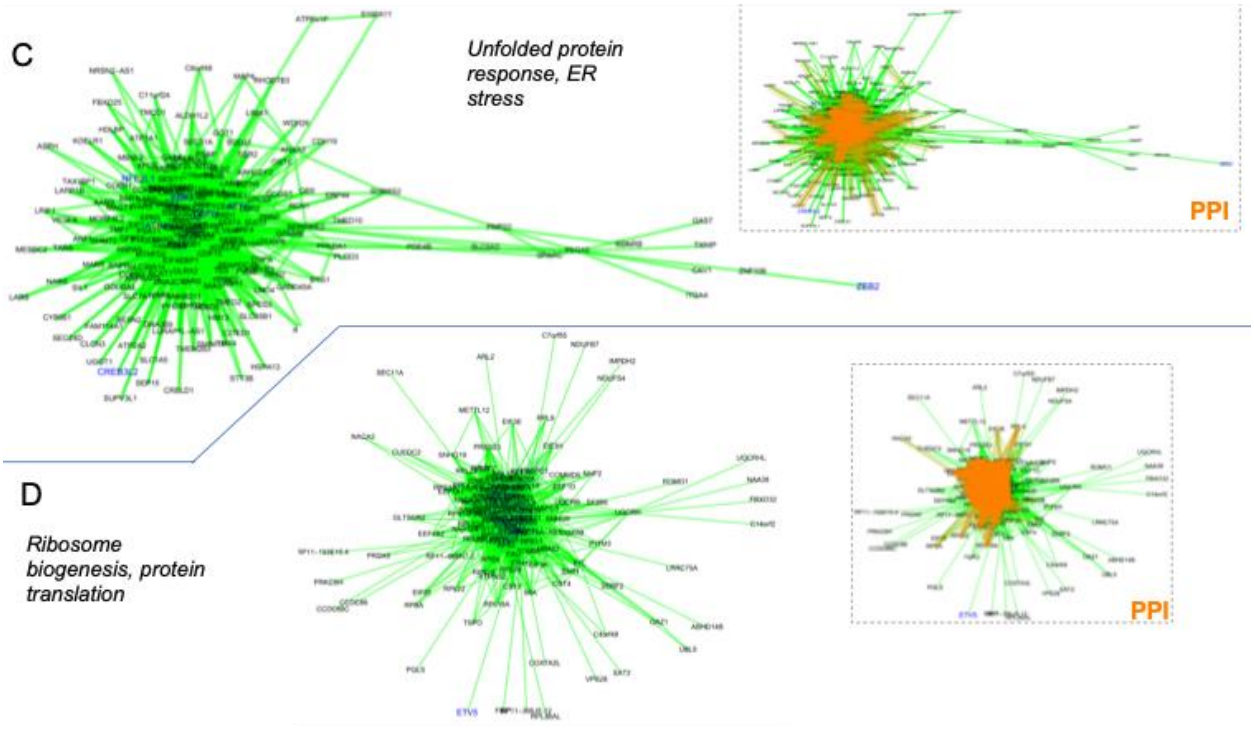
Supplementary Figure 2. Significance of modified corrected Pearson correlation coefficients, (PCC'), as calculated by BigSur versus the Fisher formula, binned by gene expression. scRNAseq data were as described in Figure 3. Data points representing pairs of genes were divided into 21 bins based on the mean expression levels of each gene, and the results for each bin were plotted as in Fig. S1. The inset compares the PCC'-p-value relationship determined by BigSur (for genes with relatively high expression levels) with that predicted by the Fisher formula (dashed red line), showing that, for highly expressed genes, the two methods agree well.



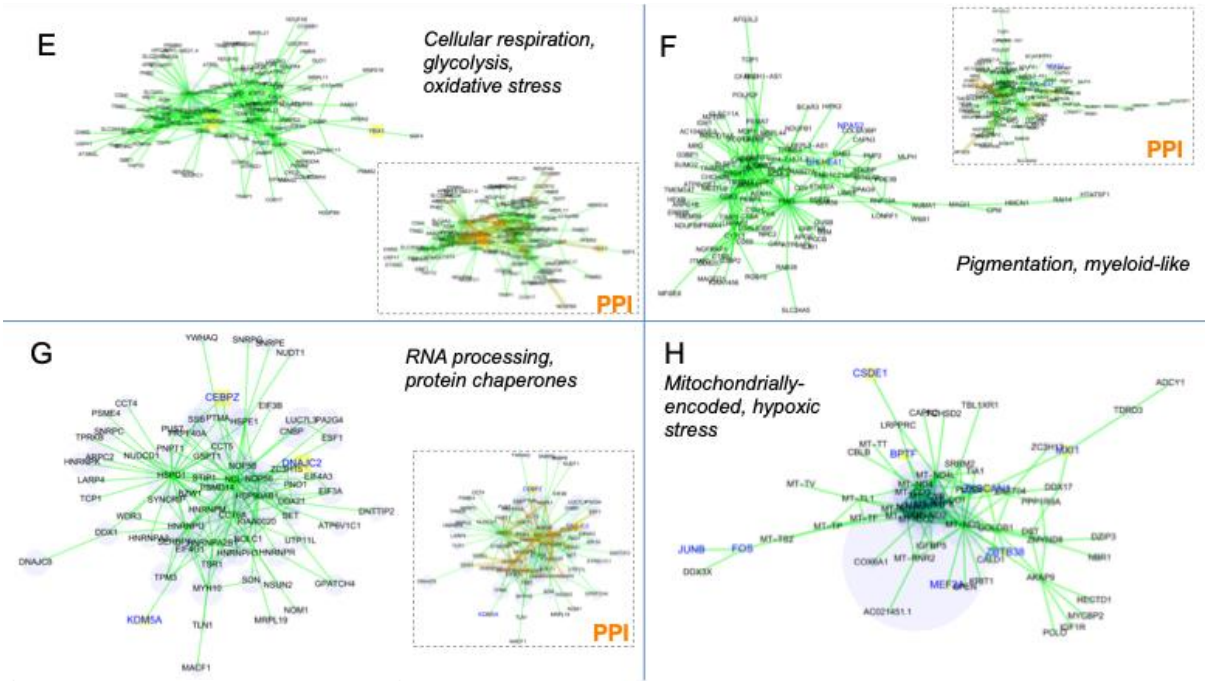
Supplementary Figure 3. Distribution of equivalent PCCs. The p-values obtained by BigSur for the melanoma cell line were transformed using the inverse of the Fisher formula to a set of "equivalent" PCCs. Since the Fisher formula operates on the absolute values of correlations, each calculated equivalent PCC was assigned the sign of the PCC' value for the same gene pair. Equivalent PCCs provide a measure of correlation strength that can be compared across data sets with differing numbers of cells. They may be understood as a measure of how strongly correlated two normally distributed vectors (of any given length) would need to be to produce the observed p-value. Here, only those gene pairs judged significant by BigSur are shown. The fact that so many weakly correlated gene pairs (|equivalent PCC| <0.05) are nevertheless statistically significant is a function of the long vector length in this experiment (>8000 cells).



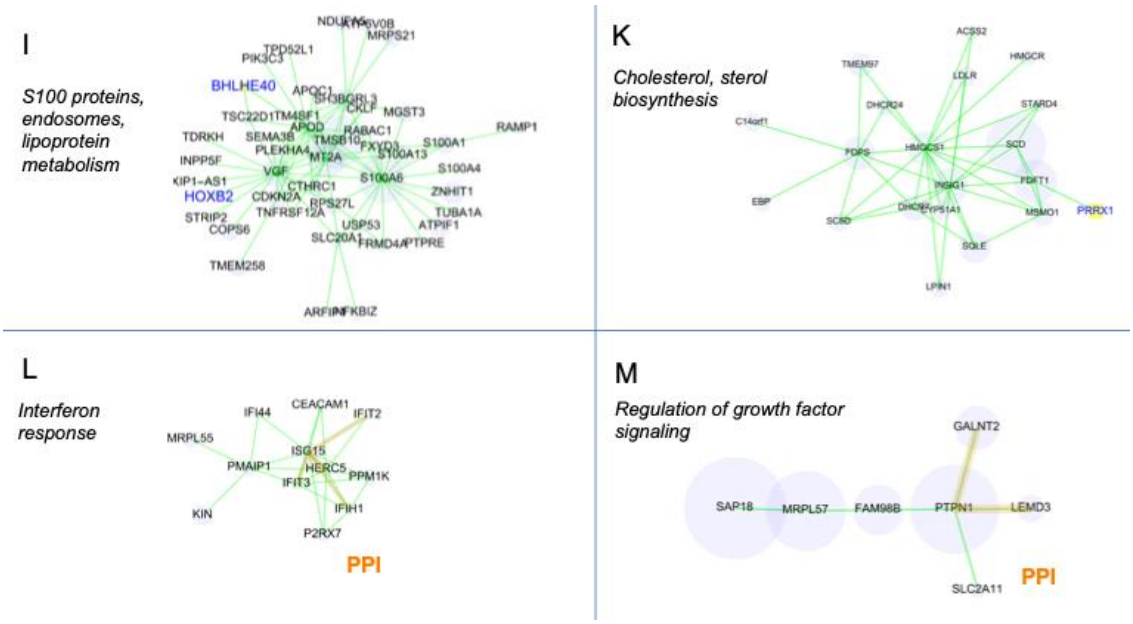
Supplementary Figure 4. Gene communities A and B (see Table 1) from cell cluster 1.2. Green edges depict significant correlations. Transcription factor vertices are displayed as yellow boxes with gene names in blue. In the boxed insets the same graphs are overlayed in brown to highlight links supported by known protein-protein interactions.



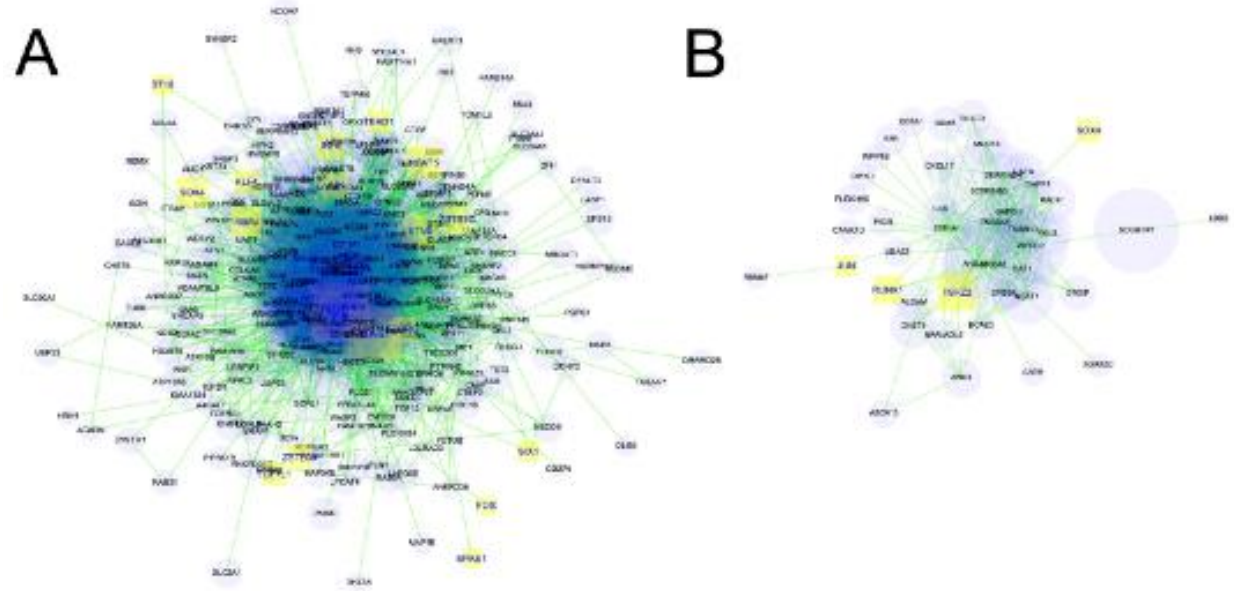
Supplementary Figure 5. Gene communities C, and D (see Table 1) from cell cluster 1.2. Genes and links are highlighted as in Fig. S4.



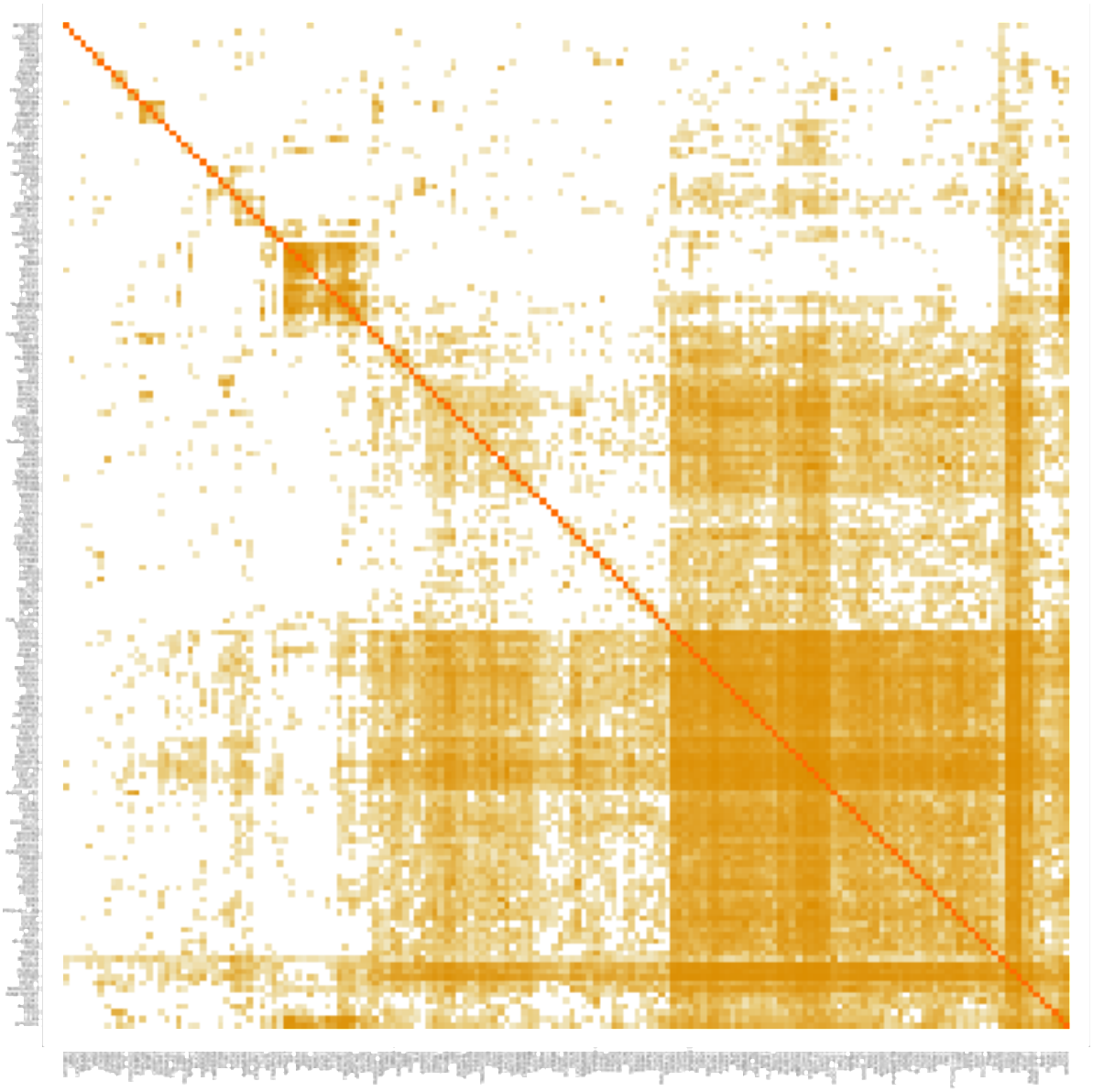
Supplementary Figure 6. Gene communities E, F, G and H (see Table 1) from cell cluster 1.2. Genes and links are highlighted as in Fig. S4.



Supplementary Figure 7. Gene communities I, K, L and M (see Table 1) from cell cluster 1.2. Genes and links are highlighted as in Fig. S4. In communities L and M, links supported by known protein-protein interactions are highlighted in brown.



Supplementary Figure 8. Graphical representation of positive correlations among genes significantly upregulated in lung epithelial cells treated with IL13 (Jackson et al., 2020). Of 419 upregulated genes, 313 form a single connected community in the treated cells (A) whereas 45 of those correlate in the untreated group (B). Green lines represent statistically significant positive correlations. Transcription factors are marked with blue text and a yellow box.



Supplementary Figure 9. Gene-gene correlation among genes that positively correlate with MUC16 in IL13-treated secretory lung epithelial cells. Genes were clustered using complete-linkage hierarchical clustering. Darker color indicates stronger correlation; white indicates no significant correlation.

2.12 References

1. Tritschler S, Buttner M, Fischer DS, Lange M, Bergen V, Lickert H, Theis FJ: Concepts and limitations for learning developmental trajectories from single cell genomics. *Development* 2019, 146.

2. Tam PPL, Ho JWK: Cellular diversity and lineage trajectory: insights from mouse single cell transcriptomes. *Development* 2020, 147.

3. Nguyen H, Tran D, Tran B, Pehlivan B, Nguyen T: A comprehensive survey of regulatory network inference methods using single cell RNA sequencing data. *Brief Bioinform* 2021, 22.

4. Xie B, Jiang Q, Mora A, Li X: Automatic cell type identification methods for single-cell RNA sequencing. *Comput Struct Biotechnol J* 2021, 19:5874-5887.

5. Junttila S, Smolander J, Elo LL: Benchmarking methods for detecting differential states between conditions from multi-subject single-cell RNA-seq data. *Brief Bioinform* 2022, 23.

6. Das S, Rai A, Rai SN: Differential Expression Analysis of Single-Cell RNA-Seq Data: Current Statistical Approaches and Outstanding Challenges. *Entropy (Basel)* 2022, 24.

7. Simmons S: Cell Type Composition Analysis: Comparison of statistical methods. *bioRxiv* 2022:2022.2002.2004.479123.

8. Wang H, Ma X: Learning discriminative and structural samples for rare cell types with deep generative model. *Brief Bioinform* 2022, 23.

9. Grun D, Lyubimova A, Kester L, Wiebrands K, Basak O, Sasaki N, Clevers H, van Oudenaarden A: Single-cell messenger RNA sequencing reveals rare intestinal cell types. *Nature* 2015, 525:251-255.

10. Jiang L, Chen H, Pinello L, Yuan GC: GiniClust: detecting rare cell types from single-cell gene expression data with Gini index. *Genome Biol* 2016, 17:144.

11. Herman JS, Sagar, Grun D: FateID infers cell fate bias in multipotent progenitors from single-cell RNA-seq data. *Nat Methods* 2018, 15:379-386.

12. Jindal A, Gupta P, Jayadeva, Sengupta D: Discovery of rare cells from voluminous single cell expression data. *Nat Commun* 2018, 9:4719.

13. Setty M, Kiseliovas V, Levine J, Gayoso A, Mazutis L, Pe'er D: Characterization of cell fate probabilities in single-cell data with Palantir. *Nat Biotechnol* 2019, 37:451-460.

14. Wegmann R, Neri M, Schuierer S, Bilican B, Hartkopf H, Nigsch F, Mapa F, Waldt A, Cuttat R, Salick MR, et al: CellSIUS provides sensitive and specific detection of rare cell populations from complex single-cell RNA-seq data. *Genome Biol* 2019, 20:142.

15. Dong R, Yuan GC: GiniClust3: a fast and memory-efficient tool for rare cell type identification. *BMC Bioinformatics* 2020, 21:158.

16. Gerniers A, Bricard O, Dupont P: MicroCellClust: mining rare and highly specific subpopulations from single-cell expression data. *Bioinformatics* 2021, 37:3220-3227.

17. Bej S, Galow AM, David R, Wolfien M, Wolkenhauer O: Automated annotation of rare-cell types from single-cell RNA-sequencing data through synthetic oversampling. *BMC Bioinformatics* 2021, 22:557.

18. Lange M, Bergen V, Klein M, Setty M, Reuter B, Bakhti M, Lickert H, Ansari M, Schniering J, Schiller HB, et al: CellRank for directed single-cell fate mapping. *Nat Methods* 2022, 19:159-170.

19. Jin S, Guerrero-Juarez CF, Zhang L, Chang I, Ramos R, Kuan CH, Myung P, Plikus MV, Nie Q: Inference and analysis of cell-cell communication using CellChat. *Nat Commun* 2021, 12:1088.

20. Zhang L, Nie Q: scMC learns biological variation through the alignment of multiple single-cell genomics datasets. *Genome Biol* 2021, 22:10.

21. Sha Y, Wang S, Bocci F, Zhou P, Nie Q: Inference of Intercellular Communications and Multilayer Gene-Regulations of Epithelial-Mesenchymal Transition From Single-Cell Transcriptomic Data. *Front Genet* 2020, 11:604585.

22. Bocci F, Zhou P, Nie Q: spliceJAC: transition genes and state-specific gene regulation from single-cell transcriptome data. *Mol Syst Biol* 2022, 18:e11176.

23. Dann E, Henderson NC, Teichmann SA, Morgan MD, Marioni JC: Differential abundance testing on single-cell data using k-nearest neighbor graphs. *Nat Biotechnol* 2022, 40:245-253.

24. Bageritz J, Willnow P, Valentini E, Leible S, Boutros M, Teleman AA: Gene expression atlas of a developing tissue by single cell expression correlation analysis. *Nat Methods* 2019, 16:750-756.

25. Kolodziejczyk AA, Kim JK, Tsang JC, Ilicic T, Henriksson J, Natarajan KN, Tuck AC, Gao X, Buhler M, Liu P, et al: Single Cell RNA-Sequencing of Pluripotent States Unlocks Modular Transcriptional Variation. *Cell Stem Cell* 2015, 17:471-485.

26. Pedraza JM, van Oudenaarden A: Noise propagation in gene networks. *Science* 2005, 307:1965-1969.

27. Becskei A, Kaufmann BB, van Oudenaarden A: Contributions of low molecule number and chromosomal positioning to stochastic gene expression. *Nat Genet* 2005, 37:937-944.

28. Stewart-Ornstein J, Weissman JS, El-Samad H: Cellular noise regulons underlie fluctuations in Saccharomyces cerevisiae. *Mol Cell* 2012, 45:483-493.

29. Padovan-Merhar O, Raj A: Using variability in gene expression as a tool for studying gene regulation. *Wiley Interdiscip Rev Syst Biol Med* 2013, 5:751-759.

30. Munsky B, Neuert G, van Oudenaarden A: Using gene expression noise to understand gene regulation. *Science* 2012, 336:183-187.

31. Warmflash A, Dinner AR: Signatures of combinatorial regulation in intrinsic biological noise. *Proc Natl Acad Sci U S A* 2008, 105:17262-17267.

32. Gupta A, Martin-Rufino JD, Jones TR, Subramanian V, Qiu X, Grody EI, Bloemendal A, Weng C, Niu SY, Min KH, et al: Inferring gene regulation from stochastic transcriptional variation across single cells at steady state. *Proc Natl Acad Sci U S A* 2022, 119:e2207392119.

33. He Z, Pan Y, Shao F, Wang H: Identifying Differentially Expressed Genes of Zero Inflated Single Cell RNA Sequencing Data Using Mixed Model Score Tests. *Front Genet* 2021, 12:616686.

34. Choudhary S, Satija R: Comparison and evaluation of statistical error models for scRNAseq. *Genome Biol* 2022, 23:27.

35. Sarkar A, Stephens M: Separating measurement and expression models clarifies confusion in single-cell RNA sequencing analysis. *Nat Genet* 2021, 53:770-777.

36. Choi K, Chen Y, Skelly DA, Churchill GA: Bayesian model selection reveals biological origins of zero inflation in single-cell transcriptomics. *Genome Biol* 2020, 21:183.

37. Wang J, Huang M, Torre E, Dueck H, Shaffer S, Murray J, Raj A, Li M, Zhang NR: Gene expression distribution deconvolution in single-cell RNA sequencing. *Proc Natl Acad Sci U S A* 2018, 115:E6437-E6446.

38. Kim JK, Kolodziejczyk AA, Ilicic T, Teichmann SA, Marioni JC: Characterizing noise structure in single-cell RNA-seq distinguishes genuine from technical stochastic allelic expression. *Nat Commun* 2015, 6:8687.

39. DiCiccio CJ, Romano JP: Robust Permutation Tests For Correlation And Regression Coefficients. *Journal of the American Statistical Association* 2017, 112:1211-1220,.

40. Jin S, MacLean AL, Peng T, Nie Q: scEpath: energy landscape-based inference of transition probabilities and cellular trajectories from single-cell transcriptomic data. *Bioinformatics* 2018, 34:2077-2086.

41. Yang XH, Goldstein A, Sun Y, Wang Z, Wei M, Moskowitz IP, Cunningham JM: Detecting critical transition signals from single-cell transcriptomes to infer lineage-determining transcription factors. *Nucleic Acids Res* 2022, 50:e91.

42. Fisher RA: *Statistical Methods for Research Workers.* Eleventh edn. Edinburgh: Oliver and Boyd; 1950.

43. Lause J, Berens P, Kobak D: Analytic Pearson residuals for normalization of single-cell RNA-seq UMI data. *Genome Biol* 2021, 22:258.

44. Bahar Halpern K, Tanami S, Landen S, Chapal M, Szlak L, Hutzler A, Nizhberg A, Itzkovitz S: Bursty gene expression in the intact mammalian liver. *Mol Cell* 2015, 58:147-156.

45. Thattai M, van Oudenaarden A: Intrinsic noise in gene regulatory networks. *Proc Natl Acad Sci U S A* 2001, 98:8614-8619.

46. Schwabe A, Rybakova KN, Bruggeman FJ: Transcription stochasticity of complex gene regulation models. *Biophys J* 2012, 103:1152-1161.

47. Beal J: Biochemical complexity drives log-normal variation in genetic expression. *Engineering Biology* 2017, 1:55-60.

48. Hafemeister C, Satija R: Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. *Genome Biol* 2019, 20:296.

49. Benjamini Y, Hochberg Y: Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *J R Statist Soc B* 1995, 57:289-300.

50. Torre E, Dueck H, Shaffer S, Gospocic J, Gupte R, Bonasio R, Kim J, Murray J, Raj A: Rare Cell Detection by Single-Cell RNA Sequencing as Guided by Single-Molecule RNA FISH. *Cell Syst* 2018, 6:171-179 e175.

51. Ibn-Salem J, Muro EM, Andrade-Navarro MA: Co-regulation of paralog genes in the three-dimensional chromatin architecture. *Nucleic Acids Res* 2017, 45:81-91.

52. Pons P, Latapy M: Computing communities in large networks using random walks. In *Computer and Information Sciences - Iscis 2005,. Volume* 3733. Berlin: Springer Berlin; 2005: 284-293

53. Ilicic T, Kim JK, Kolodziejczyk AA, Bagger FO, McCarthy DJ, Marioni JC, Teichmann SA: Classification of low quality cells from single-cell RNA-seq data. *Genome Biol* 2016, 17:29.

54. Langemeijer SM, Mariani N, Knops R, Gilissen C, Woestenenk R, de Witte T, Huls G, van der Reijden BA, Jansen JH: Apoptosis-Related Gene Expression Profiling in Hematopoietic Cell Fractions of MDS Patients. *PLoS One* 2016, 11:e0165582.

55. Tyler SR, Guccione E, Schadt EE: Anti-correlated Feature Selection Prevents False Discovery of Subpopulations in scRNAseq. *bioRxiv* 2022:1-32.

56. Mootha VK, Lindgren CM, Eriksson KF, Subramanian A, Sihag S, Lehar J, Puigserver P, Carlsson E, Ridderstrale M, Laurila E, et al: PGC-1alpha-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nat Genet* 2003, 34:267-273.

57. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, Paulovich A, Pomeroy SL, Golub TR, Lander ES, Mesirov JP: Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci U S A* 2005, 102:15545-15550.

58. Huang DW, Sherman BT, Lempicki RA: Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nat Protoc* 2009, 4:44-57.

59. Sykes EK, Mactier S, Christopherson RI: Melanoma and the Unfolded Protein Response. *Cancers (Basel)* 2016, 8.

60. Rather RA, Bhagat M, Singh SK: Oncogenic BRAF, endoplasmic reticulum stress, and autophagy: Crosstalk and therapeutic targets in cutaneous melanoma. *Mutat Res Rev Mutat Res* 2020, 785:108321.

61. Manga P, Choudhury N: The unfolded protein and integrated stress response in melanoma and vitiligo. *Pigment Cell Melanoma Res* 2021, 34:204-211.

62. Netanely D, Leibou S, Parikh R, Stern N, Vaknine H, Brenner R, Amar S, Factor RH, Perluk T, Frand J, et al: Classification of node-positive melanomas into prognostic subgroups using keratin, immune, and melanogenesis expression patterns. *Oncogene* 2021, 40:1792-1805.

63. Rambow F, Job B, Petit V, Gesbert F, Delmas V, Seberg H, Meurice G, Van Otterloo E, Dessen P, Robert C, et al: New Functional Signatures for Understanding Melanoma Biology from Tumor Cell Lineage-Specific Analysis. *Cell Rep* 2015, 13:840-853.

64. Wouters J, Kalender-Atak Z, Minnoye L, Spanier KI, De Waegeneer M, Bravo Gonzalez-Blas C, Mauduit D, Davie K, Hulselmans G, Najem A, et al: Robust gene expression programs underlie recurrent cell states and phenotype switching in melanoma. *Nat Cell Biol* 2020, 22:986-998.

65. Eriksson J, Le Joncour V, Nummela P, Jahkola T, Virolainen S, Laakkonen P, Saksela O, Holtta E: Gene expression analyses of primary melanomas reveal CTHRC1 as an important player in melanoma progression. *Oncotarget* 2016, 7:15065-15092.

66. Tsoi J, Robert L, Paraiso K, Galvan C, Sheu KM, Lay J, Wong DJL, Atefi M, Shirazi R, Wang X, et al: Multi-stage Differentiation Defines Melanoma Subtypes with Differential Vulnerability to Drug-Induced Iron-Dependent Oxidative Stress. *Cancer Cell* 2018, 33:890-904 e895.

67. Verfaillie A, Imrichova H, Atak ZK, Dewaele M, Rambow F, Hulselmans G, Christiaens V, Svetlichnyy D, Luciani F, Van den Mooter L, et al: Decoding the regulatory landscape of melanoma reveals TEADS as regulators of the invasive cell state. *Nat Commun* 2015, 6:6683.

68. Wang J, Saraswat D, Sinha AK, Polanco J, Dietz K, O'Bara MA, Pol SU, Shayya HJ, Sim FJ: Paired Related Homeobox Protein 1 Regulates Quiescence in Human Oligodendrocyte Progenitors. *Cell Rep* 2018, 25:3435-3450 e3436.

69. Blattmann P, Henriques D, Zimmermann M, Frommelt F, Sauer U, Saez-Rodriguez J, Aebersold R: Systems Pharmacology Dissection of Cholesterol Regulation Reveals Determinants of Large Pharmacodynamic Variability between Cell Lines. *Cell Syst* 2017, 5:604-619 e607.

70. Capell-Hattam IM, Fenton NM, Coates HW, Sharpe LJ, Brown AJ: The Non Catalytic Protein ERG28 has a Functional Role in Cholesterol Synthesis and is Coregulated Transcriptionally. *J Lipid Res* 2022, 63:100295.

71. Gong Y, Lee JN, Brown MS, Goldstein JL, Ye J: Juxtamembranous aspartic acid in Insig-1 and Insig-2 is required for cholesterol homeostasis. *Proc Natl Acad Sci U S A* 2006, 103:6154-6159.

72. Jeong SJ, Kim S, Park JG, Jung IH, Lee MN, Jeon S, Kweon HY, Yu DY, Lee SH, Jang Y, et al: Prdx1 (peroxiredoxin 1) deficiency reduces cholesterol efflux via impaired macrophage lipophagic flux. *Autophagy* 2018, 14:120-133.

73. Schallreuter KU, Hasse S, Rokos H, Chavan B, Shalbaf M, Spencer JD, Wood JM: Cholesterol regulates melanogenesis in human epidermal melanocytes and melanoma cells. *Exp Dermatol* 2009, 18:680-688.

74. Nikolakaki E, Simos G, Georgatos SD, Giannakouros T: A nuclear envelope-associated kinase phosphorylates arginine-serine motifs and modulates interactions between the lamin B receptor and other nuclear proteins. *J Biol Chem* 1996, 271:8365-8372.

75. Singh P, Saxena R, Srinivas G, Pande G, Chattopadhyay A: Cholesterol biosynthesis and homeostasis in regulation of the cell cycle. *PLoS One* 2013, 8:e58833.

76. Jackson ND, Everman JL, Chioccioli M, Feriani L, Goldfarbmuren KC, Sajuthi SP, Rios CL, Powell R, Armstrong M, Gomez J, et al: Single-Cell and Population Transcriptomics Reveal Pan-epithelial Remodeling in Type 2-High Asthma. *Cell Rep* 2020, 32:107872.

77. Tuvim MJ, Mospan AR, Burns KA, Chua M, Mohler PJ, Melicoff E, Adachi R, Ammar-Aouchiche Z, Davis CW, Dickey BF: Synaptotagmin 2 couples mucin granule exocytosis to Ca2+ signaling from endoplasmic reticulum. *J Biol Chem* 2009, 284:9781-9787.

78. Ding L, Abebe T, Beyene J, Wilke RA, Goldberg A, Woo JG, Martin LJ, Rothenberg ME, Rao M, Hershey GK, et al: Rank-based genome-wide analysis reveals the association of ryanodine receptor-2 gene variants with childhood asthma among human populations. *Hum Genomics* 2013, 7:16.

79. Baran Y, Bercovich A, Sebe-Pedros A, Lubling Y, Giladi A, Chomsky E, Meir Z, Hoichman M, Lifshitz A, Tanay A: MetaCell: analysis of single-cell RNA-seq data using K-nn graph partitions. *Genome Biol* 2019, 20:206.

80. Ben-Kiki O, Bercovich A, Lifshitz A, Tanay A: Metacell-2: a divide-and-conquer metacell algorithm for scalable scRNAseq analysis. *Genome Biol* 2022, 23:100.

81. Persad S, Choo ZN, Dien C, Sohail N, Masilionis I, Chaligne R, Nawy T, Brown CC, Sharma R, Pe'er I, et al: SEACells infers transcriptional and epigenomic cellular states from single-cell genomics data. *Nat Biotechnol* 2023, 41:1746-1757.

82. Lun ATL, Marioni JC: Overcoming confounding plate effects in differential expression analyses of single-cell RNA-seq data. *Biostatistics* 2017, 18:451-464.

83. Bilous M, Tran L, Cianciaruso C, Gabriel A, Michel H, Carmona SJ, Pittet MJ, Gfeller D: Metacells untangle large and complex single-cell transcriptome networks. *BMC Bioinformatics* 2022, 23:336.

84. Morabito S, Reese F, Rahimzadeh N, Miyoshi E, Swarup V: hdWGCNA identifies co-expression networks in high-dimensional transcriptomics data. *Cell Rep Methods* 2023, 3:100498.

85. Squair JW, Gautier M, Kathe C, Anderson MA, James ND, Hutson TH, Hudelle R, Qaiser T, Matson KJE, Barraud Q, et al: Confronting false discoveries in single-cell differential expression. *Nat Commun* 2021, 12:5692.

86. Xu H, Hu Y, Zhang X, Aouizerat BE, Yan C, Xu K: A novel graph-based k-partitioning approach improves the detection of gene-gene correlations by single-cell RNA sequencing. *BMC Genomics* 2022, 23:35.

87. Eisen TJ, Eichhorn SW, Subtelny AO, Lin KS, McGeary SE, Gupta S, Bartel DP: The Dynamics of Cytoplasmic mRNA Metabolism. *Mol Cell* 2020, 77:786-799 e710.

88. Davis AF, Clayton DA: In situ localization of mitochondrial DNA replication in intact mammalian cells. *J Cell Biol* 1996, 135:883-893.

89. Yao Z, Luo J, Hu K, Lin J, Huang H, Wang Q, Zhang P, Xiong Z, He C, Huang Z, et al: ZKSCAN1 gene and its related circular RNA (circZKSCAN1) both inhibit hepatocellular carcinoma cell growth, migration, and invasion but through different signaling pathways. *Mol Oncol* 2017, 11:422-437.

90. Dollinger E, Silkwood K, Atwood S, Nie Q, A.D. L: A principled, robust approach to feature selection in single cell transcriptomics. *bioRxiv* 2023:*to be submitted*.

91. Cornish EA, Fisher RA: Moments and cumulants in the specification of distributions. *Review of the International Statistical Institute* 1938, 5:307-320.

92. Hu Y, Ewen-Campen B, Comjean A, Rodiger J, Mohr SE, Perrimon N: Paralog Explorer: A resource for mining information about paralogs in common research organisms. *Comput Struct Biotechnol J* 2022, 20:6570-6577.

93. Waltman L, van Eck NJ: A smart local moving algorithm for large-scale modularity-based community detection. *The European Physical Journal B* 2013, 86:471.

94. Butler A, Hoffman P, Smibert P, Papalexi E, Satija R: Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat Biotechnol* 2018, 36:411-420.

2.13 Appendix

2.13.1 Deriving p-values from expectation matrices

Let a_{ij} stand for the UMI entries of the m x n matrix of m-genes and n-cells. As described above, we use the totals over all i and the totals over all j to generate an "expectation matrix", i.e. a matrix whose entries μ_{ij} correspond to the expected values of the numbers of UMI for each gene in each cell under the null hypothesis (the hypothesis that gene expression is identical across cells).

To derive p-values for modified corrected Fano factors (ϕ') or modified corrected Pearson Correlation Coefficients (PCC') we derive, starting from the values in the expectation matrix and the expected distribution of those values, the expected distributions of the ϕ' and PCC' under the null hypothesis. Identifying those distributions enables one to determine the probability, under the null hypothesis, that any given observation would have occurred.

This is done in two steps: First we calculate the moments (or equivalently, cumulants) of ϕ' and PCC' under the null hypothesis. Second, we use the Cornish Fisher method to reconstruct the tails of distributions from their cumulants, determining how far out on the tail any empirical observation lies.

2.13.2 The Poisson-Log-Normal (PLN) Distribution

We define the Poisson Log Normal distribution as the discrete distribution obtained by random sampling of integer values from a Log normal distribution. Specifically, we parametrize the Log normal distribution in terms of its mean μ and coefficient of variation c (its actual mean and coefficient of variation, not that of the underlying normal distribution in terms of which log normal distributions are often defined). According to Bulmer (1974), compounding the the Poisson distribution with any other distribution that has a PDF of f(λ) gives a PDF of

$$P[r] == \int_0^\infty \frac{\lambda^r e^{-\lambda}}{r!} f(\lambda)\, d\lambda == \frac{1}{r!} \int_0^\infty \lambda^r e^{-\lambda} f(\lambda)\, d\lambda \qquad (1)$$

the PDF of a Log-normal distribution parametrized in terms of its actual mean and cv may be written as

$$\frac{e^{-\frac{\text{Log}\left[\frac{\sqrt{1+c^2}\lambda}{\mu}\right]^2}{2\text{Log}[1+c^2]}}}{\sqrt{2\pi}\lambda\sqrt{\text{Log}[1+c^2]}}$$

Thus we may express the PDF of the PLN distribution as:

$$P[r] == \frac{1}{\sqrt{2\pi\text{Log}[1+c^2]}r!} \int_0^\infty \lambda^{r-1} e^{-\lambda} e^{-\frac{\left(\text{Log}\left[\lambda\frac{\sqrt{1+c^2}}{m}\right]\right)^2}{2\text{Log}[1+c^2]}}\, d\lambda \qquad (2)$$

Bullmer also shows, by calculating the factorial moments of this distribution by summing over r for fixed λ, and then integrating, that the k-th factorial moment of this distribution is equal to the k-th ordinary moment of the corresponding lognormal distribution, which in our parameterization is:

$$(1+c^2)^{\frac{k^2}{2}}\left(\frac{m}{\sqrt{1+c^2}}\right)^k$$

We note that the r-th raw moment of a random variable X can be expressed in terms of its factorial moments by the formula

$$\sum_{j=0}^{r}\left\{{r \atop j}\right\}E\left[(X)_j\right]$$

where $E\left[(X)_j\right]$is the j-th factorial moment, and the curly braces denote Stirling numbers of the second kind, which are specified by the formula

$$\left\{{a \atop b}\right\}\to\frac{1}{b!}\sum_{i=0}^{b}(-1)^i\binom{b}{i}(b-i)^a$$

Putting this together gives us a formula for the raw moments of a Poisson log-normal distribution

$$r[k]=\sum_{j=0}^{k}\mathcal{S}_{k,j}(1+c^2)^{\frac{j^2}{2}}\left(\frac{m}{\sqrt{1+c^2}}\right)^j=\sum_{j=0}^{k}\mathcal{S}_{k,j}(1+c^2)^{-\frac{j}{2}+\frac{j^2}{2}}m^j$$

where $\mathcal{S}_{k,j}=\left\{{k \atop j}\right\}$.

Using this formula, we may easily write the first five moments of the PLN distribution:

$r[1]=m,$

$r[2]=m+(1+c^2)m^2,$

$r[3]=m+3(1+c^2)m^2+(1+c^2)^3m^3,$

$r[4] = m + 7(1 + c^2)m^2 + 6(1 + c^2)^3 m^3 + (1 + c^2)^6 m^4,$

$r[5] = m + 15(1 + c^2)m^2 + 25(1 + c^2)^3 m^3 + 10(1 + c^2)^6 m^4 + (1 + c^2)^{10} m^5$

Note that when c<<1, this reduces to the known moments of the Poisson distribution, which are the Bell polynomials $B_k(m)$.

The distribution of Pearson residuals under the null hypothesis will be the distribution of values of $\frac{(X-m)}{a}$ where m and α are constants (in ordinary Pearson residuals $\alpha = \sqrt{m(1 + c^2 m)}$). Division of the values of a distribution by any constant alters the moments of the distribution in a simple way: each moment r[k] is simply divided by α^k.

The distribution of values X-m, where X is drawn from a PLN distribution, can be obtained from the definition of the k-th moment for the distribution of a random variable minus the constant m, which may be written as $\sum_{i=0}^{k}(x - m)^i P[x]\, dx$ where P[x] is the PDF of the random variable. Noting that $(x - m)^i$ can be expanded as $\sum_{i=0}^{k} \binom{k}{i} (-m)^{k-i} x^i$, we can re-write this as $\sum_{i=0}^{k} \binom{k}{i} (-m)^{k-i} x^i P[x]$. Noting that $\sum_{i=0}^{k} x^i P[x]$ are simply the moments of the distribution defined by P[x], we then get that the k-th moment of the distribution of X-m is simply

$$\sum_{i=0}^{k} \binom{k}{i} (-\mu m)^{k-i} r[i]$$

Accounting for division by α which in the case of modified corrected Pearson residuals will be $\sqrt{\mu(1+c^2\mu)}$ gives us an equation for the k-th moments of a modified corrected Pearson residual, namely

$$\frac{1}{\left(\sqrt{m(1+c^2m)}\right)^k}\sum_{i=0}^{k}\binom{k}{i}(-m)^{k-i}r[i]$$

where the r[i] are defined as above.

2.13.3 Moments of modified corrected Fano factors

Modified corrected Fano factors (ϕ') are obtained by summing over n squared modified corrected Pearson residuals, and dividing by n-1. Given the definition of the i-th moment of a distribution as the expectation values of x^i, it is easy to see that the moments of the square of a random variable are just the even-numbered moments of the random variable, i.e. moments 1,2,3... of squared Pearson residuals are simply the moments 2,4,6... of the Pearson residuals themselves. Accordingly, we replace the above formula with

$$\frac{1}{\left(m(1+c^2m)\right)^k}\sum_{i=0}^{2k}\binom{2k}{i}(-m)^{2k-i}r[i]$$

Putting this together with the definitions of r[i] we may write the first 5 moments of the squared modified corrected Pearson residual, which we will denote as $\mathcal{E}_1$, as

$\mathcal{E}_1 \to 1,$

$$\mathcal{E}_2 \to \frac{1+m\left(3+c^2\left(7+m\left(6+3c^2(6+m)+c^4(6+(16+15c^2+6c^4+c^6)m)\right)\right)\right)}{m(1+c^2m)^2},$$

$$\mathcal{E}_3 \to \frac{1}{m^2(1+c^2m)^3}\times$$

$$\big(1+31(1+c^2)m+90(1+c^2)^3m^2+65(1+c^2)^6m^3+15(1+c^2)^{10}m^4-5m^5$$
$$+(1+c^2)^{15}m^5$$
$$+15m^4(1+m+c^2m)20m^2(m+3(1+c^2)m^2+(1+c^2)^3m^3)$$
$$+15m(m+7(1+c^2)m^2+6(1+c^2)^3m^3+(1+c^2)^6m^4)$$
$$-6(m+15(1+c^2)m^2+25(1+c^2)^3m^3+10(1+c^2)^6m^4+(1+c^2)^{10}m^5)\big),$$

$$\mathcal{E}_4\to\frac{1}{m^3(1+c^2m)^4}\times$$

$$\big(1+127(1+c^2)m+966(1+c^2)^3m^2+1701(1+c^2)^6m^3+1050(1+c^2)^{10}m^4$$
$$+266(1+c^2)^{15}m^5+28(1+c^2)^{21}m^6-7m^7+(1+c^2)^{28}m^7$$
$$+28m^6(1+m+c^2m)-56m^4(m+3(1+c^2)m^2+(1+c^2)^3m^3)$$
$$+70m^3(m+7(1+c^2)m^2+6(1+c^2)^3m^3+(1+c^2)^6m^4)$$
$$-56m^2\times$$
$$(m+15(1+c^2)m^2+25(1+c^2)^3m^3+10(1+c^2)^6m^4+(1+c^2)^{10}m^5)$$
$$+28m\times$$
$$(m+31(1+c^2)m^2+90(1+c^2)^3m^3+65(1+c^2)^6m^4+15(1+c^2)^{10}m^5$$
$$+(1+c^2)^{15}m^6)$$
$$-8(m+63(1+c^2)m^2+301(1+c^2)^3m^3+350(1+c^2)^6m^4$$
$$+140(1+c^2)^{10}m^5+21(1+c^2)^{15}m^6+(1+c^2)^{21}m^7)\big),$$

$$\mathcal{E}_5\to\frac{1}{m^4(1+c^2m)^5}\times$$

$$\big(1 + 511(1 + c^2)m + 9330(1 + c^2)^3m^2 + 34105(1 + c^2)^6m^3 + 42525(1 + c^2)^{10}m^4$$
$$+ 22827(1 + c^2)^{15}m^5 + 5880(1 + c^2)^{21}m^6 + 750(1 + c^2)^{28}m^7$$
$$+ 45(1 + c^2)^{36}m^8 - 9m^9 + (1 + c^2)^{45}m^9 + 45m^8(1 + m + c^2m)$$
$$- 120m^6(m + 3(1 + c^2)m^2 + (1 + c^2)^3m^3)$$
$$+ 210m^5(m + 7(1 + c^2)m^2 + 6(1 + c^2)^3m^3 + (1 + c^2)^6m^4)$$
$$- 252m^4 \times$$
$$(m + 15(1 + c^2)m^2 + 25(1 + c^2)^3m^3 + 10(1 + c^2)^6m^4 + (1 + c^2)^{10}m^5)$$
$$+ 210m^3 \times$$
$$(m + 31(1 + c^2)m^2 + 90(1 + c^2)^3m^3 + 65(1 + c^2)^6m^4 + 15(1 + c^2)^{10}m^5$$
$$+ (1 + c^2)^{15}m^6)$$
$$- 120m^2 \times$$
$$(m + 63(1 + c^2)m^2 + 301(1 + c^2)^3m^3 + 350(1 + c^2)^6m^4$$
$$+ 140(1 + c^2)^{10}m^5 + 21(1 + c^2)^{15}m^6 + (1 + c^2)^{21}m^7)$$
$$+ 45m \times$$
$$(m + 127(1 + c^2)m^2 + 966(1 + c^2)^3m^3 + 1701(1 + c^2)^6m^4$$
$$+ 1050(1 + c^2)^{10}m^5 + 266(1 + c^2)^{15}m^6 + 28(1 + c^2)^{21}m^7 + (1 + c^2)^{28}m^8)$$
$$-10(m + 255(1 + c^2)m^2 + 3025(1 + c^2)^3m^3 + 7770(1 + c^2)^6m^4$$
$$+ 6951(1 + c^2)^{10}m^5 + 2646(1 + c^2)^{15}m^6 + 462(1 + c^2)^{21}m^7$$
$$+ 36(1 + c^2)^{28}m^8 + (1 + c^2)^{36}m^9)\big)$$

From here we note that the Fano factor is obtained by taking a sum of squared, modified, corrected Pearson residuals, each of which is characterized by its own set of moments and its

own value of m. Thus, we need add indices to the values of m, i.e. using m[j] to refer to the value of m in cell j.

To obtain the moments of the sum of random variables we use the fact that the moment generating function (MGF) of a sum of independent random variables is equal to the product of the MGFs of the individual random variables. In practice, it is easier to generate solutions by switching from moments to cumulants, for which an analogous rule exists, namely that the cumulant generating function (KGF) of a sum of independent random variables is the sum of the cumulant generating functions of the individual random variables. As the k-th cumulant is simply the k-th derivative of the KGF, and differentiation distributes over addition, we see that the k-th cumulant of a sum of random variables is just the sum of the k-th cumulants of each of the random variables.

Cumulants may be obtained from moments by noting the the KGF is simply the logarithm of the MGF, from which the following relationship between the first 5 moments ($\mathcal{E}_1$, $\mathcal{E}_2$, $\mathcal{E}_3$, $\mathcal{E}_4$, $\mathcal{E}_5$) and first five cumulants (κ_1, κ_2, κ_3, κ_4, κ_5) may be derived

$$\kappa_1 = \mathcal{E}_1,$$

$$\kappa_2 = -\mathcal{E}_1^2 + \mathcal{E}_2,$$

$$\kappa_3 = 2\mathcal{E}_1^3 - 3\mathcal{E}_1\mathcal{E}_2 + \mathcal{E}_3,$$

$$\kappa_4 = -6\mathcal{E}_1^4 + 12\mathcal{E}_1^2\mathcal{E}_2 - 3\mathcal{E}_2^2 - 4\mathcal{E}_1\mathcal{E}_3 + \mathcal{E}_4,$$

$$\kappa_5 = 24\mathcal{E}_1^5 - 60\mathcal{E}_1^3\mathcal{E}_2 + 20\mathcal{E}_1^2\mathcal{E}_3 - 10\mathcal{E}_2\mathcal{E}_3 + 5\mathcal{E}_1(6\mathcal{E}_2^2 - \mathcal{E}_4) + \mathcal{E}_5$$

Finally, recalling that the modified corrected Fano factor ϕ' is obtained by summing over n squared, modified, corrected Pearson residuals and then dividing by n-1, to obtain the moments (or cumulants) of ϕ' we must divide the k-th moment (or cumulant) by $(n-1)^k$. Doing this, and

making the appropriate substitutions for κ, E and m in each cell, we obtain the following final formulae for the first five cumulants of ϕ', where the terms have been made a bit more compact by substituting χ for $1+c^2$

$$\kappa_1 \to \frac{n}{-1+n}$$

$$\kappa_2 \to$$

$$\frac{1}{(-1+n)^2} \times$$

$$\left(\sum_{j=1}^{n}\left(-1+\frac{1+m[j](-4+7\chi+6(1-2\chi+\chi^3)m[j]+(-3+6\chi-4\chi^3+\chi^6)m[j]^2)}{m[j](1+(-1+\chi)m[j])^2}\right)\right)$$

$$\kappa_3 \to \frac{1}{(-1+n)^3} \times$$

$$\left(\sum_{j=1}^{n}\frac{1}{m[j]^2(1+(-1+\chi)m[j])^3}\right.$$

$$(1+(-9+31\chi)m[j]+2(16-57\chi+45\chi^3)m[j]^2$$

$$+(-56+180\chi-21\chi^2-168\chi^3+65\chi^6)m[j]^3$$

$$+3(16-48\chi+14\chi^2+40\chi^3-6\chi^4-21\chi^6+5\chi^{10})m[j]^4$$

$$\left.+(-16+48\chi-24\chi^2-30\chi^3+12\chi^4+18\chi^6-3\chi^7-6\chi^{10}+\chi^{15})m[j]^5)\right)$$

$$\kappa_4 \to \frac{1}{(-1+n)^4} \times$$

$$\left(\sum_{j=1}^{n} \frac{1}{m[j]^3 (1+(-1+\chi)m[j])^4} \right.$$

$$(1+(-15+127\chi)m[j]+(92-674\chi+966\chi^3)m[j]^2$$

$$+(-302+1724\chi-271\chi^2-2804\chi^3+1701\chi^6)m[j]^3$$

$$+6(96-452\chi+174\chi^2+620\chi^3-102\chi^4-511\chi^6+175\chi^{10})m[j]^4$$

$$+2\times$$

$$(-320+1344\chi-822\chi^2-1390\chi^3+672\chi^4+1124\chi^6-151\chi^7-590\chi^{10}$$

$$+133\chi^{15})m[j]^5$$

$$+4$$

$$\times(96-384\chi+312\chi^2+278\chi^3-276\chi^4+18\chi^5-194\chi^6+84\chi^7-9\chi^9$$

$$+126\chi^{10}-15\chi^{11}-43\chi^{15}+7\chi^{21})m[j]^6$$

$$+(-96+384\chi-384\chi^2-160\chi^3+314\chi^4-48\chi^5+112\chi^6-120\chi^7$$

$$+12\chi^8+24\chi^9-80\chi^{10}+24\chi^{11}-3\chi^{12}+32\chi^{15}-4\chi^{16}-8\chi^{21}+\chi^{28})$$

$$\left. m[j]^7) \right)$$

$$\kappa_5 \to \frac{1}{(-1+n)^5} \times$$

$$\left(\sum_{j=1}^{n} \frac{1}{m[j]^4 (1+(-1+\chi)m[j])^5} \right.$$

$$(1+(-25+511\chi)m[j]+30(8-119\chi+311\chi^3)m[j]^2$$

$$+5(-248+2540\chi-561\chi^2-7208\chi^3+6821\chi^6)m[j]^3$$

$$+(3904-29880\chi+15690\chi^2+68000\chi^3-12990\chi^4-86865\chi^6$$

$$+42525\chi^{10})m[j]^4$$

$$+(-7872+49360\chi-39660\chi^2-81110\chi^3+46460\chi^4+98270\chi^6$$

$$-13365\chi^7-74910\chi^{10}+22827\chi^{15})m[j]^5$$

$$+20\times$$

$$(512-2832\chi+2898\chi^2+3168\chi^3-3654\chi^4+216\chi^5-3109\chi^6+1499\chi^7$$

$$-240\chi^9+2953\chi^{10}-315\chi^{11}-1390\chi^{15}+294\chi^{21})m[j]^6$$

$$+10\times$$

$$(-832+4288\chi-5136\chi^2-2940\chi^3+6222\chi^4-1008\chi^5+2276\chi^6$$

$$-2778\chi^7+172\chi^8+806\chi^9-2636\chi^{10}+842\chi^{11}-65\chi^{12}-90\chi^{13}$$

$$+1420\chi^{15}-140\chi^{16}-476\chi^{21}+75\chi^{28})m[j]^7$$

$$+5\times$$

$$(768-3840\chi+5184\chi^2+1152\chi^3-5656\chi^4+1728\chi^5-936\chi^6+2432\chi^7$$

$$-420\chi^8-960\chi^9+1448\chi^{10}-912\chi^{11}+186\chi^{12}+192\chi^{13}-720\chi^{15}$$

$$+170\chi^{16}-12\chi^{18}+288\chi^{21}-28\chi^{22}-73\chi^{28}+9\chi^{36})m[j]^8$$

$$+(-768+3840\chi-5760\chi^2+320\chi^3+5280\chi^4-2536\chi^5+560\chi^6$$

$$-2240\chi^7+840\chi^8+980\chi^9-1072\chi^{10}+880\chi^{11}-300\chi^{12}-210\chi^{13}$$

$$+ 400\chi^{15} - 180\chi^{16} + 20\chi^{17} + 40\chi^{18} - 170\chi^{21} + 40\chi^{22} + 50\chi^{28} - 5\chi^{29}$$

$$- 10\chi^{36} + \chi^{45})m[j]^9)\Bigg)$$

2.13.4 Moments of modified corrected PCCs

The procedure for deriving the moments, and subsequently cumulants, of PCC' are similar to those used above for ϕ', except that certain approximations are required due to the more complicated set of terms that go into defining PCC'. As described in the main text, the PCC is fundamentally an average over products of pairs of Pearson residuals, i.e.

PCC' = $\frac{1}{(n-1)\sqrt{\phi'_a \phi'_b}} \sum_{j=1}^{n} P'_{a,j} P'_{b,j}$

where $P'_{a,j}$ and $P'_{b,j}$ represent the modified corrected Pearson residuals for genes a and b, respectively, in cell j. Calculating the moments of $\sum_{j=1}^{n} P'_{a,j} P'_{b,j}$ requires only one additional rule beyond those enumerated in the previous section, which is that that moment of the product of two random variables is the product of the moments of the individual random variables. With this information it is possible to write out a general formula for the moments of $\sum_{j=1}^{n} P'_{a,j} P'_{b,j}$.

However, accounting for the further division by $(n-1)\sqrt{\phi'_a \phi'_b}$ is less straightforward because ϕ'_a and ϕ'_b are not constants, but are rather distributed according to the Fano factor distributions of genes a and b, which we had just calculated. Moreover, not only must ϕ'_a and ϕ'_b be treated as random variables, they are technically not independent of the other random variables, i.e. those that define the modified corrected Pearson residuals. Despite this, however, when the number of cells is in the hundreds or greater, they may be treated as very nearly

independent of the individual modified corrected Pearson residuals because they effectively average over a very large number of those residuals.

Doing that leaves us only to calculate the moments of $1/\sqrt{\phi'}$ for each gene, which we may then multiply by the moments of $\frac{1}{(n-1)}\sum_{j=1}^{n} P'_{a,j}P'_{b,j}$ to produce the moments of PCC'. Since the first moment of $\frac{1}{(n-1)}\sum_{j=1}^{n} P'_{a,j}P'_{b,j}$ under the null hypothesis is zero (no correlation), we don't actually need the first moments of $1/\sqrt{\phi'}$, just the higher ones.

Unfortunately, analytically deriving the moments for the square root of a random variable with an arbitrary distribution is not generally straightforward, nor is the derivation of moments of the multiplicative inverse of such a distribution. As it happens, when the number of cells is large (at least many hundreds), these moments turn out to be rather close to 1, i.e. simply omitting them has relatively little effect on the moments of PCC'. However, since we cannot always be certain that this will be the case, we developed an approach to approximate their values by simulation.

Briefly, for any given dataset with m genes and n cells, for which a value of the coefficient of variation of gene expression noise has been chosen to be c, we first use the cell specific sequencing depths to calculate the rows of the expectation matrix that would be generated for 8 values of total expression (summing over the cells of the UMI values for any given gene) ranging from the lowest to the highest in the data set (and distributing them logarithmically). From each row of this 8 x n matrix, we repeatedly generate a vector by replacing each entry μ_i with a Poisson Random Variate or a Random Variate of a Log Normal distribution with mean μ_i and coefficient of variation c. We then calculate a modified corrected Fano factor using these entries and takes its square root and multiplicative inverse. We gather a

sufficiently large number of these that the first 5 moments of their distribution adequately converge; not surprisingly this requires a larger number of such vectors when the total gene count is low. Typically, between 300 and 2000 iterations are required, depending on the level of gene expression.

In this way we generate empirical estimates of the moments of $1/\sqrt{\phi'}$ for 8 representative levels of gene expression. From there we use interpolation to estimate the moments of $1/\sqrt{\phi'}$ for any level of gene expression in our data set. Finally to obtain the moments of $1/\sqrt{\phi'_a \phi'_b}$ for pairs of genes a and b we simply multiply the moments of $1/\sqrt{\phi'_a}$ and $1/\sqrt{\phi'_b}$ (since they are independent).

All told, given that the k-th moment of any constant φ times a random variable is φ^k times the k-th moment of the random variable, we get that, for PCC':

$$\mathcal{E}_k(\mathrm{PCC}_{a,b}) = \mathcal{E}_k\left(\frac{1}{(n-1)\sqrt{\phi'_a \phi'_b}} \sum_{j=1}^{n} P'_{a,j} P'_{b,j}\right)$$

$$= \frac{1}{(n-1)^k} \mathcal{E}_k\left(\frac{1}{\sqrt{\phi_a}}\right) \mathcal{E}_k\left(\frac{1}{\sqrt{\phi_b}}\right) \mathcal{E}_k\left(\sum_{j=1}^{n} P'_{a,j} P'_{b,j}\right)$$

where $\mathcal{E}_k$ stands for the k-the moment, and a and b are individual genes. Converting the raw moments to cumulants gives the following formula, where κ_k represents the k-th cumulant, and $m_a[j]$ and $m_b[j]$ stand for the values in column j of the expectation matrix for genes a and b, respectively.

$$\kappa_1 \to 0$$

$$\kappa_2 \to \frac{n}{(n-1)^2} \mathcal{E}_2\left(\frac{1}{\sqrt{\phi_a}}\right) \mathcal{E}_2\left(\frac{1}{\sqrt{\phi_b}}\right)$$

$$\kappa_3 \to$$

$$\frac{1}{(n-1)^3}\mathcal{E}_3\left(\frac{1}{\sqrt{\phi_a}}\right)\mathcal{E}_3\left(\frac{1}{\sqrt{\phi_b}}\right)\times$$

$$\sum_{j=1}^{n}\frac{m_a[j](1+3c^2m_a[j]+c^4(3+c^2)m_a[j]^2)m_b[j](1+3c^2m_b[j]+c^4(3+c^2)m_b[j]^2)}{\left(m_a[j](1+c^2m_a[j])\right)^{3/2}\left(m_b[j](1+c^2m_b[j])\right)^{3/2}}$$

$$\kappa_4 \to \frac{1}{(n-1)^4}\times$$

$$\left(-3n\left(\mathcal{E}_2\left(\frac{1}{\sqrt{\phi_a}}\right)\mathcal{E}_2\left(\frac{1}{\sqrt{\phi_b}}\right)\right)^2\right.$$

$$+\mathcal{E}_4\left(\frac{1}{\sqrt{\phi_a}}\right)\mathcal{E}_4\left(\frac{1}{\sqrt{\phi_b}}\right)\sum_{j=1}^{n}\big((1+(3+7c^2)m_a[j]+6(c^2+3c^4+c^6)m_a[j]^2$$

$$+c^4(3+16c^2+15c^4+6c^6+c^8)m_a[j]^3)(1+(3+7c^2)m_b[j]+6(c^2+3c^4+c^6)m_b[j]^2$$

$$\left.+c^4(3+16c^2+15c^4+6c^6+c^8)m_b[j]^3)\big)/(m_a[j](1+c^2m_a[j])^2m_b[j](1+c^2m_b[j])^2)\right)$$

$$\kappa_5 \to \frac{1}{(n-1)^5} \times$$

$$\left(-10\mathcal{E}_2\left(\frac{1}{\sqrt{\phi_a}}\right)\mathcal{E}_2\left(\frac{1}{\sqrt{\phi_b}}\right)\mathcal{E}_3\left(\frac{1}{\sqrt{\phi_a}}\right)\mathcal{E}_3\left(\frac{1}{\sqrt{\phi_b}}\right) \times \right.$$

$$\sum_{j=1}^{n}\left[\frac{1 + c^2 m_a[j](3 + c^2(3 + c^2)m_a[j])}{\sqrt{m_a[j]}(1 + c^2 m_a[j])^{3/2}}\right.$$

$$\left.\times \frac{1 + c^2 m_b[j](3 + c^2(3 + c^2)m_b[j])}{\sqrt{m_b[j]}(1 + c^2 m_b[j])^{3/2}}\right]$$

$$+ \mathcal{E}_5\left(\frac{1}{\sqrt{\phi_a}}\right)\mathcal{E}_5\left(\frac{1}{\sqrt{\phi_b}}\right)$$

$$\times \sum_{j=1}^{n}\left[\frac{1}{m_a[j]^{3/2}(1 + c^2 m_a[j])^{5/2} m_b[j]^{3/2}(1 + c^2 m_b[j])^{5/2}}\right.$$

$$\times (1 + 5(2 + 3c^2)m_a[j] + 5c^2(8 + 15c^2 + 5c^4)m_a[j]^2$$

$$+ 10c^4(6 + 17c^2 + 15c^4 + 6c^6 + c^8)m_a[j]^3$$

$$+ c^6(30 + 135c^2 + 222c^4 + 205c^6 + 120c^8 + 45c^{10} + 10c^{12} + c^{14})m_a[j]^4)$$

$$(1 + 5(2 + 3c^2)m_b[j] + 5c^2(8 + 15c^2 + 5c^4)m_b[j]^2$$

$$+ 10c^4(6 + 17c^2 + 15c^4 + 6c^6 + c^8)m_b[j]^3$$

$$\left.+ c^6(30 + 135c^2 + 222c^4 + 205c^6 + 120c^8 + 45c^{10} + 10c^{12} + c^{14})\right]$$

2.13.5 From cumulants and observations to p-values

Although there is no analytical method for reconstructing an arbitrary distribution from a finite number of its moments, several methods exist for approximating the body or tails of such distributions. Since we are interested in assessing the probabilities of relatively rare events, we

need a way to estimate tail probabilities reasonably accurately. The Cornish Fisher expansion [Cornish and Fisher, 1938] provides a useful algorithm for doing so. Briefly, from a finite number of cumulants of the distribution under the null hypothesis, and an observed value (in this case an observed PCC'), this expansion estimates the probability of that observation occurring under the null hypothesis, i.e. a p-value. Doing so requires finding the closest real root to zero off polynomials whose coefficients are derived from the cumulants and the observed data. We solve one such equation for each PCC', using a variety of shortcuts to speed the process. Given the existence of potential error in the estimation of moments and PCC' values from finite numbers of cells (see Sources of Error, below), it is sometimes necessary to accept roots when they bring the polynomial close to, but not exactly equal to, zero. This is particularly important when both of the genes involved are of very low expression level, such that the data vectors are very sparse.

2.13.6 Estimating False Discovery rate

The Bonferroni correction is much too conservative for use in multiple hypothesis testing correction because pairwise correlations are far from independent measurements. We find that the Benjamini Hochberg procedure [Benjamini and Hochberg, 1995] for false-discovery rate estimation works reasonably well.

2.13.7 Sources of Error

The values in the expectation matrix are estimates based on a finite sample, not the true mean for gene expression. The error introduced by this is discussed by Lause et al., (2021). It is unlikely to create a significant problem for genes where the total UMI is relatively large (i.e. where the number of cells times the observed UMI per cell is above a few hundred). However, since it is the total number of observations, and not the observations per cell, that dictate the

uncertainty in sample estimates, we may expect that the fewer the cells, and the more shallow their sequencing, the greater the magnitude of this problem will be. In practice we find that, with typical sequencing depths, working with samples of at least 500 cells is advisable.

For genes in which total UMI is small, as mentioned above, the errors in estimation can be sufficient to shift the Cornish Fisher polynomial equations so that curves that should have crossed the abscissa near the origin (indicative of a high p-value) may fail to do so, curving away and returning to cross it elsewhere; this typically produces a highly erroneous p-value, which can usually be spotted on inspection of the raw gene expression vectors. To eliminate such cases, in all instances where the two genes being tested for correlation are both expressed at very low levels (a threshold of 84 total UMI across all cells is used), Cornish Fisher polynomials are tested for the presence of an extremum near the origin, and those gene-pairs are rejected if they display one.

In BigSur the distribution we use for gene expression noise is log-normal; furthermore the value of c for that distribution is taken to be constant for all genes. While theoretical arguments and empirical data support the use of the lognormal distribution [Bahar Halpern, 2015; Beal 2017], some theoretical arguments suggest that different distributions may be more appropriate for some genes, but it seems unlikely that the precise choice of distribution matters greatly to the results obtained by BigSur. Of greater consequence is the assumption of a common value of c, the coefficient of variation of this distribution, for every gene, when the value of c is likely to be different for different genes. In future, with greater knowledge, it may be possible to introduce gene-specific corrections to c. In the meantime, it is helpful to know that the choice of c only impacts the statistics for genes with relatively high expression level, among which correlations are most easily detected.

In determining the moments of PCC' under the null hypothesis, we use moment values for the inverse of the square root of the modified corrected Fano factors that are calculated by simulation and interpolation, using the actual sequencing depth distribution among the cells in the data set. As the simulations involve finite samples, there will be a small amount of error in these values. Furthermore, when we obtain the moments of PCC' by multiplying these moments by the moments of the sums of products of pairs of Pearson residuals, we are implicitly assuming that the Fano factors are independent of the distribution of the products of pairs of Pearson residuals. While technically this is not the case, the fact that the Fano factors average over a very large number of (squared) Pearson residuals makes them effectively independent. When cell numbers are small, however, this may cease to be true (providing a second reason not to apply BigSur to datasets with small numbers of cells). As discussed above, the moments of the inverse of the square root of the modified corrected Fano factors generally turn out to be quite close to 1, unless gene expression levels are low; in practice this means that correlations found to be significant when these moments are ignored tend to be almost identical to those obtained when they are included, suggesting that accuracy in determining their values is not very important.

2.13.8 References

Bahar Halpern, K., Tanami, S., Landen, S., Chapal, M., Szlak, L., Hutzler, A., Nizhberg, A., Itzkovitz, S., 2015. Bursty gene expression in the intact mammalian liver. Mol Cell 58, 147-156.

Beal, J., 2017. Biochemical complexity drives log-normal variation in genetic expression. Engineering Biology 1, 55-60.

Benjamini, Y., Hochberg, Y., 1995. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. J. R. Statist. Soc. B 57, 289-300.

Bulmer, M.G., 1974. On fitting the Poisson lognormal distribution to species-abundance data. Biometrics 30, 101-110.

Cornish, E.A., Fisher, R.A., 1938. Moments and cumulants in the specification of distributions. Review of the International Statistical Institute 5, 307-320.

Lause, J., Berens, P., Kobak, D., 2021. Analytic Pearson residuals for normalization of single-cell RNA-seq UMI data. Genome Biol 22, 258.

CHAPTER 3: STATISTICALLY PRINCIPLED FEATURE SELECTION FOR SINGLE CELL TRANSCRIPTOMICS

3.0 Foreword

Though the focus of this thesis revolves around the development use of correlation statistics, we realized early on that modified-corrected Pearson residuals could potentially be used to improve the statistics of other metrics. Here, we explore one of those possibilities. As we alluded to in the previous chapter, we developed the modified corrected Fano factor as a feature selection statistic. This work similarly resulted in a paper in BMC Bioinformatics, reproduced here, where I appear as the second author. I was a part of the methods development process and developed and analyzed the synthetic scRNAseq data that appears in the paper. Emmanuel Dollinger, the first author, performed all of the analyses on real data and code optimization.

3.1 Abstract

Background: The high dimensionality of data in single cell transcriptomics (scRNAseq) requires investigators to choose subsets of genes ("feature selection") for downstream analysis (e.g., unsupervised cell clustering). The evaluation of different approaches to feature selection is hampered by the fact that, as we show here, the difficulty of feature selection can vary greatly, depending on the dataset being analyzed.

Results: For routine cell type identification, even randomly chosen features can perform well, but for cell type differences that are subtle, both number of features and selection strategy matter strongly. We present a simple feature selection method grounded in an analytical model that allows for interpretable delineation of how many and which features to choose, facilitating identification of biologically meaningful rare cell types. We compare this method to default methods in scanpy and Seurat, as well as SCTransform, showing how greater accuracy can often be achieved with

surprisingly few, well-chosen features.

Conclusions: Feature selection is a critical step in scRNAseq for downstream analyses. We explore the pitfalls that can arise from incautious feature selection and present a statistical method to facilitate improved outcomes.

3.2 Keywords

Fano factor; clustering; cell type; cell state; rare cell identification; single cell transcriptomics; feature selection

3.3 Background

Single cell RNA sequencing (scRNAseq) measures the transcriptional profiles of individual cells, enabling cell type classification, lineage inference, and elucidation of experimental differences in both gene expression and cell type abundance [1-7]. As cell type identity is generally not known *a priori*, unsupervised clustering is commonly used to group transcriptomically similar cells. As a first step, genes ("features") considered most likely to be markers of cell type or state are selected and used for subsequent dimensionality reduction and clustering [8].

In general, limiting features in high dimensional data to those that are most informative for downstream applications improves interpretability, increases computational efficiency, prevents overfitting, and improves the performance of clustering algorithms [9]. However, in scRNAseq, there is no accepted definition of "most informative." Most widely used algorithms calculate gene expression variation across cells, but they differ significantly in how variation is measured, as well as how the appropriate number of features is determined [10-19]. Although some comparisons have been made of the effects of different feature selection methods on clustering [13, 20, 21], it occurred to us that performance of a selection method will likely depend on aspects of the dataset

to which it is applied. These could include how many cell types are present, their relative abundance, and the number and magnitude of gene expression differences between cells of different types. To our knowledge, the interaction of these factors with different methods for ranking and selecting genes has not been explored in a systematic way.

Here we show that, for datasets in which the desired goal is to cluster relatively abundant cells that differ greatly in gene expression, how features are selected is almost irrelevant. Even random sets of genes, if large enough, tend to perform nearly as well as algorithmically chosen features. In more demanding situations, such as when clustering cells with similar gene expression, we identify cases in which both the method of feature selection and the number of features selected markedly influence clustering outcomes.

This led us to revisit the question of how best to identify genes for which cell-to-cell variation is greater than otherwise expected. Starting with an analytical model for the distribution of un-normalized scRNAseq data, we developed a method to quantify the magnitude of biologically meaningful gene expression variation and to estimate the probability that such variation occurs by chance. This method, which we call BigSur (Basic Informatics and Gene Statistics from Unnormalized Reads), provides a theoretical framework for scRNAseq data analysis which enables both feature selection and the inference of gene regulatory networks from gene-gene correlations (the latter use having been outlined elsewhere [22]). Here we show that using BigSur for feature selection enables identification of biologically relevant groups of cells while minimizing loss of discriminatory power due to the use of excessive numbers of features.

3.4 Results

3.4.1 For common tasks, random sets of genes can perform nearly as well as algorithmically-chosen features.

Existing feature selection algorithms often choose up to several thousand genes for use in cell clustering (e.g. [19, 23-25]). As this can represent a substantial fraction of the expressed genome, one is left to wonder how many of these features are necessary, and whether the number chosen is particularly important.

In common scenarios, in which cell types of interest are relatively abundant and well separated in gene expression space, we find that features chosen at random often perform nearly as well as those selected by popular algorithms. We illustrate this by clustering the 10k cell PBMC (peripheral blood mononuclear cell) dataset from 10x Genomics, which is commonly used for methods evaluation. Initially, we used scanpy's default feature selection method (highly variable genes, "HVGs"), with cutoffs and parameters set to their default values. HVGs bins genes by normalized mean expression, calculates a z-score for each gene's Fano factor (variance / mean) relative to the other genes' Fano factors in the same bin, and ranks genes by those z-scores ("normalized dispersion") [18]. Using the 2,307 genes chosen by HVGs, cells were clustered according to the default pipeline: calculating the first 50 principal components (PCs) using principal component analysis (PCA), creating a nearest neighbor graph in PCA space, and assigning clusters on that graph using the Leiden algorithm. Clusters were then labeled based on the expression of known marker genes. As expected, cell groups were well separated on a UMAP plot (Figure 1A, left; see also Figure S1).

For comparison, we used an equal number of genes (2,307) chosen at random as features and performed the same task. As shown in Fig. 1A, right, the same cell types grouped well and separated similarly from each other, with the one exception that some of the CD4+ and CD8+ T cells were intermixed.

We next asked how many random genes were needed to cluster cells correctly, and how

repeatable the results were. Since, in this case, labels are already assigned to each cell, we can treat this question as a supervised problem. We therefore set as ground truth the cell types shown in Figure 1A and assessed the ability of a straightforward classifier (a linear support vector machine, or SVM) to identify the cell types from the 50 top PCs of randomly selected genes (see methods). We first trained a classifier on 2,307 randomly selected genes (the number of genes selected by default by HVGs) and found that, with the exception of one of the doublet clusters, the classifier correctly identified at least 70% of each cell type (Figure S2A).

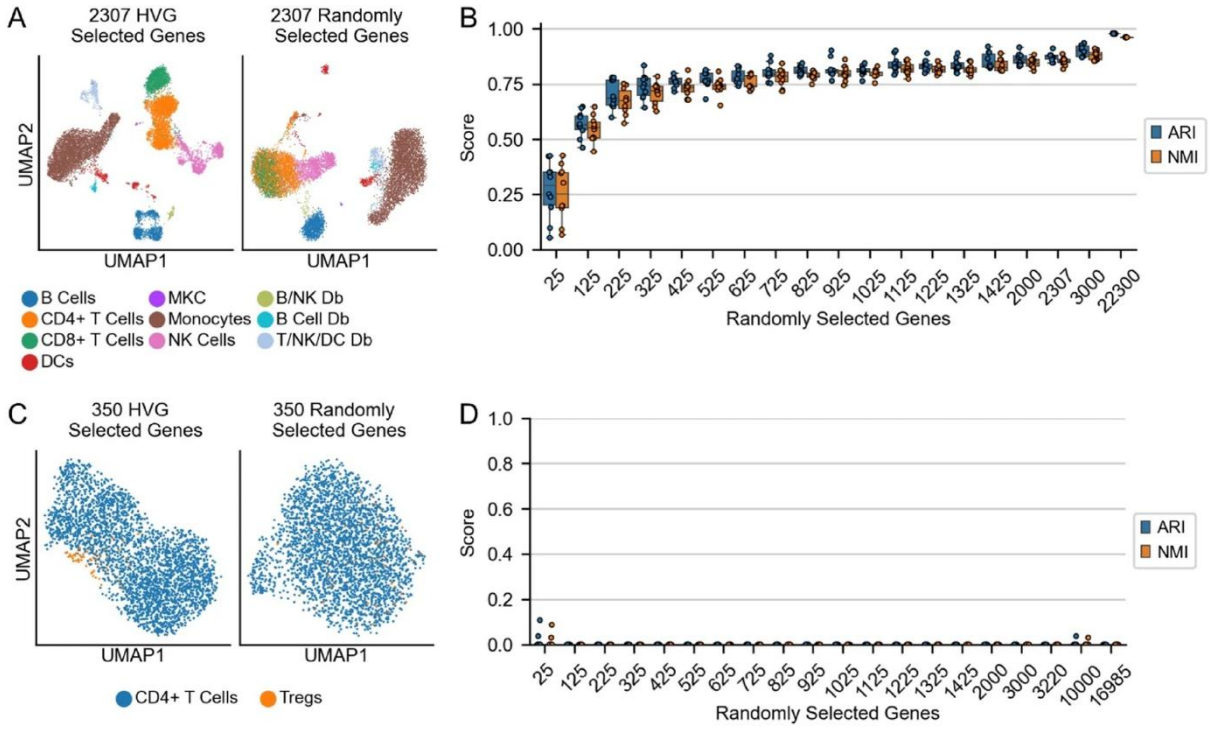


Figure 1. Performance of randomly selected genes as features.
A. UMAPs of PBMCs calculated using either HVG-selected genes (left) or the same number of randomly selected genes (right). B. Adjusted Rand index (ARI) and normalized mutual information (NMI) of prediction of cell type as a function of number of genes selected. The predictions were generated by a linear Support Vector Machine (SVM), trained on the top principal components (PCs) calculated on randomly selected genes. Ten sets of random genes were selected at each step. C. UMAP of CD4+T cells, including the T regulatory cell (Tregs) subset, calculated using either HVG-selected genes (left) or same number of randomly selected genes (right). D. ARI and NMI of Treg identification, as a function of number of randomly selected genes. The predictions were generated by a linear Support Vector Machine (SVM) as in panel B. 20 sets of random genes were selected at each step.

We then asked how the SVMs would perform with varying numbers of selected genes. We

first varied the number of genes selected by HVGs, retraining the SVM at each step, and concluded that, as long as at least 725 HVGs-selected genes are used, either the adjusted Rand index (ARI) and the normalized mutual information (NMI) – two common clustering metrics – will be above 0.95 (Figure S2B). We then performed the same task, varying the number of randomly selected genes from 25 to 22,300 (all genes present in the dataset), testing 10 samplings at each number and retraining the SVM at each step. As the numbers of randomly selected genes increased, both the ARI and NMI increased quickly to an ARI of 0.8 at 725 random genes and an NMI of 0.8 at 925 random genes (Fig. 1B). Both scores continued to rise up to a maximum of 0.98 and 0.96 for ARI and NMI, respectively, when using all genes as features. Across the 10 trials, the variance in the ARI and NMI scores decreased as the number of genes increased but was already relatively low by 525 random genes (Fig. 1B).

To verify that the number of cells belonging to each cell type didn't substantially influence these results, we also downsampled the dataset to only contain 1,000 cells of each cell type and again trained SVMs on increasing numbers of randomly selected genes. We again found that ARI and NMI scores of the predictions increased to 0.77 and 0.79, respectively, with 725 randomly selected genes, and continued to rise up to 0.97 and 0.96 with all genes (Figure S2C-D).

These results suggest that almost any random set of genes of size greater than a few hundred is potentially sufficient to correctly classify groups of cells that are well separated in gene expression space. Although we are not aware of this observation having been explicitly noted before, it supports the view that the effective dimensionality of gene expression is far lower than the number of expressed genes [26-28]. In other words, any random sample of reasonable size would be expected to include a substantial number of genes that correlate with all major patterns of gene expression.

3.4.2 Some tasks are sensitive to choice of feature selection algorithm

Given the results above, the task displayed in Fig. 1A is clearly not suitable for assessing the performance of any feature selection algorithm, as the lack of an algorithm (using all genes) and a trivial algorithm (random gene selection) perform reasonably well.

This led us to consider more challenging tasks, for example one involving more subtle gene expression differences. We therefore subset the PBMC dataset to just the CD4+ T cells and performed unsupervised clustering. Using 350 features selected by HVGs, we could identify a FOXP3+ T regulatory cell (Treg) cluster of approximately 1.8% (53/2,908) of the cells (Fig. 1C, left; Treg identification is shown in Figure S2E-F). Tregs are a well described CD4+ T cell subtype, regulating immune responses in many biological systems [29]. In this case, using an equal number of random genes as features, the Treg population was not identifiable (Fig. 1C, right). There was a small population of cells, visible in the lower left quadrant of the UMAP calculated using randomly selected genes, that separated from the main group, but this separation seemed to be driven by the substantially lower sequencing depth of those cells, rather than significant differences in gene expression (Figure S2G).

Even using much larger numbers of genes—up to all 16,985 expressed genes–and testing 20 random gene sets for each sample size, ARI and NMI scores remained close to zero (Fig. 1D). Thus, in a sufficiently difficult task, HVGs performed much better than random.

3.4.3 Number of features can influence success or failure in unpredictable ways

The fact that random gene selection failed to identify Treg cells, even when the entire expressed transcriptome was used, emphasizes that a good feature selection method must not only include enough genes that are predictors of cell types or states, but also exclude enough genes that

are not. Indeed, even with features ranked by HVGs, the use of too many features substantially decreased calculated ARI and NMI scores (Figure S2H). Such behavior reflects a well-known phenomenon in machine learning, in which the accuracy of a classifier initially rises with increasing number of features and then falls [30]. This happens because, in high dimensions, inclusion of features that have low predictive power can degrade the effects of data with high predictive power (i.e., the "noise" can overwhelm the "signal") [31].

Because of this, it is important that feature selection algorithms for scRNAseq not only order genes by their utility as features, but also decide how far down the list one should go, i.e., the number of features to use when performing unsupervised clustering. Currently popular algorithms typically choose feature number based either on arbitrary variability cutoffs or on calculations that depend upon arbitrarily adjustable hyperparameters.

To assess how important these choices are, we selected three clustering tasks: subclustering CD4+ T cells (as in Fig. 1C-D); CD8+ T cells (also from the PBMC 10K data set), and human retinal amacrine cells [32]. We selected a particular cluster of interest in each dataset, shown in Figure S3A – C, with the aim of determining how the number of features used influences the Leiden clustering algorithm's ability to isolate the cell state of interest. We therefore varied the number of features by starting from those considered by HVGs to be most highly variable and adding an increasing number of features from further down the list. To assess clustering performance, we defined a *purity score*: the fraction of target cells within the cluster containing the most target cells (see methods).

As shown in Figure 2, the outcomes of clustering differed markedly among the three tasks. With amacrine cells, a population of SLC12A7-expressing cells could be cleanly identified with either 150 or 3,424 features, the latter representing HVG's default selection (Fig. 2A). In this case,

purity was relatively insensitive to the number of features used (Fig. 2B). Since the Leiden clustering algorithm requires a starting seed, for each set of features, we used 50 randomly chosen starting seeds, and calculated purity scores for each case (Figure 2C) [33].

With CD8+ T cells, we identified a subpopulation of memory T cells (a CCL5+ population equaling 9.3% of the total) using the top 100 HVG features (Fig. 2D). Here, performance degraded markedly when larger numbers of features were used and sometimes declined to very low levels (Fig. 2E). Indeed, using the default number of features suggested by HVGs (2,744; red circle in Fig. 2E), the ability to identify memory T cells as a distinct cluster was lost. We found that the observed "choppiness" in panel 2E – where purity scores jump from high to low with the addition or subtraction of just a few features –reflected a high sensitivity in this particular dataset to the choice of random starting seed used by the Leiden clustering algorithm (Figure 2F).

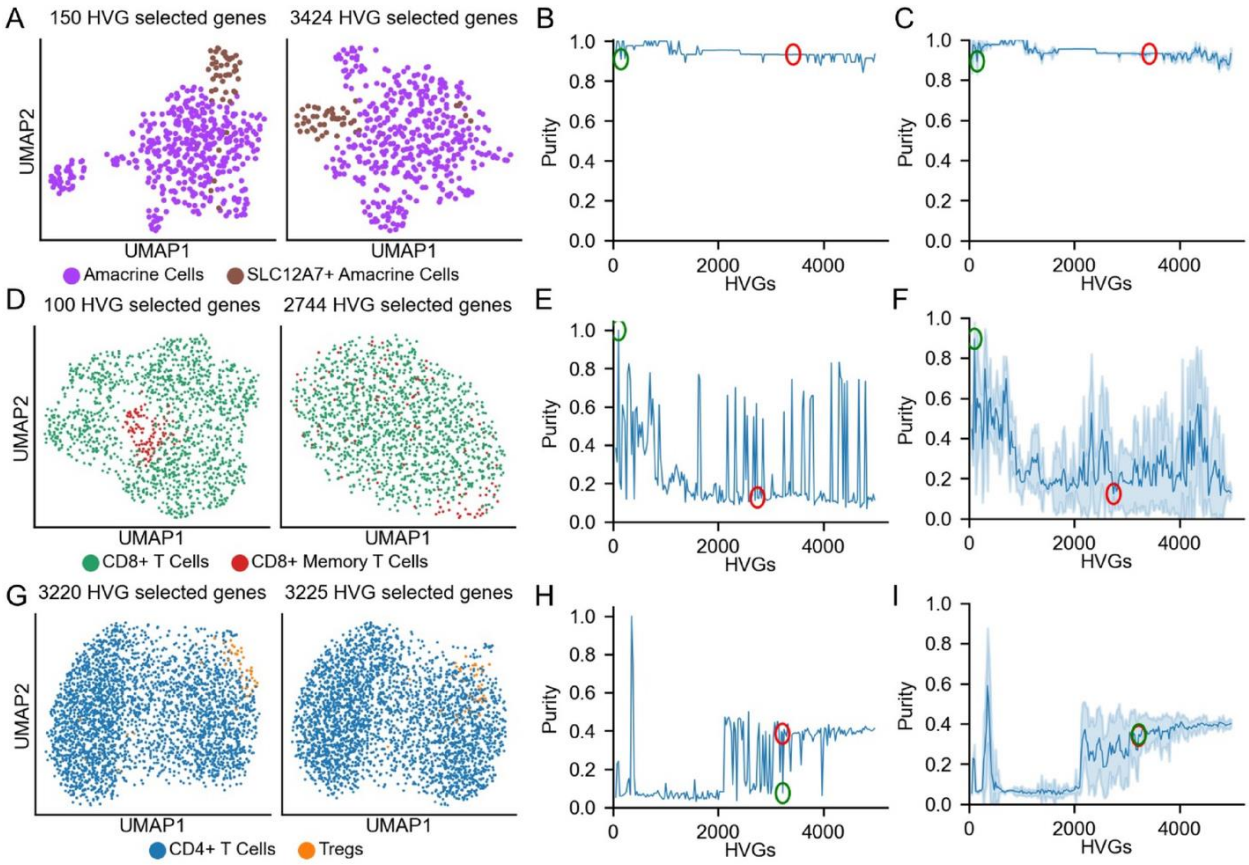


Figure 2. Numbers of features and feature selection method influence clustering.
A. UMAPs of amacrine cells and SLC12A7-expressing amacrine cells using either 150 HVGs (left) or the default number of HVGs (3,424; right). B. Purity score (see main text) for SLC12A7+ cells using different numbers of features selected by HVGs. Green and red circles show results using 150 and 3,424 genes, respectively. C. Mean and standard deviation of purity scores for SLC12A7+ cells, using 50 randomly selected starting seeds for the Leiden clustering algorithm. D. UMAPs of CD8+ T cells and memory CD8+ T cells using 100 HVGs (left) and the default number of HVGs (right). E. Purity score of memory CD8+ T cells with increasing number of HVGs. Green and red circles mark enrichment scores at 100 HVGs and the default number of HVGs (2,744), respectively. F. Mean and standard deviation of purity scores of memory CD8+ T cells using 50 different Leiden starting seeds, as in C. G. UMAPs of CD4+ T cells and Regulatory T cells (Tregs) using either 3,220 genes (left) or default number of HVGs (right, 3,225). H. Purity score of Tregs as in panels B and E. I. Mean and standard deviation purity scores of 50 randomly selected Leiden starting seeds, as in panels C and F.

Finally, we turned again to the CD4+ T cells. The Treg population previously discussed in Figure 1C mostly grouped together when the default number of 3,220 HVGs was used (Fig 2G-H, Fig. 2H red circle), but adding in even 5 more features destroyed the ability to identify this cluster (Fig. 2H, green circle). We also noted that the purity score of the CD4 dataset, when using the

default starting seed of the Leiden algorithm, was markedly higher at 350 genes (the "spike" visible in panel H) than that of the average purity score with the same number of genes. This suggests that the cells in the Treg cluster identified in Figure 1 were clustered together at least partially due to a lucky choice of starting seed.

As one might expect, variability in clustering success due to changes in the number of selected features has a substantial effect on common downstream tasks, such as differential expression analysis. To illustrate this, we first clustered the CD8+ T cells using 100 HVGs and determined the differentially expressed genes (DEGs) using the Benjamini-Hochberg FDR-corrected *p*-values of the Mann-Whitney-U test. We then repeated this task using the default number of HVGs (2,744) and compared the outcomes. Changing the number of HVGs changed both the number of clusters identified and the genes identified as DEGs (Figure S4). In particular, the expression of CCL5, which marked CD8+ memory T cells when using 100 HVGs, no longer specifically marked any cluster when using the default number of HVGs. In addition, the magnitudes of differential expression tended to be larger in the clusters assigned using 100 HVGs than in those assigned using the default number of genes.

Taken together, the results in Fig. 2 indicate that, for tasks for which having an algorithm for feature selection matters, selecting the right number of features can affect whether cells are correctly clustered together. Too many features can be as harmful as too few, and for some datasets, even small changes in feature number can have dramatic consequences. Importantly, feature numbers chosen by commonly used procedures under default conditions could lead to systematic misclassification. We therefore asked whether there might be a principled way to optimize the joint tasks of finding features and determining how many to use.

3.4.4 A statistical approach to feature selection

Common feature selection methods tend to choose genes by the degree to which they are variable within a dataset, with differences among methods typically involving the way data are initially transformed, the way variability is defined and measured, and how the appropriate number of features is determined [19, 23, 24]. Ideally, one would want to select features that are more variable than expected by chance, but this requires knowledge of the expected gene expression distribution when all cells are biologically equivalent (i.e., the only sources of variability being technical and biological noise), which we refer to as the "null" distribution. Because there has long been a lack of general agreement on the appropriate null distribution for scRNAseq data, it has become common to empirically determine an appropriate model from data, sometimes with additional modifications to produce a better fit (e.g., SCTransform) [17-19].

We chose the alternative approach of constructing a null distribution analytically, based on assumptions about how scRNAseq data arise. Specifically, we assume that biological noise—random fluctuations in transcript numbers in identical cells—follows a log-normal distribution, which agrees with both theoretical predictions and empirical observations in mammalian cells [34, 35]. In addition, we assume that the technical noise associated with cell preparation, library preparation, sequencing, etc., can simply be approximated as sampling noise, i.e., by the Poisson distribution. This agrees with a number of recent observations and arguments in the scRNAseq literature.

We thus write the following model:

$$x_{ij} \sim Poisson\left(Log\text{-}Normal(\mu_j, c_j)\right)$$

where μ_j is the expected number of transcripts of gene j, c_j is the coefficient of variation of that gene, and x_{ij} is the observed counts (UMIs) detected for gene j in cell i. Note that we parametrize

the log-normal distribution in terms of its actual mean and coefficient of variation, and not the mean and coefficient of variation of the underlying normal distribution from which it may be derived.

To guide the use of this model in analyzing real-world data, we first generated simulated data reflective of a situation in which two cell types or states are present, in equal proportions, and differ only in the expression levels of a fixed number of genes. Specifically, for 2,000 cells and 15,000 genes expressed at a range of levels, we generated data using a Poisson log-normal model with an underlying coefficient of variation of gene expression (c_j) of 0.7 for each gene (Fig. 3A; see methods). For a subset of genes, the two cell types were made "truly variable", i.e., different expression means were used for the two cell types, whereas for the rest of the genome, the means for the two cell types were the same. For simplicity, in Fig. 3 we did not vary the sequencing depths among cells (a common problem in scRNAseq, which we address later on).

When we generate datasets in which the number of truly variable genes is between 100 and 1,000, we observe a threshold above which randomly selected features will enable separation of the two cell types (Fig. 3B). The threshold depends on how many features are truly variable and how many random genes are selected as features, but tends to occur when the number of truly variable genes expected to be found among the random features ranges from 30-150 (values along the diagonal in Figure 3B). As the number of randomly selected features increases, the fraction of truly variable features necessary to separate the two cell types also increases. This agrees with previous results arguing that separating distributions in high-dimensional space depends on including informative features ("signal") while excluding non-informative features ("noise") [36].

Ideally, an algorithm for identifying features should seek to compare the observed variance of a feature with the variance expected under the null hypothesis, i.e., when features are not more

variable than expected. For data that are Poisson-distributed, variance is equal to the mean, so the ratio of observed variance to mean (also known as the Fano factor) equals 1 under the null hypothesis. Hereinafter we use ϕ_j to stand for the Fano factor of gene j, σ_j^2 the observed variance of gene j, and μ_j the observed mean. Thus, $\phi_j \equiv \frac{\sigma_j^2}{\mu_j}$.

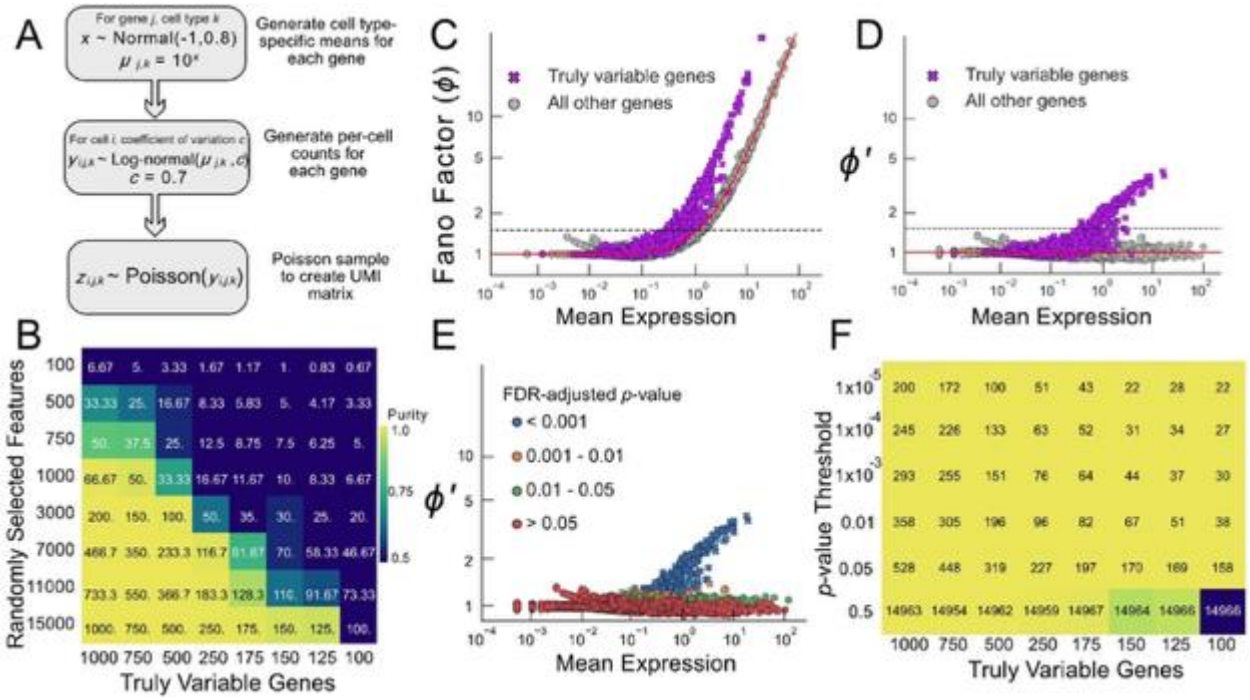


Figure 3. BigSur identifies truly variable features in simulated data.
A. Simulated data generation pipeline (see methods). B. Average purity score of simulated datasets, clustered using varying numbers of randomly selected genes. Each purity score is the average of ten rounds of random feature selection. The numbers on each tile indicate the expected number of truly variable genes to be found among the selected features (the proportion of genes that were selected of the total number of genes multiplied by the number of truly variable genes). C. Fano factor ϕ of simulated data with 1,000 truly variable genes plotted against their means. Orange crosses indicate genes that were truly variable. Blue points are genes that were selected from the same distributions for the two cell types. The red line depicts the expected relationship, under the null hypothesis, between the Fano factor and mean expression. The dashed line at ϕ=1.5 illustrates the large number of non-truly-variable genes that would be chosen if the Fano factor is used to select features. D. Observed relationship between the modified Fano factor (ϕ') and mean expression. The red line again depicts the expected relationship under the null hypothesis. E. Points from panel D colored by their FDR-adjusted p-value. Markers are as in panels C-D. F. Average purity scores (color scale as in panel B) for the dataset shown in panel B for clusters obtained using features selected by BigSur at different p-value thresholds. Numbers overlaid on each tile indicate the number of features selected by BigSur in each case.

Fig. 3C shows the distribution of observed Fano factors in simulated data generated as in Fig. 3B, in which the number of truly variable genes was 1,000. Although one should only expect to observe ϕ>1 for truly variable genes, many non-variable genes also display large values of ϕ; in particular this is true for all of the highly expressed genes. This relationship, previously observed in real data (e.g., [19]), reflects the fact that the variance of a Poisson sample from a distribution (hereafter referred to as a Poisson compound distribution) is equal to the variance of the Poisson distribution (i.e., the mean) plus the variance of that distribution. As we may equivalently express variance as the coefficient of variation times the mean squared, i.e., $(c\mu)^2$, a modified Fano factor expressed as:

$$\frac{\sigma_j^2}{\mu_j+(c_j\mu_j)^2} = \frac{\sigma_j^2}{\mu_j\left(1+c_j^2\mu_j\right)} \qquad (1)$$

should have an expectation value of 1 for a compound Poisson distribution. Indeed, Figure 3D shows that, when applied to simulated data, the modified Fano factor is indeed centered around 1 for non-variable genes, and exceeds 1 for variable genes, as long as the latter are sufficiently highly expressed.

Although this modification to the Fano factor clearly facilitates the identification of truly variable genes in the simulated data in Fig. 3, it does not correct for an additional problem encountered in real scRNAseq data, which is that sequencing depth often varies widely (by orders of magnitude) between cells. Simply normalizing observed counts to the total number of reads in each cell is inappropriate—the expected distribution of counts across cells under the null hypothesis is no longer knowable—but sequencing depth variation can be appropriately accounted for by correcting each gene's Pearson residual [37]. The Pearson residual, a measure of deviation from the mean, is defined, in cell *i* and gene *j*, as

$$P_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\mu_j}} \quad (2)$$

where μ_j is the mean expression over all cells. The Fano factor may in fact be expressed as an average of squared Pearson residuals:

$$\phi_j = \frac{1}{n-1} \sum_{i=1}^{n} {P_{ij}}^2 \quad (3)$$

Lause et al. noted that, when sequencing depth is variable, the expectation value for ϕ_j can be made equal to 1 if the term μ_j in the Pearson residual of every cell *i* is replaced by μ_{ij} as follows [37]:

$$\mu_{ij} = \frac{\sum_j x_{ij} \sum_i x_{ij}}{\sum_{ij} x_{ij}}$$

If we further modify (3) by dividing by $\sqrt{\left(1 + c_j^2 \mu_{ij}\right)}$, along the lines described above, we obtain ${P'}_{ij}$, which we refer to as a "modified corrected Pearson residual":

$${P'}_{ij} = \frac{x_{ij} - \mu_{ij}}{\sqrt{\mu_{ij}\left(1 + c_j^2 \mu_{ij}\right)}} \quad (4)$$

as well as a "modified corrected Fano factor", ϕ'_j:

$$\phi'_j = \frac{1}{n-1} \sum_{i=1}^{n} {{P'}_{ij}}^2 \quad (5)$$

Note that ${P'}_{ij}$ and ϕ'_j are calculated from raw, unnormalized counts, as it is the μ_{ij}, and not the x_{ij}, that undergo correction for sequencing depth variation.

Because we expect $\phi'_j = 1$ under the null hypothesis, observing $\phi'_j > 1$ should identify features likely to be truly variable. To utilize those observations in a principled way, however, requires knowing the probability of observing any value of ϕ'_j under the null hypothesis (i.e., when

the cells are all of the same state). While it is generally not possible to obtain an analytical expression for the full distribution of ϕ'_j, we can estimate it, by constructing an arbitrary number of moments of that distribution from the moments of the distributions of the individual P'_{ij}, which we can in turn construct from the moments of the Poisson-Log Normal distribution parametrized by μ_{ij} and c_j in each cell [22]. Given a sufficient number of moments, procedures exist for estimating tail probability densities [38]. In this manner, one can associate any set of gene expression values with a *p*-value, i.e., the probability of observing it by chance. Given a set of *p*-values, one can then identify a *p*-value threshold for any level of false discovery [39]. In Fig. 3E, the data displayed in panels C-D are colored by the false discovery rate (FDR) threshold interval into which their *p*-values fall. It is immediately apparent that, for these data, few non-variable genes display FDR-adjusted *p*-values less than 0.05, and none display values less than 0.001.

We have named the algorithm for jointly computing modified corrected Fano factors and their *p*-values *BigSur*, which stands for Basic Informatics and Gene Statistics from Unnormalized Reads. Fig. 3F applies BigSur to all eight datasets (columns) in Fig. 3B, reporting both the number of features that satisfy different significance thresholds, and their success in clustering the two cell types (the latter represented using the same color scheme as in Fig. 3B). A surprisingly small number of features suffice to produce good clustering (Figure S5B). For example, in the dataset containing only 100 truly variable genes, as few as 22 statistically significant features recovered pure clusters, whereas no number of random features (from 100 to 15,000) could do so. This reflects the ability of BigSur to recover a large fraction of true positives while minimizing false discovery. Although other feature selection methods, such as HVGs, could also successfully identify the two cell types in these simulated data (Figure S5), the default number of features selected to use in clustering was often much higher. For example, on the same dataset, at its default

dispersion threshold of 0.5, HVGs selected thousands of genes, whereas we found that just using the top 366 of those genes was adequate to produce essentially the same results (Figure S6).

The only hyperparameter used by BigSur is c_j, the underlying coefficient of variation of gene expression (i.e. the actual biological variation among equivalent cells). While this number may, in principal, differ for each gene, experimental studies suggest it does not vary greatly, and we provide a method here to estimate a consensus value of c_j (which we hereafter simply call c) from a full gene expression dataset (see methods).

It is worth noting that the modification and correction of Pearson residuals may also be generalized to higher order statistics, such as correlation coefficients, and that this can be leveraged to more accurately assess gene-gene correlations [22].

3.4.5 BigSur helps identify rare, biologically relevant cell states

We next examined the performance of BigSur on experimental data, focusing on cases in which the presence of rare cell types, or subtly different cell types, might be expected to pose challenges for clustering. First, we chose the CD4+ T cell subset, which contains previously identified Treg cells, from the 10k PBMC set. As shown in Figure 4A, the (uncorrected) Fano factors associated with gene expression rose sharply with expression level, similar to what was observed in simulated data (Fig. 3C). In contrast, the modified corrected Fano factors shown in Figure 4B (using a fitted $c = 0.25$) were generally mean-independent, with most genes displaying a value of ϕ' near 1. These results suggest that the statistical properties of simulated data resemble those of real data; they also suggest that, in this dataset, the patterns of expression of most genes are consistent with not being truly variable. For a small subset of genes, however, values of ϕ' up to 10 were observed, and most of those above 1.5 were associated with FDR-adjusted p-values

less than 0.05. Overall, 156 genes were characterized by ϕ' >2 and p <0.05 (Fig. 4C). Among these was *FOXP3*, considered a marker for Tregs.

To investigate which statistic, ϕ' or the p-value, played a greater role in enabling identification of Tregs as a unique cluster, we carried out Leiden clustering using features identified at different thresholds for these parameters and calculated a *FOXP3*-enrichment score for each cluster produced. The enrichment score was calculated by dividing the mean normalized expression of *FOXP3* in each cluster by the mean expression of *FOXP3* in the dataset. As shown in Fig. 4D-E, use of an adjusted p-value threshold of 0.05 was particularly important to achieve good cell separation, i.e., the inclusion of non-statistically significant features appears to be especially detrimental to rare cell type identification (Tregs are only 1.8% of the cells in this dataset). Among statistically significant features, a cutoff based on ϕ' appeared to have little impact, until that cutoff became so high that the absolute number of selected features became very small—in the vicinity of 50-150 genes (Fig. 4E). We show the clusters and their enrichment scores when using a ϕ' cutoff of 3 and p-value cutoff of 0.05 in Figure 4F.

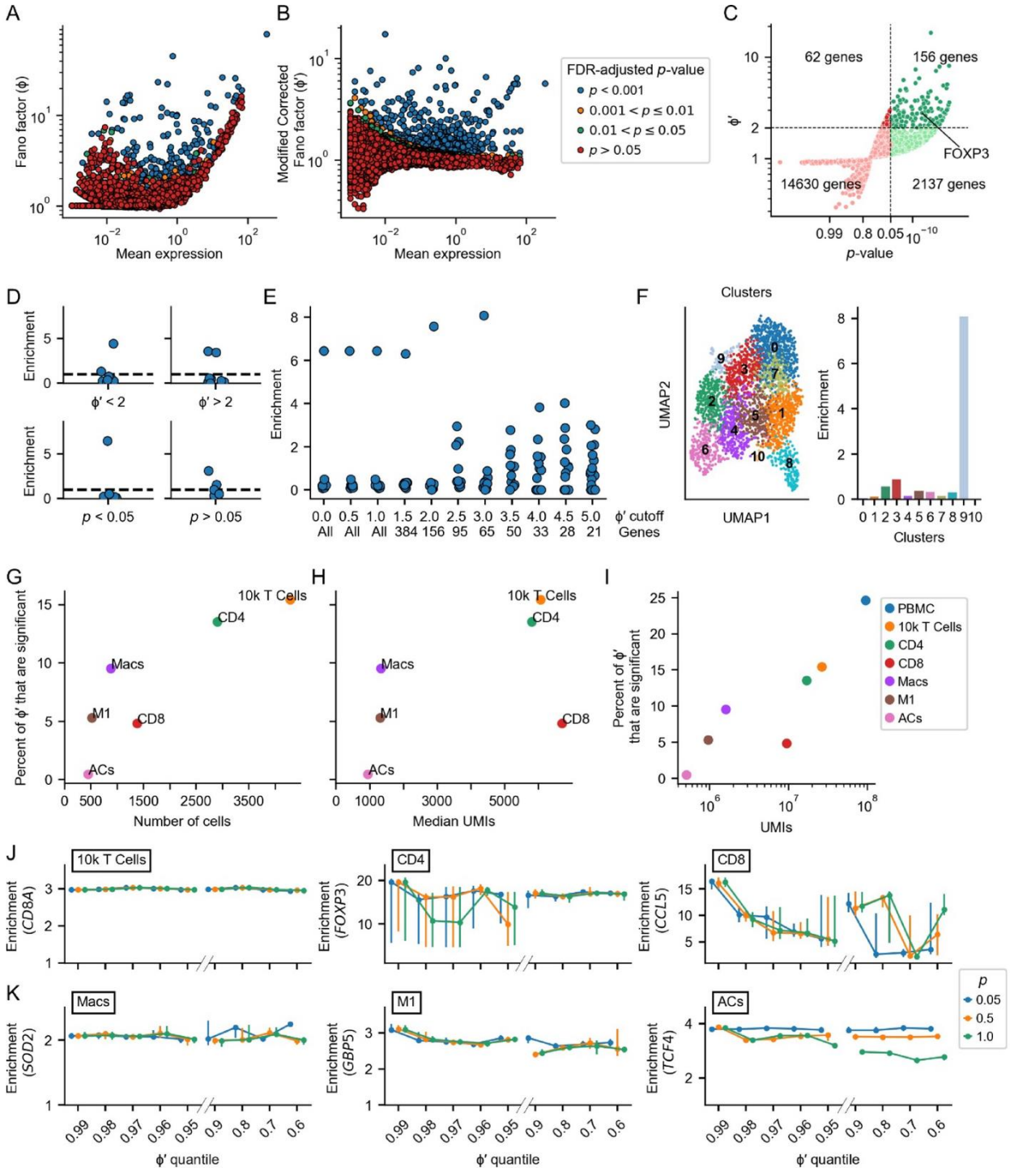


Figure 4. Clustering performance using values and significance levels of modified corrected Fano factors. The Fano factor ϕ (A) and the modified corrected Fano factor ϕ' (B) were calculated for all genes in the CD4+ T cell dataset and are plotted as a function of gene expression level and colored according to the p-value of ϕ'. C. Values of ϕ' and their p-values for genes in the CD4+ T cell are plotted against one another. D-E. Enrichment scores (see main text) for FOXP3 after Leiden clustering using features selected using different p-value and ϕ' cutoffs. Each point represents an individual cluster. Panel D shows the enrichment scores using solely a ϕ' threshold (top) or solely an adjusted p-value threshold. Panel E

displays the enrichment scores as the ϕ' is increased. Only genes with ϕ' p-values < 0.05 were used. F. UMAP of the CD4+ T cell dataset with genes ϕ' > 3 and p-value < 0.05 along with each cluster's FOXP3 enrichment (the enrichment of clusters 0 and 10 were 0.04 and 0, respectively). G-H. For each of six datasets, the percentage of ϕ' that were significant was plotted against the number of cells (panel G) and the median sequencing depth (panel H). I. For the datasets in panel H, as well as the full PBMC dataset (displayed in Fig. 1), the percentage of ϕ' that are significant was plotted against the total number of transcripts in the dataset. J-K. The highest enrichment score for six datasets (displayed in panel F) as a function of features selected using different quantile thresholds for ϕ' (abscissa) and different p-value cutoffs (by color). The Leiden clustering algorithm was run on the selected features using 40 different randomly chosen starting seeds. For each Leiden seed, the enrichment level in the cluster with the greatest enrichment was found. Each point on the plot indicates the median of this value over the different Leiden seed choices, and the bars show the inter-quartile range (25% to 75% of the data).

We next extended this analysis to five other datasets that contain identified cell "subtypes" (shown in Figure S7): the full T cell subset of the 10k PBMC dataset, the CD4+ and CD8+ subsets of those T cells, a macrophage dataset, an M1 subset of those macrophages, and the retinal amacrine cell subset. These datasets span a wide range of at least three characteristics: the fraction of all ϕ' observed to be statistically significant (p<0.05), the number of cells sequenced (Fig. 4G), and the median sequencing depth (median UMI/cell) (Fig. 4H). The latter two of these may be expected to have a strong influence on the statistical power to identify differences in cell types—fewer cells mean fewer observations, and lower sequencing depth means more observations that are zero and thus minimally informative. A simple metric that captures a sense of the combined statistical power of these two observations is the total number of UMI in the entire dataset. Plotting this against the fraction of ϕ' values that is statistically significant (Fig. 4I) shows that, overall, the significant ϕ' fraction varies directly with total UMI. This general pattern is to be expected, as loss of statistical power should correlate with identification of fewer ϕ' as significant, however the magnitude of the effect differs between datasets (for example, the fraction of significant ϕ' values is particularly low for CD8 T cells). A low fraction of significant ϕ' values, in a dataset containing a sufficient number of well-sequenced cells to provide good statistical power, suggests that gene expression differences between cells are especially subtle in this case.

We next examined how well Leiden clustering using features selected by BigSur performed in enriching for markers of known cell subsets. We used enrichment for *CD8A* to assess clustering of CD8+ T cells from mixed T cells, *FOXP3* for Tregs from CD4+ T cells, *CCL5* for memory T cells from CD8+ T cells, *SOD2* for M1 macrophages from mixed macrophages, *GBP5* for a subset of M1 macrophages, and *TCF4* for glycinergic amacrine cells from total amacrine cells (Figure S7; see methods). In each case we considered three different FDR-corrected *p*-value thresholds of 0.05, 0.5 and 1.0 and varied the ϕ' cutoff. To facilitate comparisons between datasets, the ϕ' cutoff was varied by quantile, i.e., 0.6 implies the top 40% of ϕ' values, 0.9 the top 10%, and so on. In each case Leiden clustering was performed 40 times using the features obtained, with a different random seed each time. Shown in Fig. 4J-K are the median and upper and lower quartiles of the enrichment scores obtained from these runs. For the amacrine and T cell datasets, we also performed a full characterization of the effect of using different numbers of BigSur-selected features on the purity score, similar to the analysis done using HVGs in Figure 2. The results when calculating the purity score and the enrichment score were similar (Figure S8).

The outcomes support several of the previous conclusions and suggest guidelines for the use of BigSur in practice. For datasets in which cells are numerous, sequencing is deep, and differences between cell types considerable, clustering succeeds regardless of how features are selected, as in the separation of CD8+ T cells from other T cells or FOXP3+ T cells from other CD4+ T cells, although in the latter case, a too stringent ϕ' quantile cutoff leads to a decrease in the reliability of clustering (Figures 4E, 4J and S8F). In such cases, we suggest selecting genes that are statistically significant (p<0.05) and in the top 10% of ϕ'. In contrast, for deeply sequenced, large datasets in which fewer than 5% of ϕ' values are statistically significant, such as the CD8+ T cell dataset, we find that restricting feature selection by the relative magnitude and

statistical significance of ϕ' can improve performance. For datasets with these characteristics, we suggest only selecting the top 1% of ϕ' that are significant.

In datasets with few cells or shallow sequencing depth, such as the macrophage and amacrine cell datasets, feature selection becomes more challenging. Characteristics of such datasets include having fewer than 150 cells, or less than 3,000 median UMI/cell. In these cases, we suggest selecting the top 10% of the ϕ' that are significant and carefully evaluating the resulting clusters for biological meaning.

3.4.6 BigSur yields comparable or higher enrichment than other common feature selection methods

We compared BigSur to three of the most common methods for feature selection: HVGs (current default in scanpy), FindVariableFeatures (FvF, default in Seurat V5), and SCTransform v2 (SCT) [19, 23-25]. As mentioned earlier, HVGs ranks features by binning genes by mean normalized expression level, calculating z-scores for the Fano factors of genes with respect to their bin, and ranking the genes by those z-scores [18, 24]. FvF fits the relationship between the logarithm of gene variance and log-mean expression and ranks genes by standardized residuals [23]. SCT fits each gene to a negative binomial model with sequencing depth as an explanatory variable, then regularizes the parameters, and ranks genes by residual variance [19, 25].

We compared the performance of each feature selection method using the datasets introduced in Figure 4. For each dataset, we either randomly selected 2,000 genes, or used genes selected by the other feature selection methods using their default cutoffs. We selected BigSur's cutoffs using the guidelines described above; consequently, we used the genes with top 1% significant ϕ' for the CD8 dataset, and genes with top 10% significant ϕ' for the other datasets. We then assigned cells to clusters using the Leiden algorithm with 40 unique starting seeds and

calculated the enrichment score of the marker gene for each dataset (as in Figure 4). For the deeply sequenced datasets with many cells and large differences between cells, i.e., the 10k T Cells and CD4 datasets, the enrichment yielded by BigSur was equal to, or greater than, any other method (Figure 5A-B). For the deeply sequenced dataset with many cells and subtle differences between cells, i.e., the CD8 dataset, the median enrichment yielded by BigSur was more than twice than that of the next best method (SCT), and the median enrichments of other methods performed similarly to random feature selection (Figure 5C). For the three shallowly sequenced datasets, i.e., the two macrophage datasets and the amacrine cells dataset, BigSur again yielded comparable or higher enrichment than the enrichments yielded by other methods (Figure 5D-F).

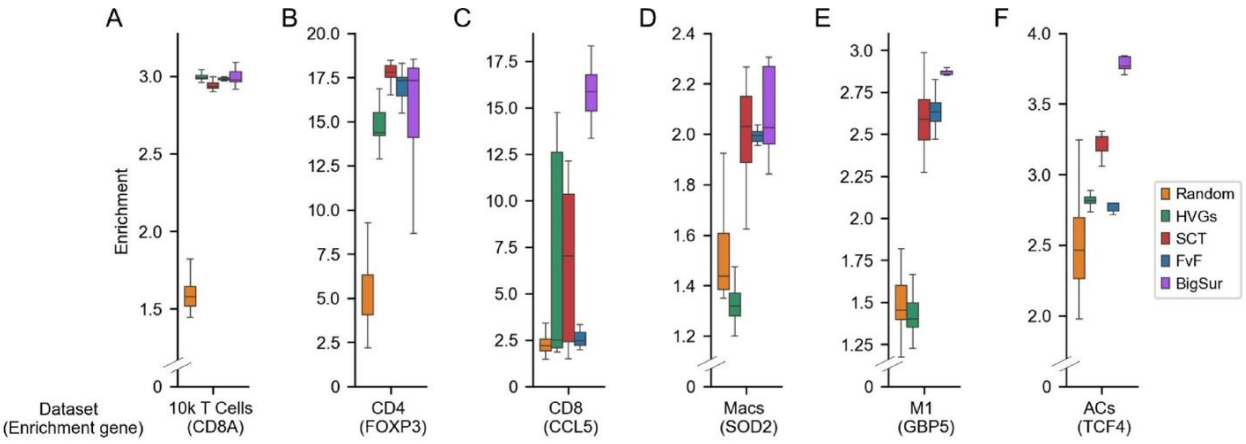


Figure 5. BigSur performs comparably to or outperforms common feature selection methods. Greatest enrichment score (calculated by dividing the mean expression of a gene in each cluster by the mean expression of that gene in the dataset and retaining the greatest value) of six datasets when using different feature selection methods and different Leiden starting seeds. For each of six datasets, features were selected, using either HVGs, FindVariableFeatures (FvF), SCTransform (SCT), BigSur, or a random selection of 2,000 features; clusters were assigned using the Leiden algorithm (with resolution = 1), with 40 different starting seeds; and the highest enrichment score of the gene written in paratheses under each dataset's name was calculated. HVGs, FvF and SCT selected their default number of genes. BigSur's cutoffs were set to the top 1% of significant □' for the CD8 dataset, and 10% otherwise.

These results suggest that BigSur's feature ranking and the cutoffs chosen for each dataset

are comparable to, if not better than, those of commonly used methods.

3.4.7 Comparative performance of BigSur on datasets with known labels

The data in Figure 5 assess the ability of various feature selection methods to enrich for genes known to mark specific cell subtypes. An alternative method for comparing performance is to use datasets in which ground truth cell type labels are known *a priori*, comparing the ability of unsupervised methods to cluster them correctly. To create experimental data in which ground truth labels are known, yet are also sufficiently challenging to be sensitive to the choice of feature selection method, we generated datasets consisting of pairs of groups of cells (generally ones that could be identified by any of the features selection algorithms) and then downsampled one of the groups to make the task of discriminating between them more challenging.

We started with four datasets – a 1 million cell (1M) PBMC dataset, a skin dataset [40], the 10k PBMC dataset introduced in Figure 1, and the macrophage dataset introduced in Figure 4. Each was subset to contain only two groups of related cells (i.e., cells that display common cell-type specific markers; see Figure S9 and methods). Initially, ground truth labels corresponded to cluster identities found using features selected by BigSur; however, similar clusters could be obtained using any of the feature selection methods (using their default parameters), in almost every case (the one exception being that HVGs was unable to separate M1 and M2 macrophages; Figure S10). Interestingly, this was observed despite the fact that the rank-orders of features produced by different methods were quite different; indeed, all feature selection methods highly ranked some genes that BigSur considered to be non-significant (Figure S11). We then randomly selected a small homogenous population of cells from one of the two groups (the “rare” cells) and a larger homogenous population from the other group (the “common” cells), and combined them so that the first group was 5% of the total. The steps in generating these “semi-synthetic” datasets

are outlined in Figure S12 and described in methods. Because sampling was random, the process was replicated 20 times, generating 20 semi-synthetic datasets from each of the four datasets.

The total numbers of cells in these semi-synthetic datasets ranged from 132 to 705 (Fig. 6A). The percentages of ϕ' values that were judged significant ($p < 0.05$) was relatively low in all cases, around 3%, and median total UMI varied between about 1,500 and 10,500 (Fig. 6B). Given the results in Fig. 4, we expected these conditions would make it relatively difficult to identify the rare cell type in these cases.

For each of the 80 semi-synthetic datasets, we ran the three common feature selection methods (using default parameters) and BigSur. In each case, we also randomly chose a single set of 2,000 genes to serve as random features.

The default number of features selected by the different algorithms is shown in Fig. 6C. SCT and FvF uniformly chose 3,000 and 2,000 features, respectively. The number of features HVGs chose varied between a high of 3,094 for the 10k T cell semi-synthetic datasets and a low of 589 for the macrophage semi-synthetic datasets. Thresholds for feature selection using BigSur were informed by the results in Fig. 4, as described above. Since the 10k T cell and keratinocyte semi-synthetic datasets had relatively many cells (more than 300), displayed high median UMI/cell (around 5,870 and 9,900, respectively), and each had less than 5% of ϕ' that were significant, we used a ϕ' quantile cutoff of 0.99. Because the 1M T cell and macrophage semi-synthetic datasets both have low median UMI/cell (1,561 and 2,910 UMI/cell, respectively) as well as a low numbers of cells (298 and 132), we used a more generous ϕ' quantile cutoff of 0.9. The number of features selected by BigSur ranged from 79 to 227, far fewer features than those selected by any other method.

We then used the selected features for clustering each of the 80 semi-synthetic datasets. As

before, we used 40 independent starting seeds for Leiden clustering. We assessed performance in correctly isolating the rare cells using the same purity score as was used in Fig. 2 (fraction of rare cells in the cluster with the most rare cells). For each of the four datasets, we show results from 10 representative semi-synthetic datasets in Fig. 6D-6G, presenting the remaining 10 in Figure S12E. Fig. 6H summarizes the findings of Fig. 6D-6G, showing the median and interquartile range of purities obtained using each feature selection method.

For the 10k T cell datasets (Fig. 6D), clustering using random features yielded purities ranging from 0.08 to 0.16. Using HVGs and SCT, purities ranged from 0.26 to 0.66 and 0.27 to 0.62, respectively. FvF yielded a wider range of purities, from 0.25 to 0.74. With BigSur, purities were higher than 0.8, with the exception of a single semi-synthetic dataset (colored in red, purity of 0.42), however this dataset was particularly sensitive to Leiden starting seed, and for many starting seeds BigSur again produced very high purity. Overall, the average performance of features selected by BigSur was higher than with any other method (Fig. 6H).

For the 1M T cell datasets (Fig. 6E), random feature selection yielded purities ranging from 0.22 to 0.33, whereas HVGs, SCT, FvF and BigSur produced similar maximal purities (1.0, 0.88, 1.0 and 0.97 respectively), albeit with considerable variability. Overall, BigSur produced the highest median purity (Fig. 6H).

With the keratinocyte datasets, random feature selection and HVGs both performed poorly, yielding purities spanning 0.28 to 0.39 and 0.32 to 0.36 respectively (Fig. 6F). SCT, FvF and BigSur performance were all highly variable, ranging from poor to very good depending on the dataset. The median performance of FvF exceeded that of BigSur but the range of purities yielded by FvF was greater than that of BigSur (Fig. 6H).

Finally, with the macrophage datasets, all methods yielded very poor purities, although an

occasional dataset displayed slightly better purity when using either HVGs or FvF (Fig. 6G, colored in red).

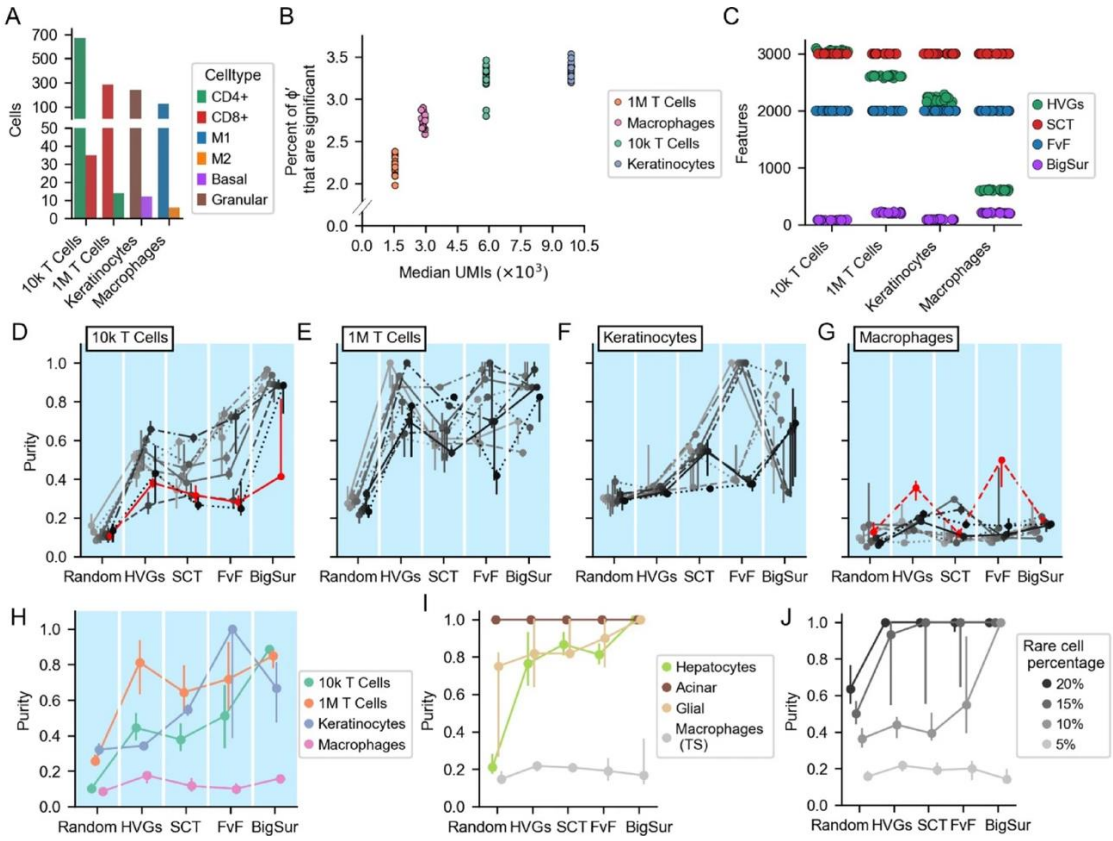


Figure 6. BigSur improves clustering performance using fewer total features.
For each of four datasets (two PBMC datasets, a skin dataset and a macrophage dataset, see main text), 20 "semi-synthetic" datasets were generated (see methods). A. Bars show the cell numbers for each cell type in each semi-synthetic dataset. B. The percent of modified corrected Fano factor values with $p<0.05$ is plotted as a function of the median UMI/cell per dataset. C. The number of features selected by each method for each semi-synthetic dataset (see main text for BigSur's cutoffs). D–H. Purity scores of each semi-synthetic dataset as the feature selection method was changed. For each of the semi-synthetic datasets, Leiden clustering, using 40 different random starting seeds, was performed using features that were either a random selection of 2,000 genes, or those chosen by HVGs, SCT, FvF and BigSur. Purity scores were calculated (as in Fig. 2). Purity scores of semi-synthetic datasets generated from the 10k T cell, 1M T cell, keratinocyte and macrophage datasets are shown in panels D, E, F and G respectively. Each line represents a different semi-synthetic dataset (for clarity, only 10 of the 20 are shown in each plot; see Fig. S12 for the remaining data), and the markers and bars are the medians and interquartile ranges (IQRs) of the purities (over the 40 different Leiden seeds). The purities from selected semi-synthetic datasets are colored for discussion purposes (see main text). H. Median purities of semi-synthetic datasets. The markers represent medians and the bars represent IQRs. I. Median purities of 20 semi-synthetic datasets generated from four datasets included in the Tabula Sapiens (TS) human cell atlas (see main text). Purities were calculated using the same procedure as for the data shown in panel H. J. Purity scores of semi-synthetic datasets generated from the macrophage (TS) dataset with varying

percentage of rare to total cells, holding the number of total cells constant. 20 semi-synthetic datasets were generated for each percentage.

Overall, even though no feature selection method consistently outperformed in all datasets, BigSur was the only method that had median scores above 0.6 for the first three datasets (Fig. 6H). While the median purity yielded by FvF was larger than that of BigSur on one dataset, the purity scores FvF produced in the 10k and 1M T cell datasets and the keratinocyte datasets were more widely spread than those of other methods, suggesting that FvF was overly sensitive to the specific cells composing each semi-synthetic dataset.

Taken together, these results argue that the performance of BigSur is generally as good as or better than other feature selection methods.

3.4.8 BigSur also outperforms other methods in less challenging datasets

The semi-synthetic datasets described in the previous section not only contained a rare cell type, but many contained relatively small numbers of cells as well as relatively low values of UMI per cell (Figure 6A-B). Such characteristics would be expected to make feature selection especially challenging. To see whether BigSur continued to provide advantages with less challenging data sets, we generated additional semi-synthetic datasets in which sequencing depth was higher. For this purpose, we downloaded four datasets from the Tabula Sapiens (TS) human cell atlas (liver, pancreas, eye and lymph nodes) [41]. For each, we selected a particular cell type (hepatocytes from the liver, acinar cells from the pancreas, Müller glial cells from the retina and macrophages from the lymph node), clustered it, and selected two subclusters (see Figure S13, procedure detailed in methods), which were subsequently used to generate semi-synthetic datasets as described above (Figure S14A-D). Figure S14E-F shows the number of cells and statistics of these datasets. Dataset sizes varied from ~100 to ~250 cells, and median sequencing depth ranged

from 9,136 to 31,997 UMI/cell.

Using each dataset we selected features, assigned clusters using 40 different Leiden starting seeds, and calculated purity scores. We selected BigSur's cutoffs by following the previously stated guidelines: for the hepatocyte and acinar semi-synthetic datasets, we selected genes with the top 1% of significant ϕ', and for the macrophage (TS) and glial semi-synthetic datasets, we selected genes with the top 10% of significant ϕ'.

All methods performed poorly in the macrophage (TS) semi-synthetic datasets and performed nearly perfectly in the acinar datasets; as for the other datasets, only BigSur yielded median and 25th quantiles equal to unity (Figure 6I). The macrophage (TS) semi-synthetic datasets only contained 5 rare cells, out of a total of 114 cells; we therefore generated additional macrophage (TS) semi-synthetic datasets with increasing percentages of rare to total cells, keeping the total number of cells at 114. When increasing the percent of rare cells to 10%, an increase of only 5 cells, the median purity using BigSur increased to unity, whereas the other methods generally performed similarly to random feature selection (FvF did slightly better; Figure 6J). As the ratio of rare to total cells was increased, the purity scores yielded by all methods increased, and all methods except FvF had median and 25th quantile scores of unity.

These results suggest that in small, deeply sequenced datasets with few rare cells, BigSur still outperforms common feature selection methods.

3.4.9 Validation of semi-synthetic results

In comparing the performance of different feature selection methods, a variety of choices had to be made regarding, for example, the metric used to assess performance, the resolution used during clustering, the number of genes selected by each method, the size of the semi-synthetic

datasets, and how semi-synthetic datasets were generated. To ascertain whether any of these choices might have inadvertently biased results in favor of BigSur, we explored making a variety of alternative choices. As summarized below, in each case, we did not observe substantive changes in qualitative conclusions.

First, we considered two alternatives to purity scores: we either replaced purity scores with their calculated *p*-values (i.e., their probability of occurring by chance under the assumption that target cells are evenly distributed among all clusters), or we replaced purity scores with an alternative measure of diversity, the normalized entropy [42]. When calculating these metrics for the 10k T cell, 1M T cell, keratinocyte and macrophage datasets, we found that normalized entropy and purity score had similar trends, and all *p*-values of the purity scores – besides those produced in the macrophage semi-synthetic datasets – were lower than 0.05 (Figure S15A-B).

Next, we examined the effect of the resolution parameter used during Leiden clustering, which influences the number of clusters that are identified. We varied Leiden resolution in each semi-synthetic dataset generated from the 10k T cell dataset and calculated purity scores [33]. We found that the purity scores from BigSur selected features were consistently higher than those from any other selection method at the same resolution (Figure S15C-D). Interestingly, we also found that the number of clusters yielded by BigSur tended to increase more slowly with increasing resolution than other methods (Figure S15E).

We next asked whether reducing the number of features selected by HVGs, FvF and SCT from their default values might improve purity scores in the 10k T cell, 1M T cell, keratinocyte and macrophage semi-synthetic datasets (in effect testing whether these methods were, in fact, ranking features well, but simply choosing too many of them). We thus lowered the number of features selected by each method to the number of features that BigSur had selected, assigned

clusters, and calculated purity scores. The results were mixed: only the purities yielded by SCT improved for all datasets. With HVGs and FvF, some purities increased and others decreased (Figure S16A). In a converse experiment, we increased the number of genes selected by BigSur in semi-synthetic datasets with 10% rare-to-total cells (shown in Figure 6I) and found that median purities fell from 1 to 0.45 (Figure S16B).

We also investigated how important the number of cells in a dataset was in influencing the difficulty of feature selection. To do this, we generated additional semi-synthetic datasets with greater numbers of cells (numbers of cells and statistics of the datasets shown in Figure S17A-B; see methods for details) and found that most feature selection methods yielded purities of 1, with the exception of FvF which performed similarly to random feature selection in the macrophage (TS) semi-synthetic datasets (Figure S17C).

Finally, since we had used BigSur to initially identify the cell types used in creating semi-synthetic datasets, we considered the possibility that doing so might have unfairly advantaged BigSur in performing the task of clustering the two cell types. To test this, we generated additional semi-synthetic datasets, using HVGs to select features at each subsetting step and during the semi-synthetic generation process (the intermediate steps for doing so are outlined in Figure S18-19 and detailed in the methods). The statistics of these semi-synthetic datasets are shown in Figure S20A-B. We found that BigSur still yielded the highest or second highest purities for all semi-synthetic datasets, except for those generated from the 10k T cell datasets (Figure S20C-D). Interestingly, during subsetting of the pancreas dataset, we noticed that two clusters with high expression of an acinar marker (*CLPS*) defined using features produced by HVGs had an order-of-magnitude difference in sequencing depth (clusters 0 and 1, shown in Figure S20E), suggesting that artifacts reflective of sequencing depth differences might have driven the clustering of these cells into two

distinct groups. Consistent with this view, BigSur reported that nearly all modified corrected Fano factors (ϕ') in datasets produced by mixing these clusters were statistically insignificant, and therefore did not justify subdivision into two separate clusters.

Collectively, these results strengthen our conclusions that BigSur performs as well or better than other methods in a wide range of datasets.

3.4.10 Comparison of speed and memory usage of different feature selection methods

We compared the speed of computation of each feature selection method on datasets of varying size. To do so, we generated datasets with increasing numbers of cells by randomly sampling from the full 1M PBMC dataset and calculated the speed of each method for each set of cells (see methods). BigSur was slower than HVGs and FvF but faster than SCT (Figure S21A). At least 90% of the computation time for BigSur appears to be spent on *p*-value calculation (compare dashed line in Figure S21A). Even with 100,000 cells, however, BigSur finishes in about a minute and a half.

We also wanted to measure the speed of computation relative to the number of genes. Accordingly, we generated datasets from the 1M PBMC dataset with 10,000 cells and varying number of genes. SCT was the slowest method overall, followed by BigSur (with and without *p*-value calculations), FvF and HVGs (Figure S21B).

We also measured the total memory used by each method on datasets of varying cell and gene numbers. When varying numbers of cells, HVGs and BigSur's memory usage stayed relatively small, up to 0.59 GB and 0.97 GB respectively when each method was run on a dataset of 250,000 cells (Figure S21C). The memory usage of FvF on a dataset of the same size was 9.5 GB. We were unable to run SCT on a dataset of this size, due to insufficient memory; the total

memory usage of SCT increased substantially with increasing number of cells, up to 815 GB with 100,000 cells. When varying the number of genes, SCT was also the most memory intensive method, requiring up to 49 GB in a dataset with 10,000 cells and 21,819 genes (Figure S21D). In that dataset, the memory usages of the other methods were similar to one another, at around 0.86 GB.

3.5 Discussion

Feature selection is known to be an important general step in machine learning [9], but the impact of different methods of feature selection on single cell transcriptomics has not been fully explored. Currently, several methods are in common use as part of popular analysis packages, and multiple others have been proposed as potential improvements [10-19]. Clear guidance about how to choose among them, or when to consider adjusting their parameters, is difficult to come by. Although some head-to-head comparisons have been published [e.g., [13, 20, 21]], attention is not often paid to the difficulty of the task for which feature selection is used. Here, we focus on unsupervised cell clustering and note that, in many cases – such as when cells are highly distinct in gene expression – feature selection hardly matters, and random sets of genes often do nearly as well as carefully chosen ones. Under these circumstances, even if large differences exist in the accuracy with which different methods identify truly variable genes (e.g., [12]) they are likely to be of little practical importance.

In contrast, it is when attempting to separate subtly dissimilar groups of cells, especially when present as minor cell subpopulations, that the choice of feature selection method and the number of selected features can greatly impact accuracy of clustering. The impact of feature selection in this regime has been relatively unexplored, even though it may be expected to arise in

many common applications, such as when repetitively subclustering cell populations; characterizing cell heterogeneity in tumor samples; or distinguishing intermediate states in cell lineages.

The results presented here argue that, in such cases, it is important to use feature selection methods that not only identify genes that are truly variable but also eliminate ones that are not. Ultimately, the success of clustering must reflect a balance between signal (true positives) and noise (false positives). When the proportion of true positives is high, contamination by false positives may be irrelevant, but at lower signal strengths eliminating false positives is essential.

Here we developed a straightforward analytical approach, BigSur, that models the null distribution of gene expression observations as Poisson random variates from a distribution reflecting biological gene expression noise, the coefficient of variation of which is estimated from the data. For each gene, BigSur returns both a measure of variability ϕ', and—by modeling the gene expression noise distribution as log-normal—the probability of observing that value by chance. The method automatically accounts for differences in data sparsity across genes, and differential sequencing depth across cells, so no data normalization or transformation is required. Whereas other feature selection methods sometimes work by fitting data to distributions, those distributions are typically inferred from empirical observations of scRNAseq datasets (e.g. [19, 23]), whereas with BigSur the use of the lognormal distribution to represent gene expression variability is grounded in both theory and observation .

We showed here that BigSur can select informative genes and eliminate non-informative ones in both simulated and real datasets and described how simple dataset statistics can be used to limit the number of selected genes. We then compared the performance of BigSur against three popular feature selection methods: FindVariableFeatures (the default in Seurat), HVGs (the default

in scanpy), and SCTransform v2 [19, 23-25]. Both simulated data, real data, and "semi-synthetic" datasets formed by guided subsetting of real data were used, concentrating on what we expected to be especially challenging regimes. Not only did these three methods highly rank genes considered non-significant by BigSur, they tended to deliver much longer lists of features. For the most part, BigSur performed as well as or better than other algorithms (Fig. 5 and 6), despite using far fewer features, suggesting that, overall, it has a substantially better signal-to-noise profile.

We did not assess the ability of BigSur to assist tasks other than cell clustering, such as in pseudotemporal ordering, but suspect that the improvement in signal-to-noise ratio in feature selection will be helpful in that setting as well. Also, as noted above, the modified corrected Pearson residuals produced by BigSur may be used to generate modified corrected Pearson correlation coefficients, from which meaningful networks of gene co-expression correlation may be inferred, thanks to a dramatic reduction in false positive correlation [22].

3.6 Conclusions

We show that the importance of feature selection algorithm – how features are ranked and the number of features selected – is related to the difficulty of the clustering task. From simple assumptions, we derive a statistically principled model, BigSur, that accounts for biological and technical noise in single cell transcriptomics. We show how summary statistics calculated using BigSur, along with summary statistics of the dataset, can inform feature selection. Careful consideration of feature selection in single cell transcriptomics analyses improves clustering accuracy, which will in turn improve the quality of results from subsequent tasks.

3.7 Methods

3.7.1 The modified corrected Fano factor ϕ_j'

The Fano factor quantifies how much the variance of a distribution differs from that of a Poisson distribution. To the extent that scRNAseq data are not generally Poisson-distributed, the Fano factor does not reliably measure unexpected variability, but it can be modified, given an appropriate model, to do so. For any gene, assuming the null hypothesis (all cells draw from the same distribution), we model scRNAseq data as a random Poisson sample from a log-normal distribution, the mean of which is scaled in each cell by that cell's sequencing depth.

To account for variance of sequencing depth across cells, we follow [37] in correcting Pearson residuals. The Pearson residual is a measure of each observation's difference from the mean, scaled to the square root of the mean (equation (2), Results section). We correct it by using a different definition of mean in each cell, one that is scaled to account for total number of UMI in the cell.

The corrected Pearson residual is then further modified to account for the expected greater variance of a compound Poisson log-normal distribution, rather than that of a Poisson distribution. The variance of a compound distribution (assuming independence) is the sum of variances of the two underlying distributions. If the first is a Poisson distribution, then its variance is equal to its mean. For the second distribution, we equivalently express variance as $c^2\mu^2$, the square of the coefficient of variation times the mean. The total variance is therefore $\sigma^2 = \mu + c^2\mu^2 = \mu(1 + c^2\mu)$. We then use the square root of this expression to replace the denominator of the corrected Pearson residual. Introducing indices *i* and *j* to represent cell and gene, respectively, we obtain the following modified corrected Pearson residual P'_{ij}:

$$P'_{ij} = \frac{x_{ij} - \mu_{ij}}{\sqrt{\mu_{ij}\left(1 + c^2\mu_{ij}\right)}}$$

Just as the Fano factor, for gene *j*, may be expressed as an average over squared Pearson residuals, so may the modified corrected Fano factor be defined as

$$\phi_j' = \frac{1}{n-1}\sum_{i=1}^{n} {P'_{ij}}^2 = \frac{1}{n-1}\sum_{i=1}^{n} \frac{\left(x_{ij}-\mu_{ij}\right)^2}{\mu_{ij}\left(1+c^2\mu_{ij}\right)}$$

Where *n* is the number of cells. Note that the modified corrected Pearson residual can also be used to derive other useful statistics, such as a modified corrected Pearson correlation coefficient [22].

The only hyperparameter used in these calculations is *c*, which may be estimated from the behavior of an entire dataset, as described below. Note that the expectation value of ϕ_j' under the null hypothesis is unity regardless of the underlying distribution that describes gene expression. The choice of the log-normal distribution here only influences the expectation value for higher moments of P', which are used in the calculation of *p*-values.

3.7.2 Calculation of ϕ_j' p-values

The analytical nature of the model underlying the calculation of ϕ_j' allows one to estimate, given a set of observations and cell sequencing depths, the probability that any given value of ϕ_j' would arise by chance. Specifically, for each gene, we calculate the moments of the Poisson log-normal distribution [43], from which we calculate the moments of the modified corrected Pearson residuals, which in turn are used to calculate the first 5 moments of ϕ_j'. These moments describe the distribution of ϕ_j' under the null hypothesis, which is that gene *j* is measured in a homogenous group of cells. We then use numerical procedures [38] to estimate the quantile of an observed ϕ_j' in this distribution, which may be expressed as a *p*-value. Derivation of the equations for the moment calculations may be found in the appendix of [22].

3.7.3 Fitting a coefficient of variation for underlying gene expression

Under the assumptions that c is approximately equal across genes, and that most of the genes in a dataset are not significantly differentially expressed across cells, one can estimate c by finding the value that minimizes the difference between ϕ_j' and 1, for the majority of genes. Because c influences the modified corrected Pearson residual only through the term $1 + c^2\mu_{ij}$, it follows that the choice of c has little influence on ϕ_j' when μ_{ij}<<1. We learn c by finding the value that minimizes the absolute value of the slope of a linear fit to a plot of $ln(\phi_j')$ vs. $ln(\mu_j)$, for $\mu_j \in [0.01, 10]$.

3.7.4 Generation of simulated datasets

We first generate underlying means for each gene:

$$x \sim Normal(-1, 0.8)$$

$$\mu_{j,k} = 10^x$$

Where $\mu_{j,k}$ is the mean of gene j for cell type k. We then generate transcript counts for each cell by sampling from a log-normal distribution with c = 0.7:

$$y_{i,j,k} \sim Log\text{-}Normal(\mu_{j,k}, c)$$

Where $y_{i,j,k}$ is the count of gene j in cell i of cell type k. Note that in many texts, log-normal distributions are parametrized in terms of the mean and standard deviation of the underlying normal distribution from which they may be derived, but here $\mu_{j,k}$ and c refer to the actual mean and coefficient of variation of the log-normal distribution.

Finally, we use $y_{i,j,k}$ as the rate parameter for a Poisson distribution and sample from it to simulate the sequencing of transcripts from each cell:

$$z_{i,j,k} \sim Poisson(y_{i,j,k})$$

Where $z_{i,j,k}$ is the simulated UMI for each gene in each cell. The datasets in Fig. 3 simulate 2,000 cells, each expressing 15,000 genes. The cells are divided into two equally sized groups of 1,000. For some of the genes ("not truly variable" genes), expression values for both groups are generated using a single $\mu_{j,k}$. For other genes ("truly variable" genes), the $\mu_{j,k}$ in the two groups of cells are generated using independent, random samples of x (i.e., $\mu_{j,k=1} \neq \mu_{j,k=2}$). Thus, the degree to which "truly variable" genes differ in expression between the two groups of cells is itself distributed about a mean of zero.

For the simulated data, clusters were assigned using the Leiden algorithm with a resolution parameter of 0.1. In Fig. 3B, ten rounds of random feature selection and Leiden clustering were done on each dataset and the average results across the ten trials is presented. In Fig. 3F, Leiden clustering was performed 50 times using unique random seeds and results were averaged across all trials.

3.7.5 Dimensionality reduction and clustering

Dimensionality reduction and clustering were done using scanpy v1.8.2. Counts were log-normalized as follows:

$$x_{i,j} = ln\left(\frac{counts_{ij}}{\sum_i counts_{ij}} * 10^4 + 1\right)$$

Where $x_{i,j}$ is the log-normalized count of gene j in cell i.

Once features were selected, regardless of feature selection method, principal component analysis (PCA) was performed on the log-normalized data and the top 50 principal components (PCs) were retained. The *k*-nearest neighbors graph was calculated from the PCs, and UMAP dimensions were calculated using scanpy default parameters. Clusters were identified using the

Leiden algorithm implemented by scanpy with default parameters and, if not otherwise stated, resolution = 1 and random seed = 0 (default).

3.7.6 Classification of cell types using random genes

Cell types in the 10k PBMC dataset were identified using marker genes on clusters calculated as specified above, using HVGs with defaults as feature selection. scrublet v0.2.3 [44] was used to identify doublets using default parameters. For every set of randomly selected genes (uniform sampling using numpy.default_rng.choice function, setting the replace parameter to False [45]), PCA was calculated using sklearn's [46] implementation of TruncatedSVD with number of PCs = 50, except for 25 genes where number of PCs = 25. A linear support vector machine (SVM, using sklearn.svm.SVC) was trained on each set of PCs calculated by TruncatedSVD, with ground truth labels being the cell types identified as described above. The adjusted Rand index (ARI) and normalized mutual information (NMI) scores were calculated by comparing predicted labels of the SVM to ground truth labels, using sklearn's implementation of both scores.

3.7.7 Differential expression testing

To test for differential gene expression (as in Figure S4), for a given clustered dataset, we compared the normalized expression of each gene in each cluster to the normalized expression of that gene in the rest of the dataset, using the Mann-Whitney-U test (MWU) [47]. We corrected the *p*-values for multiple testing using the Benjamini-Hochberg approach [39].

3.7.8 Generation of semi-synthetic datasets

To compare feature selection methods, we devised a procedure that creates datasets in which there are two populations of relatively similar but transcriptionally distinct cells, one substantially rarer than the other. The procedure entails taking pairs of transcriptomically similar

groups of cells that had previously been labeled, clustering each group to identify homogenous populations, and mixing small numbers of cells from these homogenous populations together. We detail the identification of the pairs of groups of cells belonging to each dataset in the next three sections.

Specifically, for a given pair of groups of cells, we clustered them separately, by selecting genes using either BigSur or HVGs and clustering using the Leiden algorithm with resolution = 1. For the semi-synthetic datasets analyzed in Figure 6A-I (i.e. the 10k T cell, 1M T cell, keratinocyte and macrophage semi-synthetic datasets, along with the "smaller" TS semi-synthetic datasets), we selected the largest cluster from each group. We then randomly downsampled the smaller of the selected clusters, and combined it with the larger cluster, such that the resulting ratio of cells from the smaller and larger cluster was 1:19.

For the datasets analyzed in Figure 6J, i.e. the semi-synthetic datasets with varying percentages of rare cells, we started with the same clusters used to generate the macrophage (TS) semi-synthetic datasets and adjusted the ratio of cells sampled from each cluster, holding the number of cells constant.

For the datasets analyzed in Figures S16H-J, (i.e. the "larger" TS semi-synthetic datasets), we clustered the smaller of the two groups (resolution = 1) and selected the largest of the resulting clusters. We subsequently randomly sampled cells from the larger of the two groups, without clustering, and combined the sampled cells with the selected cluster, such that the ratio was 19:1 common to rare cells.

To generate the macrophage, glial and macrophage (TS) semi-synthetic datasets using HVGs, we clustered each group using a resolution = 0.5, selected the largest cluster from each group and randomly downsampled the smaller of the selected clusters, and combined it with the

larger cluster, such that the resulting ratio of cells from the smaller and larger cluster was 1:19.

For each semi-synthetic dataset, we removed genes expressed in fewer than three cells before selecting features. The cell barcodes of each semi-synthetic dataset were saved (see data availability).

3.7.9 Subsetting of the 10K T cell, 1M T cell, macrophage and keratinocyte datasets to make semi-synthetic datasets

To make semi-synthetic datasets, we first subset each dataset to only include two groups of cells. We used the Leiden algorithm to cluster, with resolution = 1 if not otherwise specified.

For the 10k PBMC dataset, we removed all cells except for the CD4+ and CD8+ T cells, which were previously identified (see Figure S1 and Figure S9A-B).

For the 1M PBMC dataset, due to the large numbers of cells in this dataset, we did several rounds of downsampling. We started by removed all cells with no expression of CD3E, and subsequently removed all genes expressed in less than 3 cells and removed all cells expressing less than 400 genes. We then selected features using BigSur (only selecting the top 2% of ϕ', without considering *p*-value) and clustered on the top 50 PCs. This procedure identified 9 clusters (Figure S9C). We then removed clusters 6 and 8, removed all genes expressed in less than 3 cells and removed all cells expressing less than 400 genes, and selected features using BigSur (only selecting the top 2% of ϕ', without considering *p*-value), and clustered, which identified 11 clusters (Figure S9D). We removed all cells except for those in clusters 0 and 3, removed genes expressed in less than 3 cells, selected features using BigSur (selecting the top 10% of significant ϕ') and clustered, which identified 7 clusters (Figure S9E). From these clusters, we selected clusters 3 and 4, removed genes expressed in less than 3 cells, selected features using BigSur (selecting the top 10% of

significant ϕ') and clustered, which yielded 6 clusters (Figure S9F). All cells except for clusters 1 and 3 were selected, and we identified the CD4+ and CD8+ T cells using *CD4* and *CD8* (Figure S9G-H).

For the macrophage dataset, we removed all genes expressed in less than 3 cells and removed all cells expressing less than 400 genes. We subsequently removed the unstimulated macrophage progenitor cells, annotated as "BM0" cells, removed genes expressed in less than 3 cells, selected features using BigSur (selecting the top 10% of ϕ' with p-values < 0.01) and clustered. This procedure resulted in 11 clusters (Figure S9I). We subsetted the dataset to only include clusters 2 and 3. We identified the M1 and M2 macrophages using *SOD2* and *RGCC*, respectively (Figure S9J-K).

For the keratinocyte dataset, we subsetted the dataset to only contain data from a single patient (labeled ADSWT11), and subsequently removed all genes expressed in less than 3 cells and removed all cells expressing less than 400 genes. We then selected features using BigSur (only selecting the top 10% of significant ϕ') and clustered, with resolution = 0.1. This procedure identified 7 clusters (Figure S9L). We identified the keratinocyte cluster, cluster 0, using the mean expression of KRT14 in the cluster (mean expression of KRT14 > 3; Figure S9M). We selected the keratinocyte cluster, removed all genes expressed in less than 3 cells and removed all cells expressing less than 400 genes, selected features using BigSur (using the top 5% of ϕ' with p-values < 0.01), and clustered. This resulted in 15 clusters (Figure S9N). We subsetted the dataset to only include clusters 1, 2, 3 and 5. We identified the basal cells using *KRT5* and the granular cells using *IVL* (Figure S9O-P).

3.7.10 Subsetting of the Tabula Sapiens datasets to generate semi-synthetic datasets

To generate semi-synthetic datasets from the Tabula Sapiens datasets, we first subset each

dataset to only contain two groups of cells. As in the previous section, we used the Leiden algorithm to cluster, with resolution = 1 if not otherwise specified.

Specifically, we downloaded the liver, pancreas, eye and lymph node datasets from the Tabula Sapiens (TS) atlas (see data availability). Since the datasets include reads from 10x Genomics and SmartSeq2, we only used the 10x Genomics reads. For each dataset, we selected the genes with the top 10% of significant ϕ_j' as features to use for dimensionality reduction.

In each dataset, we selected and clustered a cell type, displayed in Figure S13. Specifically, for the liver dataset, we removed all cells besides the hepatocytes, identified using *PCK1* expression, from patient TSP14 (clusters 0, 3, 4, 12 and 14); for the pancreas dataset, we removed all cells besides a cluster of acinar cells, identified using the expression of *CLPS*, cluster 0; for the eye dataset, we removed all cells besides clusters 1, 5 and 11, which we identified as Müller glial cells using *GLUL*; and for the lymph node dataset, we removed all cells besides cluster 19, which we identified as macrophages from their expression of *CD68*.

Subsequently, for each of these smaller subsets, we selected features (by selecting genes with the top 10% of significant ϕ_j'), clustered and removed all cells besides two groups of cells. Specifically, we chose clusters 0 and 4 from the hepatocytes; clusters 0 and 1 from the acinar cells; clusters 2 and 4 from the glial cells; and clusters 0 and 1 from the macrophages (see Figure S13 and S14A-D).

3.7.11 Subsetting of datasets using HVGs to produce two clusters for semi-synthetic dataset generation

Since we generated semi-synthetic datasets using BigSur to select features, which could introduce bias, we also generated semi-synthetic datasets of similar size using HVGs. As above,

we used the Leiden algorithm to cluster, with resolution = 1 if not otherwise specified.

Specifically, for each of eight datasets (the macrophage dataset, the 1M PBMC dataset, the 10k PBMC dataset, the skin dataset, and the liver, pancreas, retina, and lymph node datasets from the Tabula Sapiens atlas), we removed cells with less than 400 expressed genes and genes expressed in less than 3 cells. In addition, for the macrophage dataset, we also removed data generated from macrophages in the first stage of differentiation and data generated from the progenitor cells (labeled BM0 and AHL60 in the metadata, respectively), along with the repolarized macrophages (labeled XM2M2 and ZM2M1). For the 1M PBMC dataset, due to the size of this dataset, we also removed all cells not expressing CD3E. For the skin dataset, we removed all cells not sequenced from patient ADSWT11.

We then selected features using HVGs, clustered, and identified a specific cell type using known markers. Specifically, we identified T cells in the 10k PBMC dataset using *CD3E*, keratinocytes in the skin dataset using *KRT14*, hepatocytes in the liver dataset using *PCK1*, acinar cells in the pancreas using *CLPS*, Müller glial cells in the retina dataset using *GLUL* and macrophages in the lymph node dataset using *CD68* (see Figure S18).

Then, for each clustered cell type, we selected cells from particular clusters and removed the rest of the cells. Specifically, we selected clusters 1, 3, 4, 7, 8, 9 and 14 from the PBMC dataset, clusters 4 and 10 from the T cells identified in the 1M PBMC dataset, clusters 0 and 1 from the macrophages dataset, clusters 0, 3, 5, 6, 7, 9, 10, 12, 13, 15 and 17 from patient ADSWT11, clusters 0, 1, 3, 11 and 27 from the liver dataset, clusters 1, 2, 3, 5, 6, 14 and 18 from the pancreas dataset, clusters 28 and 30 from the retina dataset, and cluster 15 from the macrophages identified in the lymph node dataset (see Figure S18).

Finally, for each resulting subset, we selected features using HVGs. For the 10k T cell, 1M

T cell, keratinocyte, and hepatocyte subsets, clusters were assigned, and the two largest clusters (0 and 1) were used to generate semi-synthetic datasets. To avoid selecting clusters with large differences in sequencing depth (see Figure S18), for the acinar subset, clusters 1 and 4 were used to generate semi-synthetic data, and for the macrophage subset, clusters 1 and 3 were used to generate semi-synthetic data.

3.7.12 Enrichment score

To calculate the enrichment score of a gene in a clustered dataset, we divide a chosen gene's mean expression in cells of the cluster by the mean expression of the gene across all cells. In Figures 5 and 6, we show the highest enrichment score of the dataset.

3.7.13 Purity score

For a dataset in which there is *a priori* knowledge of which cells have a particular identity, one may define a purity score that evaluates the ability of unsupervised clustering to place those "target cells" together within a single cluster. After clustering, we identify the cluster that has the greatest number of target cells and then define the purity score as the fraction of target cells in that cluster.

We also calculated the probability of seeing a given purity score if the target cells were distributed in each cluster at random (the p-value of the purity score), using the hypergeometric distribution, which represents the probability of seeing k successes with n draws, without replacement:

$$p(k, M, n, N) = \frac{\binom{n}{k}\binom{M-n}{N-k}}{\binom{M}{N}}$$

Where M is the size of the dataset, n is the total number of target cells, N is the size of the cluster

that has the greatest number of target cells and k is the number of target cells in that cluster.

When displaying the purity score for simulated data (Fig. 3B, 3F), we present the mean of the purity scores calculated for each of the two cell types.

3.7.14 Normalized Entropy

The normalized entropy, a common measure of diversity, is the Shannon entropy divided by the maximum possible entropy [42]:

$$\eta(x) = -\sum_{i=1}^{n} \frac{p(x_i)\ln\big(p(x_i)\big)}{\ln(n)}$$

Where n is the total number of clusters and $p(x_i)$ is the probability of seeing a target cell x in cluster i, or equivalently the fraction of all target cells that is in cluster i.

3.7.15 Speed and memory comparisons

To compare the speed and memory usage of HVGs, SCT, FvF and BigSur when varying numbers of cells, we used the 1M PBMC dataset from Parse Biosciences. The dataset was filtered to retain only cells with more than 400 genes and genes expressed in at least 3 cells. For the comparison between HVGs and BigSur, 10 datasets with increasing numbers of cells (100, 500, 10^3, $5*10^3$, 10^4, $5*10^4$, 10^5 and $2.5*10^5$ cells) were randomly sampled from the 1M PBMC dataset. Each sampled dataset was filtered to retain only genes expressed in at least 1 cell.

Since SCT and FvF were first implemented in R, we randomly sampled a $2.5*10^5$ dataset from the filtered dataset and exported to R. 10 datasets were randomly sampled from the filtered dataset per number of cells, and the features were calculated from the sampled datasets.

To compare the speed and memory usage of the different feature selection methods when varying numbers of genes, we randomly selected 10,000 cells from the 1M PBMC dataset and

removed all genes that were not expressed in at least one cell. From this dataset, ten datasets with increasing numbers of genes (100, 500, 10^3, $5*10^3$, 10^4, $2*10^4$, and the maximum number of genes in this dataset, 21,819) were randomly sampled. We exported the initial 10,000 cell dataset to R and generated ten datasets with the same numbers of genes as in python.

The total memory used by BigSur and HVGs was recorded by the tracemalloc package which is included in python. The total memory used by FvF and SCT was recorded by the profmem package [48].

All real and semi-synthetic dataset results were generated on an iMac Pro (2017) with 10 3 Ghz Intel Xeon W cores and 256 GB of RAM running Ventura 13.2.1.

3.8 Abbreviations

scRNAseq: single cell transcriptomics

BigSur: Basic Informatics and Gene Statistics from Unnormalized Reads

PCs: Principal components

PCA: Principal component analysis

UMAP: Uniform Manifold Approximation and Projection

SVM: Support Vector Machine

ARI: adjusted Rand index

NMI: normalized mutual information

PBMC: peripheral blood mononuclear cell

HVGs: highly variable genes

Treg: T regulatory cell

TS: Tabula Sapiens

UMI: Unique molecular identifier

FDR: False discovery rate

FvF: FindVariableFeatures

SCT: SCTransform

3.9 Declarations

Ethics approval and consent to participate

Not Applicable.

Consent for publication

Not Applicable.

3.10 Availability of data and materials

The 10k PBMC dataset was downloaded from 10x Genomics, https://cf.10xgenomics.com/samples/cell-exp/6.1.0/10k_PBMC_3p_nextgem_Chromium_X/10k_PBMC_3p_nextgem_Chromium_X_raw_feature_bc_matrix.h5. The 1 million PBMC dataset was downloaded from Parse Biosciences, https://support.parsebiosciences.com/hc/en-us/articles/7704577188500-How-to-analyze-a-1-million-cell-data-set-using-Scanpy-and-Harmony. The retina [32], macrophage [49] and keratinocyte [40] datasets were downloaded from GEO (GSE137537, GSE164498 and GSE141526 respectively). The Tabula Sapiens datasets were downloaded from the Tabula Sapiens website, https://figshare.com/articles/dataset/Tabula_Sapiens_v2/27921984 [41].

The simulated data can be found on GitHub: https://github.com/landerlabcode/BigSur.

The metadata for each dataset, including the cell barcodes of each semi-synthetic dataset, are included in supplementary data.

Code is freely available as a python package on GitHub: https://github.com/landerlabcode/BigSur. An R version can be found at: https://github.com/landerlabcode/BigSurR. A Mathematica version can be found at: https://github.com/landerlabcode/BigSurM. A tutorial on how to use BigSur for feature selection is included as part of the python package.

3.11 Competing interests

The authors declare no competing interest.

3.12 Funding

Research support was provided by NIH grants U54CA217378, P30AR075047, P01CA288662, R01DE019638, R01AR081671, R01GM152494 and R01CA237563. EPD and KS were also supported by training grant T32GM136624.

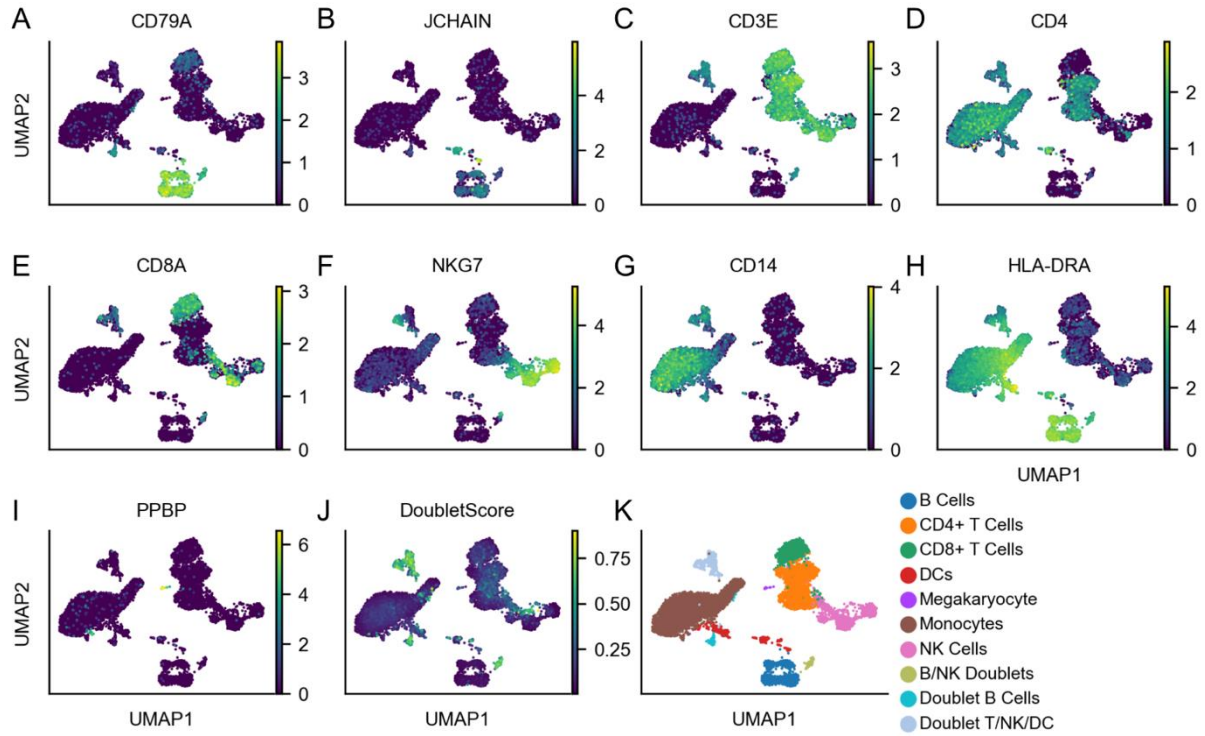
3.13 Authors' contributions

ADL, QN and SA conceived and supervised this project. EPD and KS conducted the data analysis. ADL, EPD and KS interpreted the data. ADL, EPD and KS drafted the manuscript. All authors contributed to revisions of the manuscript. All authors read and approved the final manuscript.

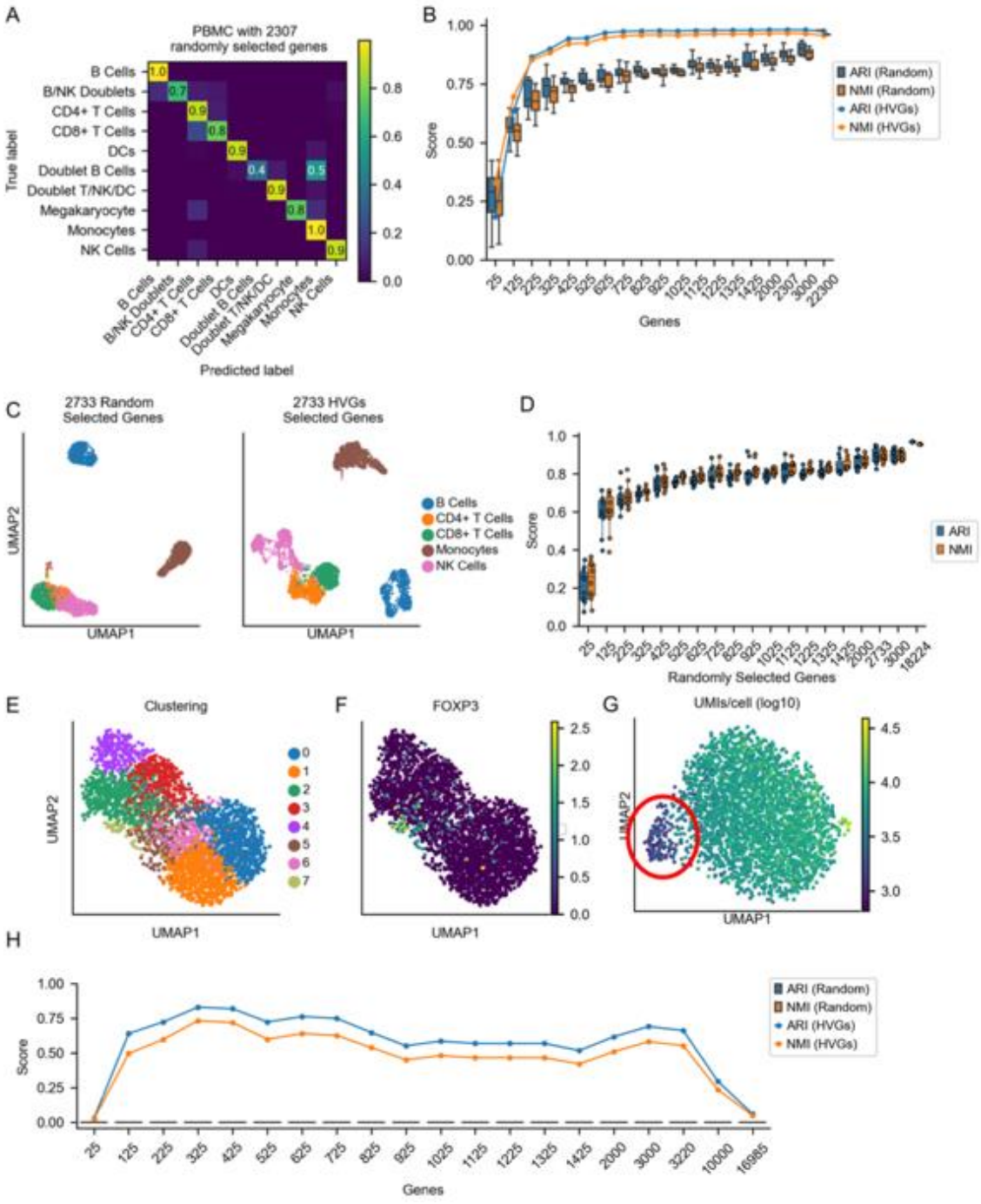
3.14 Acknowledgements

The authors thank Fairlie Reese, Samuel Morabito and Tatyana Lev for assistance with code optimization, and Joshua Gervin for helpful discussions during preparation of the manuscript.

3.15 Supplementary Figures

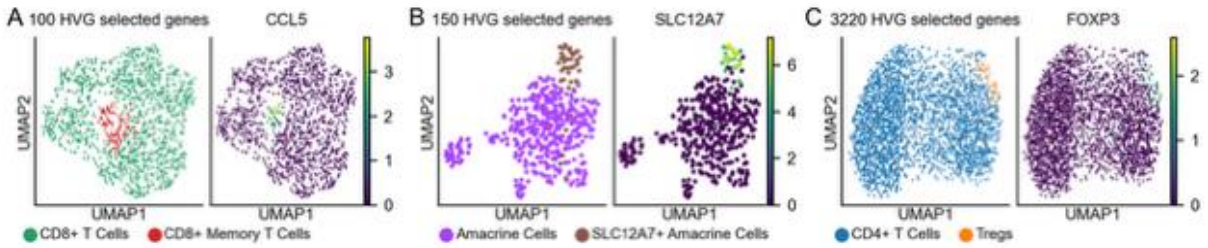


Supplementary Figure 1. Celltype identification in the PBMC dataset. UMAPs of the 10k PBMC dataset, with cells colored by expression of different marker genes, along with the doublet score calculated by scrublet and the cell type identities. Features were selected using HVGs defaults. A – B. CD79A and JCHAIN were used to identify B cells. C – E. Markers to identify the T cells: CD3E for all T cells, CD4 for CD4+ T cells, CD8A for CD8+ T cells. F. NKG7 was used to identify NK cells. G. CD14 was used to identify monocytes. H. Co-expression of HLA-DRA and CD14 was used to identify dendritic cells (DCs). I. PPBP was used to identify Megakaryoctes. J. Scrublet was used to calculate the doublet score for each cell (see methods). K. Celltypes identified using the above markers; same panel as in Figure 1A.

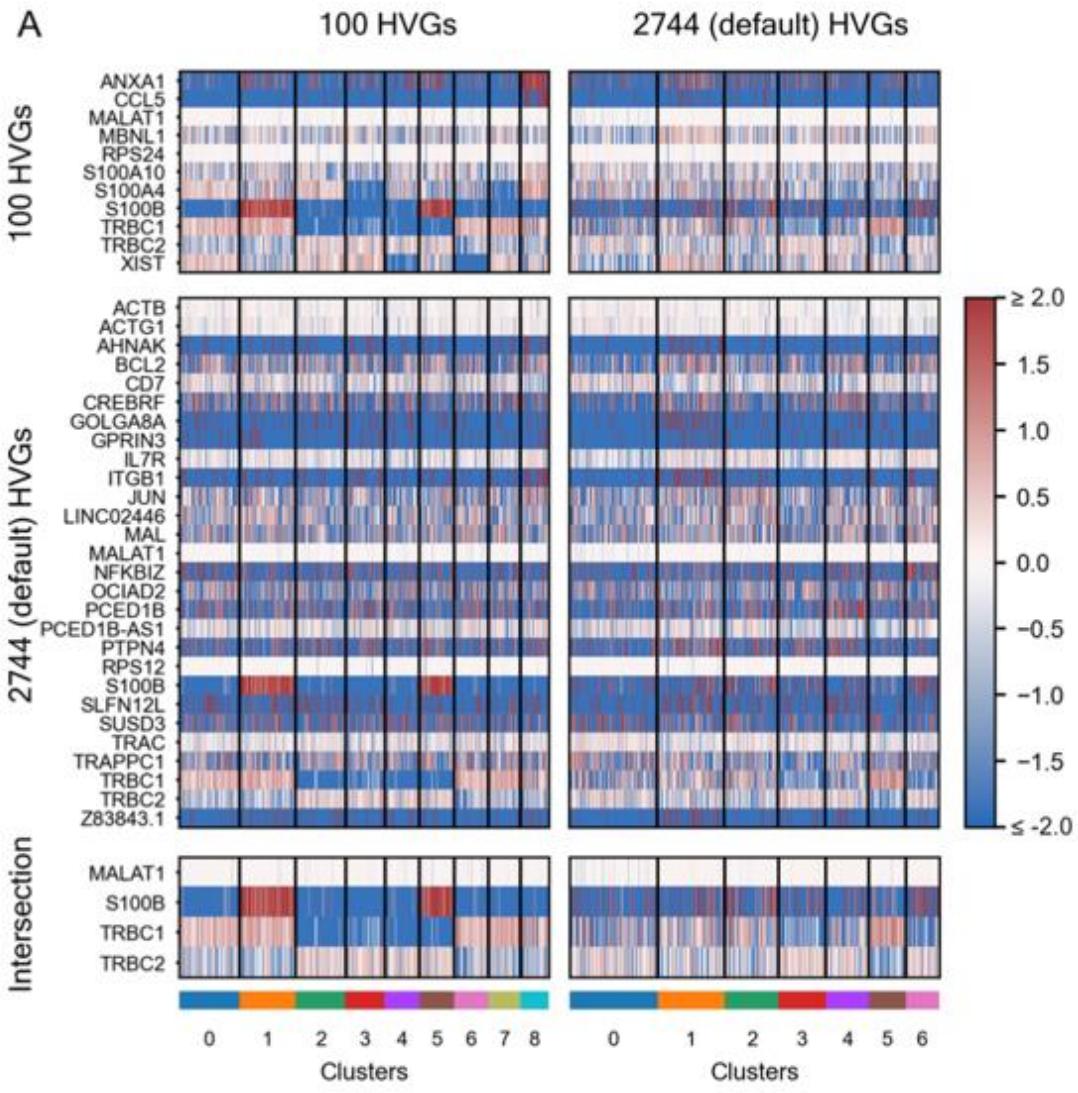


Supplementary Figure 2. Cell types can be identified using randomly selected features in some datasets. A. The confusion matrix of predictions made by a linear support vector machine (SVM) trained on the 50 top principal components (PCs) of the PBMC dataset from 2307 randomly selected genes. The rows of the confusion matrix sum to 1. The fraction of correct predictions (diagonal) are displayed in each tile, along with the fraction of doublet B cells predicted as monocytes. B. Adjusted Rand index (ARI) and normalized mutual information (NMI) scores of SVMs trained on the 50 top PCs of the PBMC dataset for varying number of genes ranked by HVGs. The ARI and NMI scores from randomly selected genes (data from Figure 1B) are included for comparison. C.– D. The cell types numbering less than 1000 in the

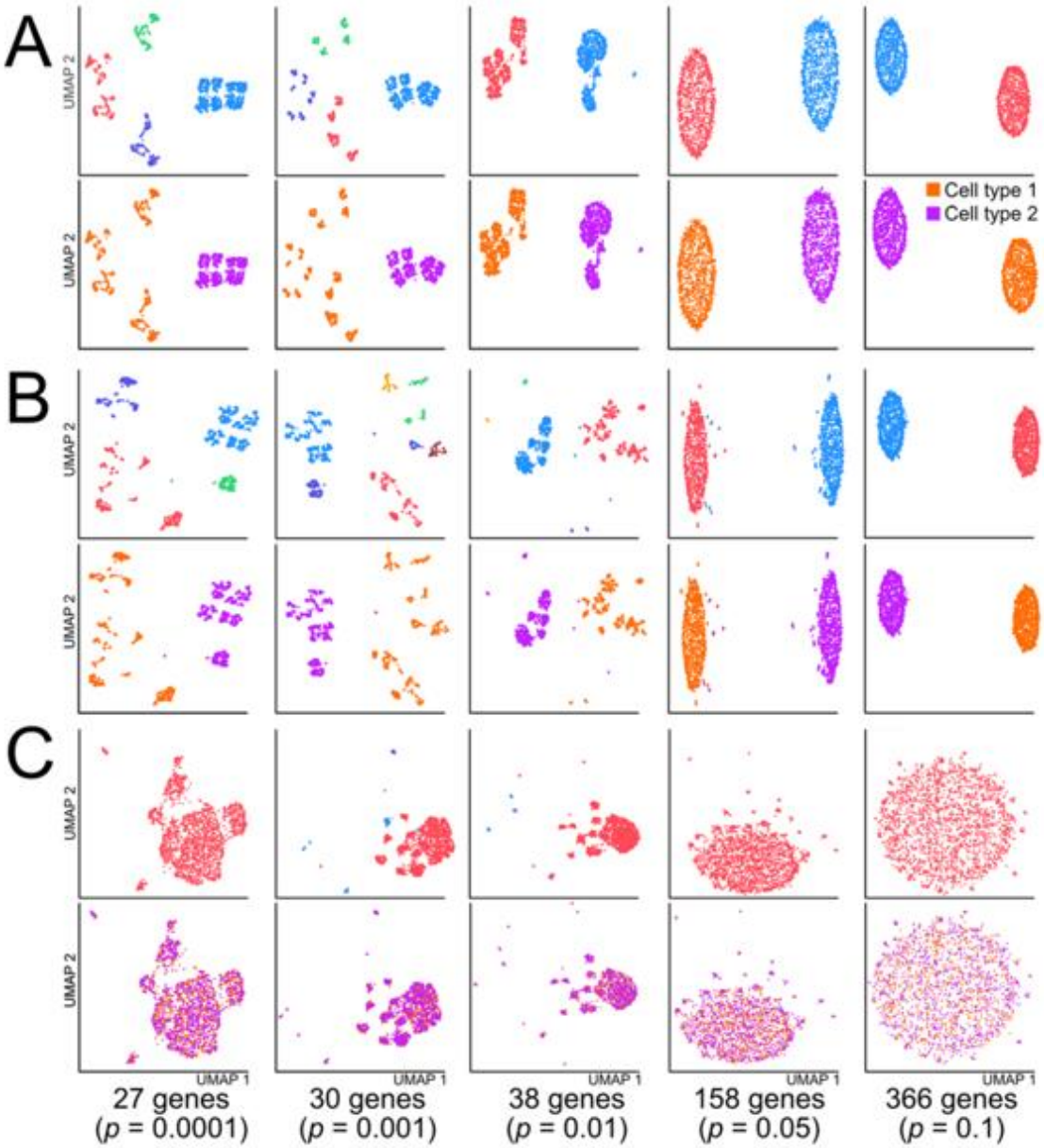
PBMC dataset were removed. The remaining cell types were randomly downsampled to 1000. Panel C displays the UMAPs of this dataset, calculating using either randomly selected genes (left) or HVGs (right). Panel D shows the ARI and NMI scores of SVMs trained on PCs calculated from increasing numbers of randomly selected genes. E. UMAP of CD4+ T cells from the PBMC dataset, generated by selecting features using HVGs and clustering using Leiden. F . Feature plot of FOXP3 expression of the CD4+ T cells. G. UMAP of CD4+ T cells, calculated by randomly selecting 350 genes and calculating their PCs (same as in Figure 1C). A small subpopulation of lowly sequenced cells (circled in red) separated out from the other cells. H. Adjusted Rand index (ARI) and normalized mutual information (NMI) scores of SVMs trained on the 50 top PCs of the CD4+ T cell dataset for varying number of genes ranked by HVGs. The ARI and NMI scores from randomly selected genes (data from Figure 1D) are included for comparison; the default number of features selected by HVGs was 3220.



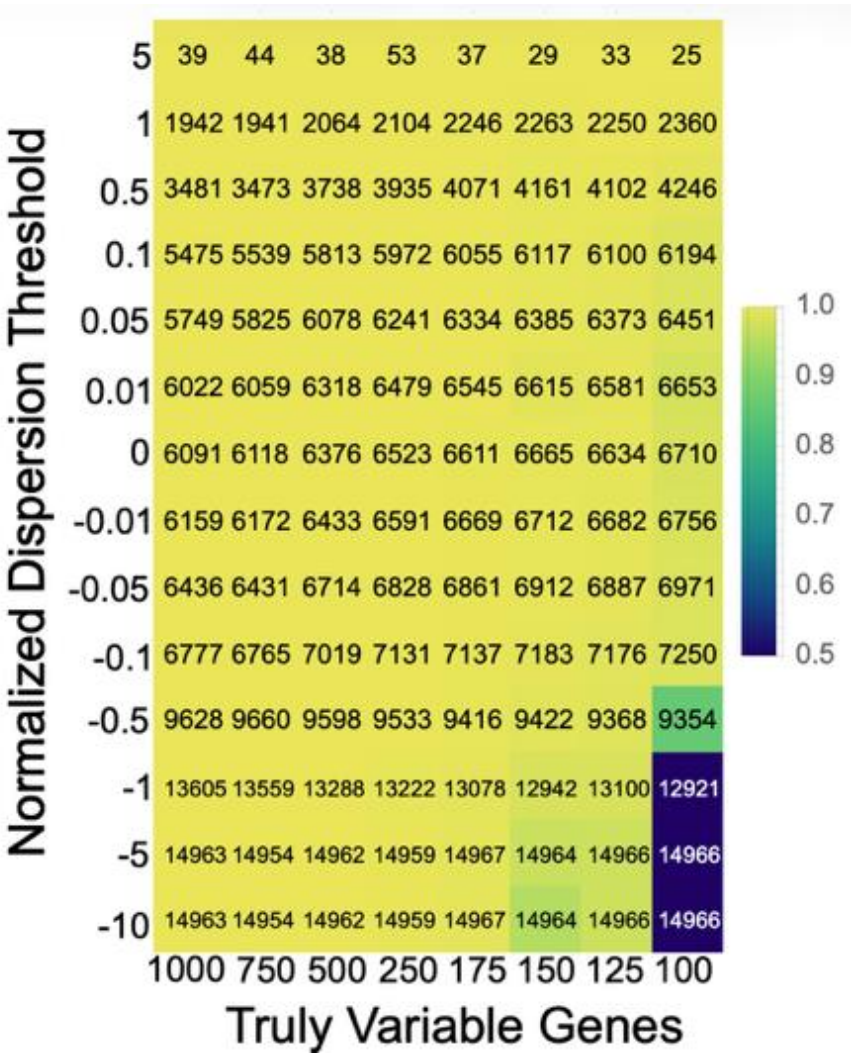
Supplementary Figure 3. Identification of clusters used in Figure 2. A. UMAPs of CD8+ T cells from the PBMC dataset, generated by selecting the top 100 features ranked by HVGs and clustering using the Leiden algorithm. The CD8+ memory T cell population was identified by expression of CCL5. B. UMAPs of amacrine cells, which were subset from the retina dataset. The top 150 genes ranked by HVGs were selected and the cells were clustered using the Leiden algorithm. A cluster of SLC12A7- expressing amacrine cells was identified. C. UMAPs of CD4+ T cells from the 10k PBMC dataset. The top 3220 HVG features were selected and the cells were clustered using the Leiden algorithm. Tregs were identified using FOXP3 expression.



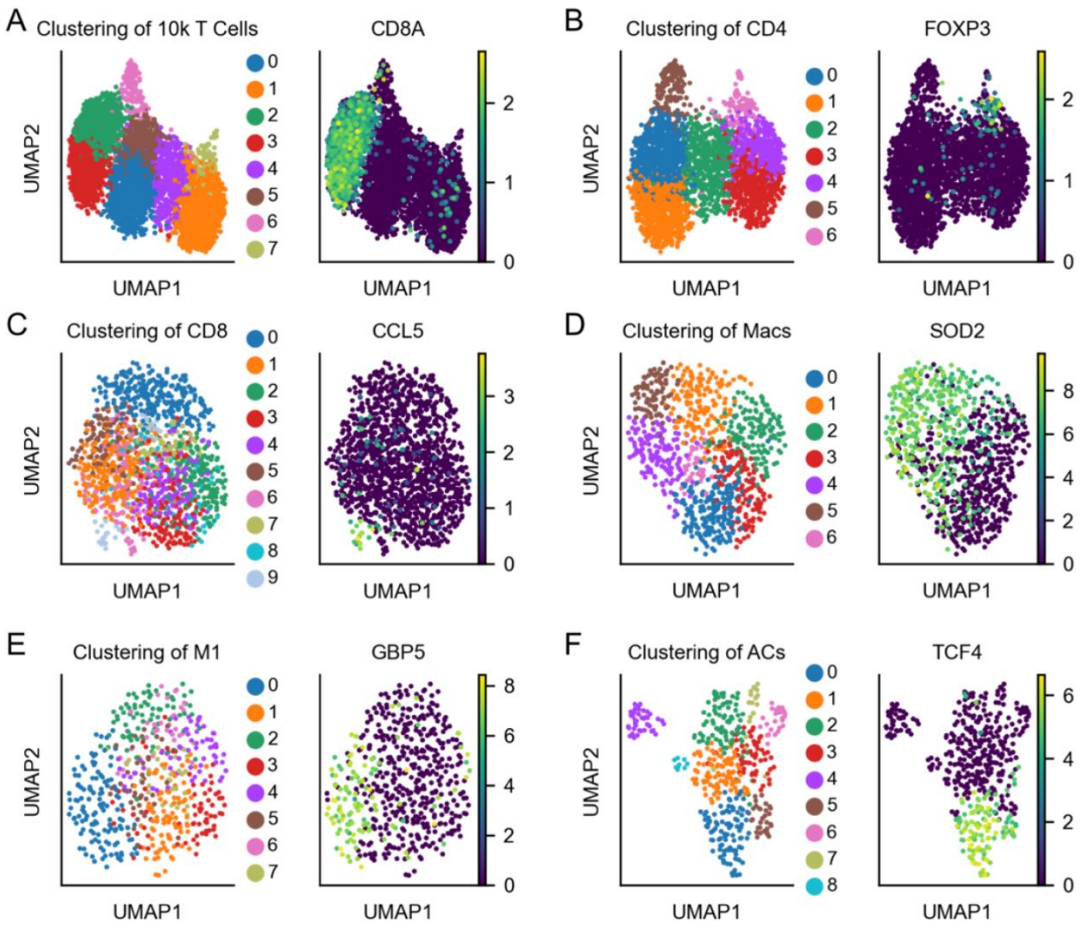
Supplementary Figure 4. Changes in differentially expressed genes in the CD8+ T cell dataset with varying number of selected features. Log2 fold change of differentially expressed genes (DEGs) of the normalized expression in each cell to the mean normalized expression in the CD8+ T cells, for different sets of clusters. The CD8+ T cells were clustered by the Leiden algorithm using either the top 100 HVGs or the default number of HVGs. The differentially expressed genes (DEGs) for each cluster were identified using an FDR-adjusted Mann-Whitney-U p-value threshold of 0.05. If more than five genes had a p-value below 0.05, only the top five DEGs (ranked by average log foldchanges) are shown. For genes in which there was no expression in a cell, the log2 fold change was set to -2.



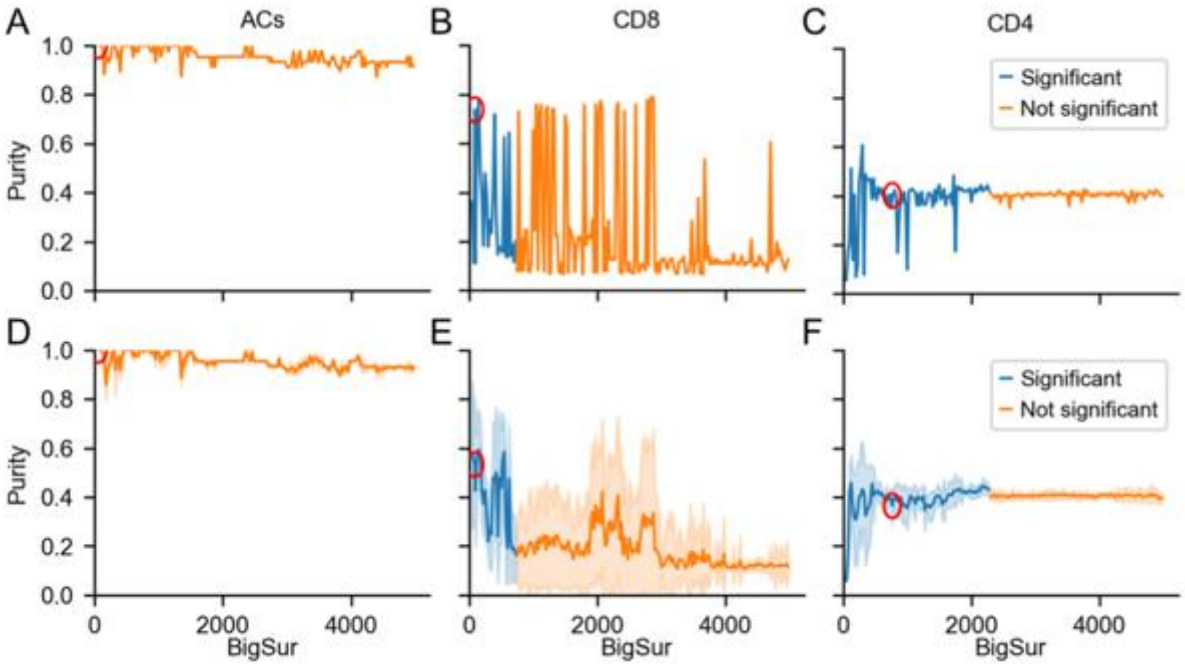
Supplementary Figure 5. Clustering of simulated data using different feature selection methods and cutoffs. UMAPs of simulated data with 100 truly variable genes (dataset used in Fig. 3B-F) calculated using different feature selection methods (BigSur, HVGs and randomly selected genes in panels A, B and C respectively) and number of genes (columns). For each panel, the top row displays cells colored by cluster assignment (using a Leiden resolution of 0.1), and the bottom row displays cells colored by cell type identity. For each UMAP, the counts matrix was log-normalized (see "Dimensionality reduction and clustering" section of the methods), features were selected, the top 50 PCs were selected (or all PCs if the number of selected features is lower than 50) and used to calculate the UMAP coordinates. A. UMAPs were calculated using varying numbers of genes using p-value cutoffs calculated using BigSur. B – C. UMAPs calculated using the same number of features as in panel A. Features were selected by HVGs (panel B) or at random (panel C).



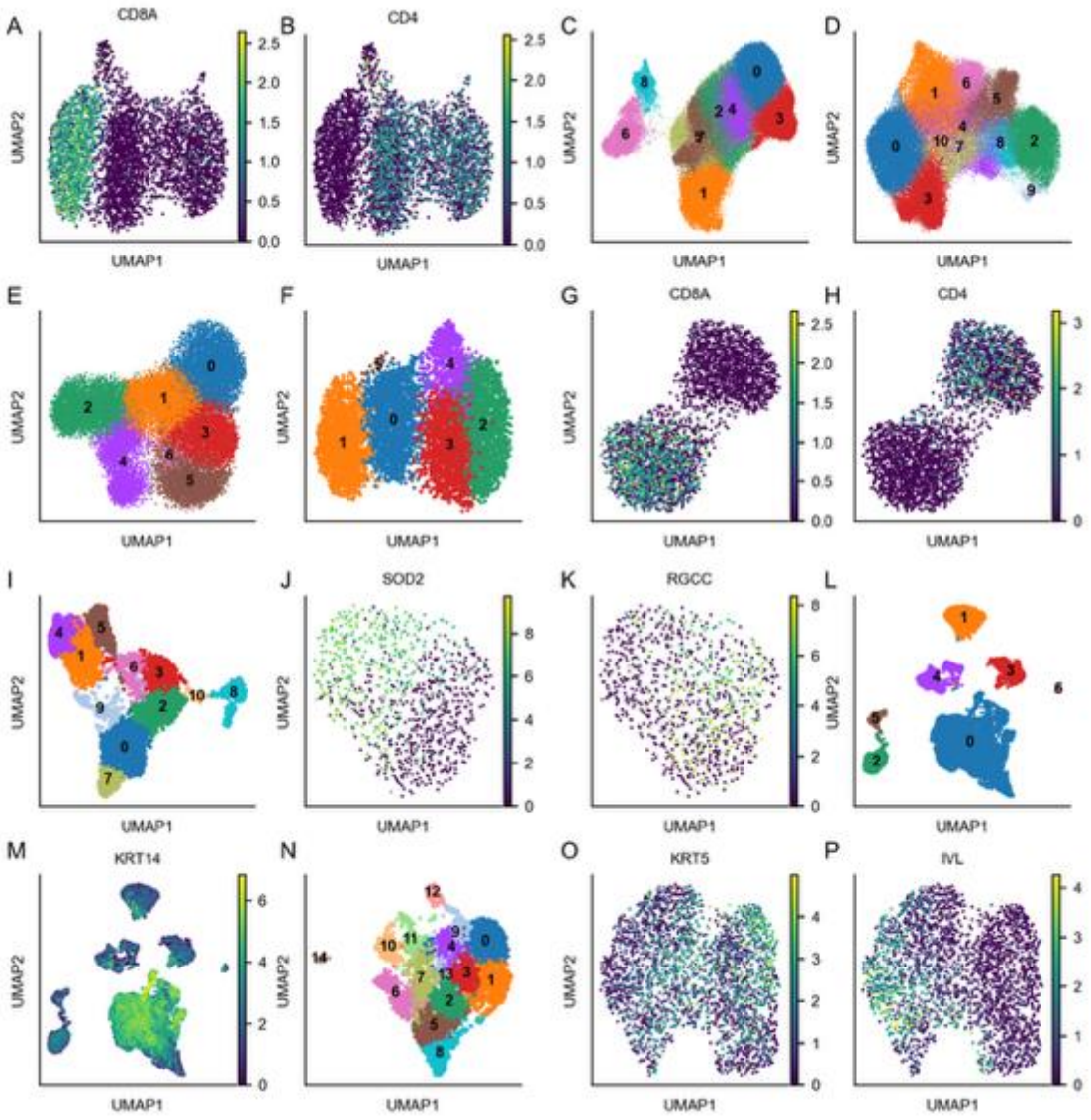
Supplementary Figure 6. Purity scores of simulated data when using HVGs. Heatmap of purity scores of the simulated data in Figure 3B yielded by features selected by HVGs at different normalized dispersion thresholds. Tile color indicates purity score. Numbers overlaid on each tile indicate the number of features selected. Note that HVGs selects all genes with normalized dispersion > 0.5 by default.



Supplementary Figure 7. Markers and clusters used for enrichment calculations. UMAPs of datasets used for the enrichment calculations in Figure 4J and K are shown, along with the marker used to calculate enrichment. Clusters were assigned using the Leiden algorithm with resolution = 1. For the 10k T cell, CD4 T cell, Macs (macrophages), and AC datasets, the genes with top 10% of ϕ' (0.9 quantile cutoff) and with p-values < 0.05 were used to calculate principal components (PCs). For the CD8 T cell and M1 macrophage datasets, the genes with top 1% of ϕ' (0.99 quantile cutoff) were used to calculate PCs.

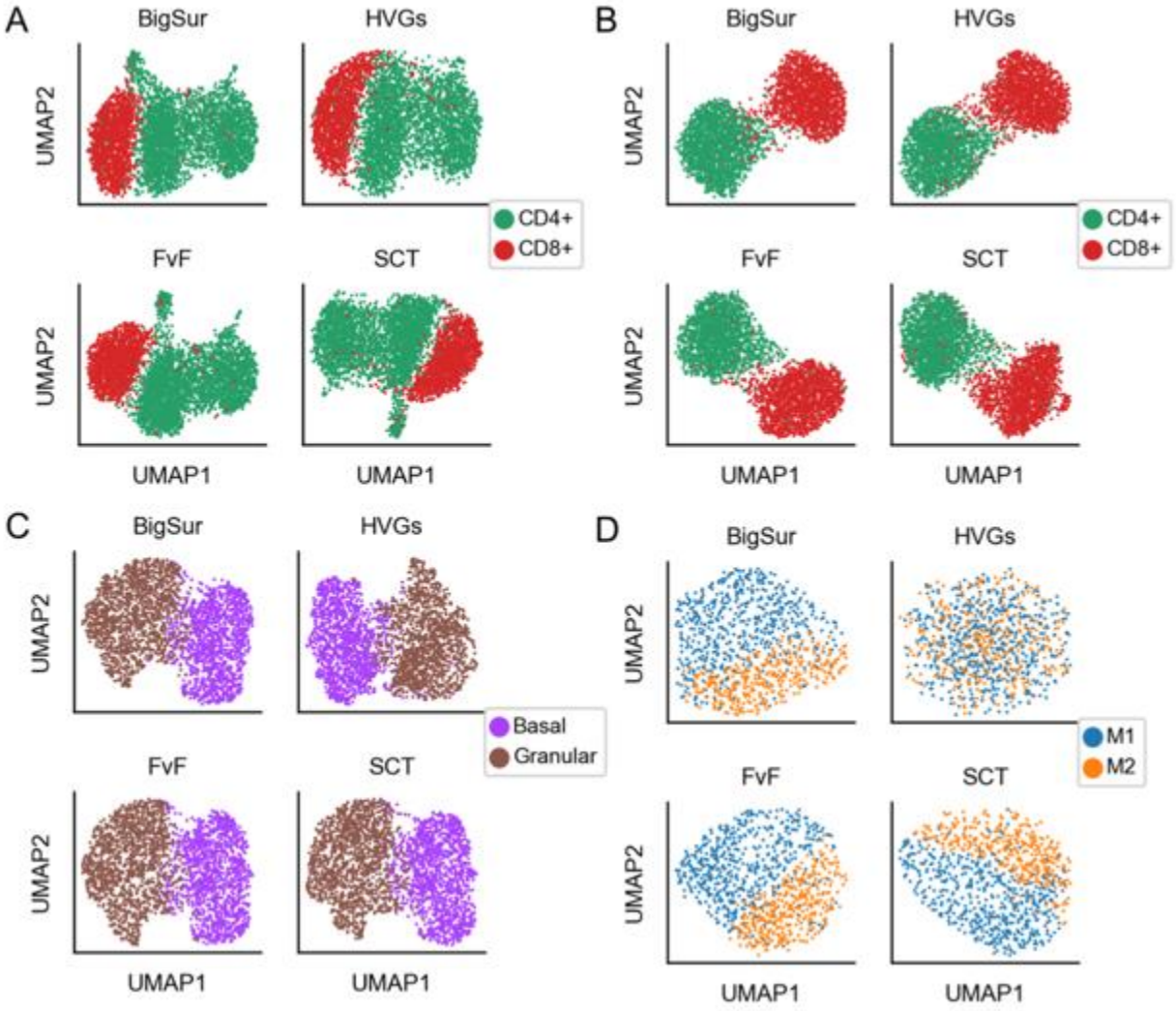


Supplementary Figure 8. Purity of Figure 2 datasets using BigSur. The purities of datasets shown in Figure 2 with varying number of genes selected by BigSur are shown. The genes were first ranked by significance of ϕ', then by magnitude of ϕ'. For each set of features, clusters were assigned using the Leiden algorithm and the purity score was calculated. The red circles mark the default selection of genes for each dataset. A – C. Purity of the ACs (panel A), CD8 (panel B) and CD4 (panel C) datasets, using the default Leiden seed of 0. D – F. Mean and standard deviation of the purity of the ACs (panel D), CD8 (panel E) and CD4 (panel F) datasets using 50 randomly selected starting seeds for the Leiden algorithm.

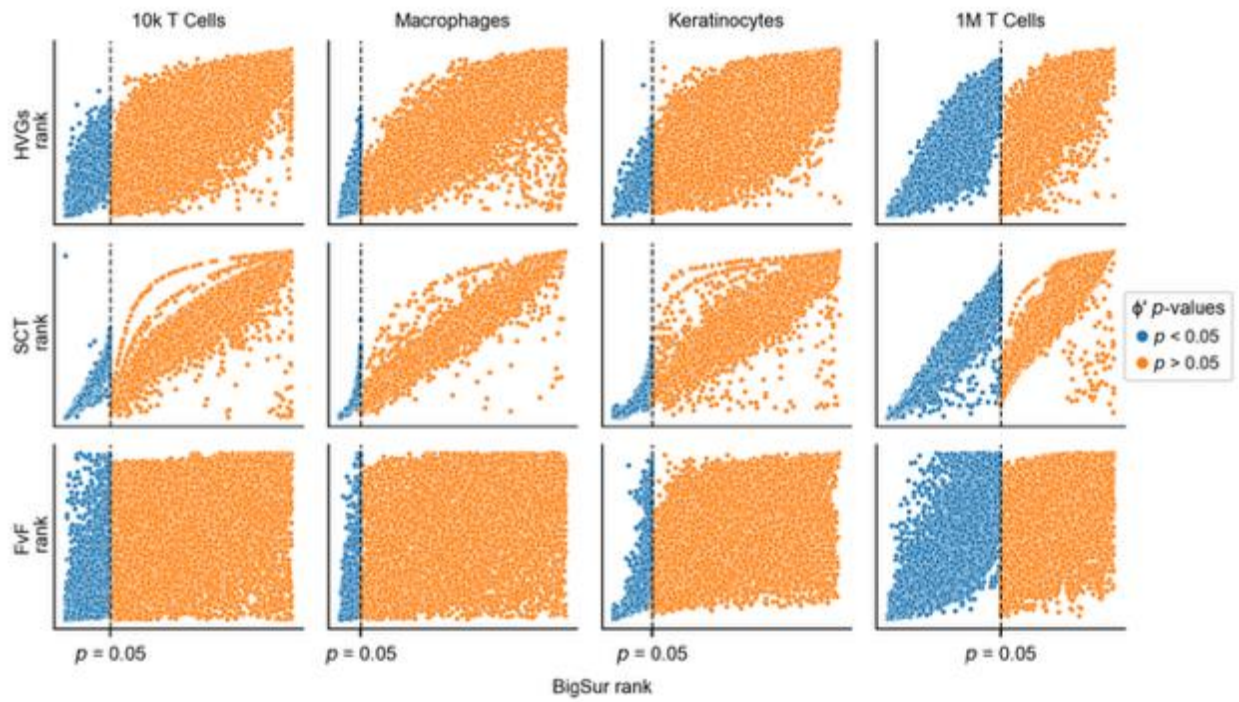


Supplementary Figure 9. Intermediate steps to downsample datasets used for semi-synthetic dataset generation. The intermediate steps to downsample datasets subsequently used to generate semi-synthetic data are visualized, along with the marker genes used to identify the two groups of cells. The specific procedure downsampling procedure is detailed in the methods. A – B. UMAPs of the T cells from the 10k PBMC dataset (calculated from the top 10% of significant ϕ'), colored by expression of CD8A and CD4. C – F. UMAPs of the four successive downsampling steps for the 1M T cell dataset. Feature were selected using BigSur, with various cutoffs (see methods). G – H. UMAPs of the CD8 and CD4 T cells from the downsampled 1M T cells dataset (calculated from the top 10% of significant ϕ'), identified by expression of CD8A (panel G) and expression of CD4 (panel H). I. UMAP of the macrophage dataset (calculated from the top 10% of ϕ' with p- values < 0.01). J – K. UMAP of clusters 2 and 3 of the macrophage dataset (previously shown in panel I; UMAPs were calculated from the top 10% of significant ϕ'). The M1 macrophages were identified using SOD2 expression (panel J) and the M2 macrophages were identified using expression of RGCC (panel K). L – M. UMAP of the cells from patient ADSWT11 (included in the skin dataset), calculated using the top 10% of significant ϕ'. Clusters shown in panel L were assigned using resolution = 0.1. The keratinocytes were identified using

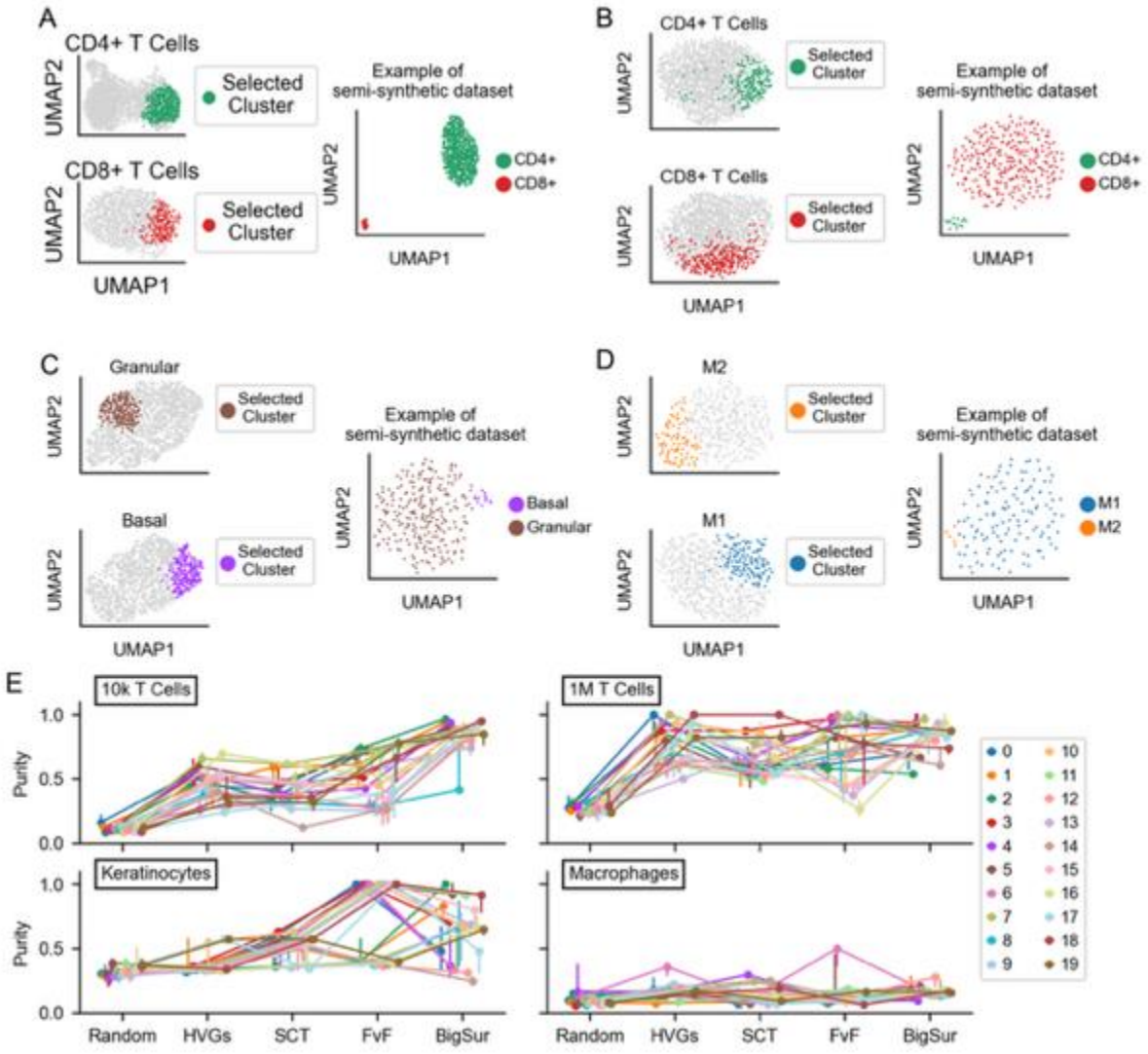
expression of KRT14 (shown in panel M). N. UMAP of the keratinocytes (cluster 0 in panel L), calculated using the top 5% of ϕ' with p-values < 0.01. O – P. UMAP of clusters 1, 2, 3 and 5 from the keratinocytes (shown in panel N), calculated from the 10% of significant ϕ'. Basal cells were identified using KRT5 expression (panel O) and granular cells were identified using IVL (panel P).



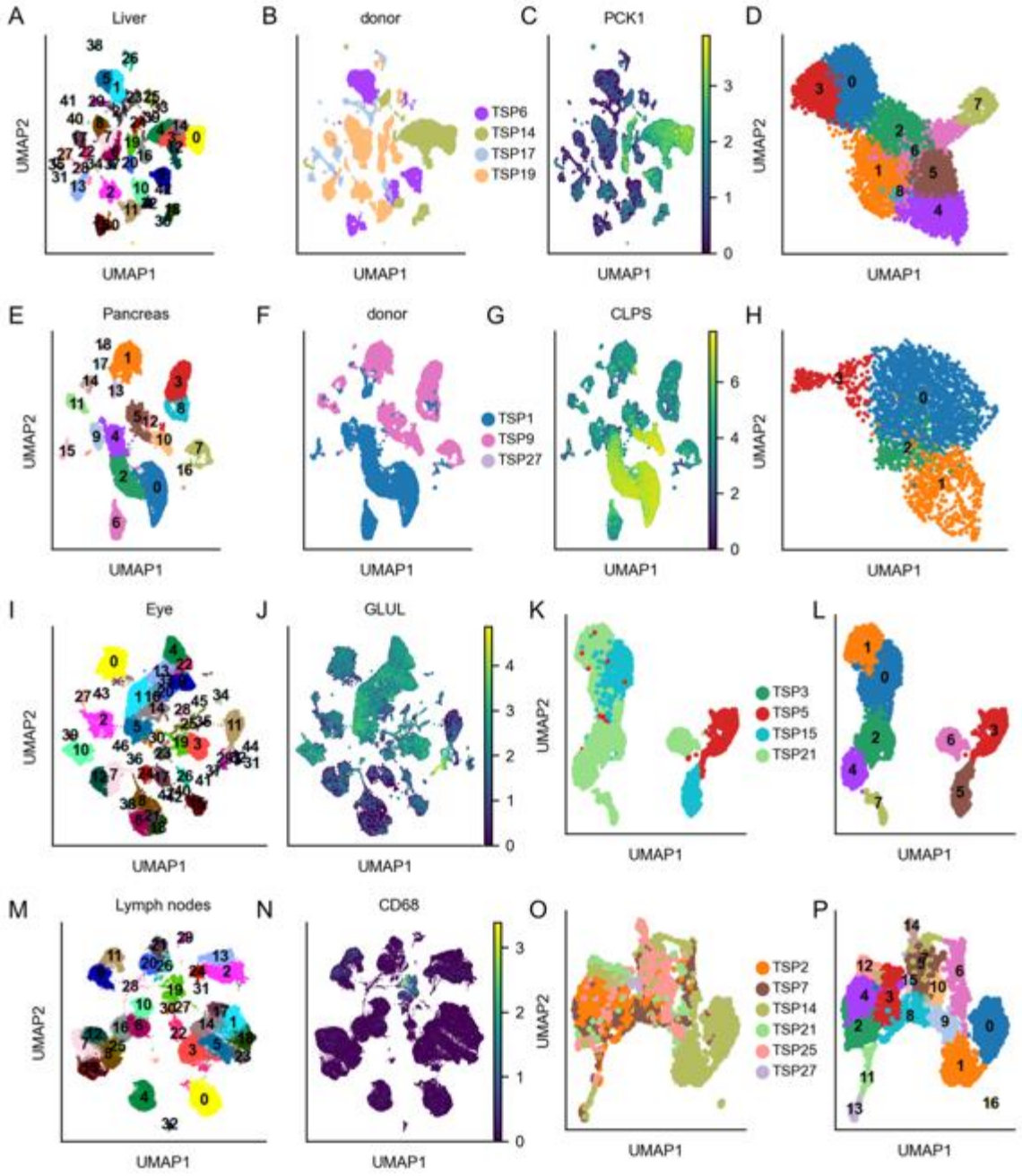
Supplementary Figure 10. Feature selection of four methods on the four datasets used to generate semi-synthetic datasets discussed in Figure 6A-H. UMAPs of the downsampled 10k T cells (panel A), 1M T cells (panel B), keratinocyte (panel C) and macrophage (panel D) datasets (intermediate steps are shown in Figure S8 and detailed in methods). For each dataset, features were selected using either BigSur (selecting the top 10% of significant ϕ'), HVGs, SCT or FvF.



Supplementary Figure 11. Ranking of features for the four datasets used to generate semi-synthetic datasets discussed in Figure 6A-H. Ranking of features by FvF, SCT and HVGs plotted against the ranking of features by BigSur, for each dataset shown in Figure S9. To rank the features using BigSur, genes were first binned by ϕ' p-value, then ranked in order of decreasing ϕ'.

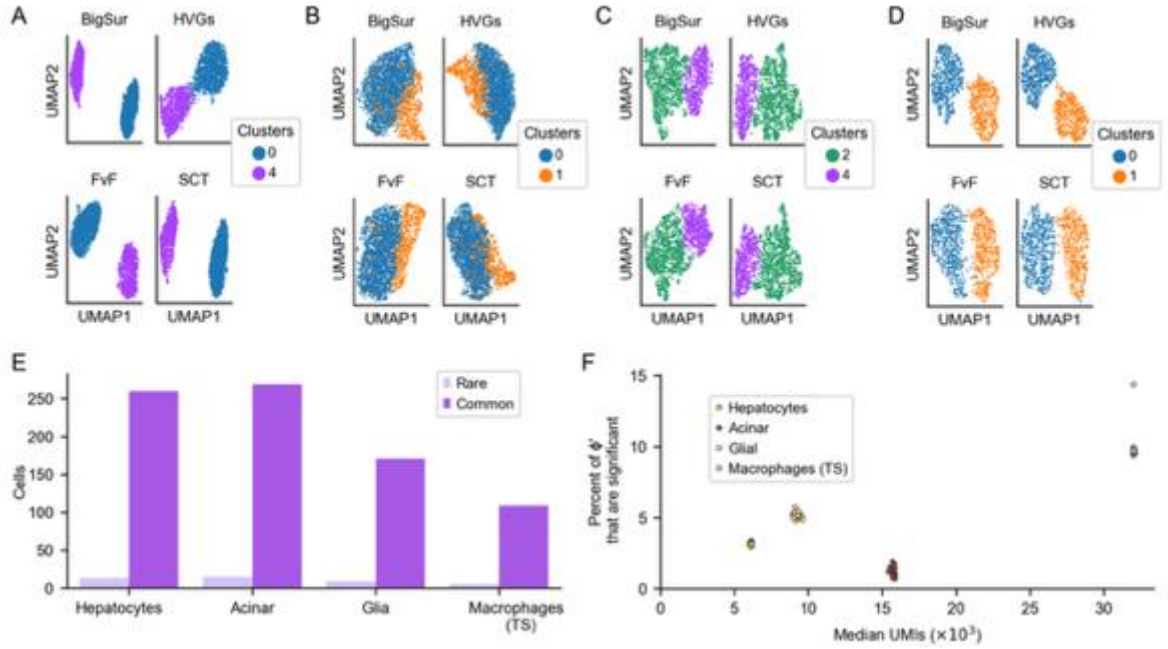


Supplementary Figure 12. Visualization of semi-synthetic dataset generation and their purity scores. A – D. UMAPs of intermediate steps (described in methods) for semi-synthetic data generation, along with an example of a semi-synthetic dataset, for the 10k T cell (panel A), 1M T cell (panel B), keratinocyte (panel C), and macrophage (panel D) datasets. Each UMAP was calculated from the top 10% of significant ϕ'. E. For each semi-synthetic dataset, features were selected using four different methods using their defaults (see main text for BigSur's cutoffs), clustered using the Leiden algorithm with 40 different starting seeds, and calculated the purity score of the rare cells. The first 10 datasets (0-9) are the datasets shown in Figure 6D – H. As in Figure 6, the dots represent medians, and the bars represent interquartile range.

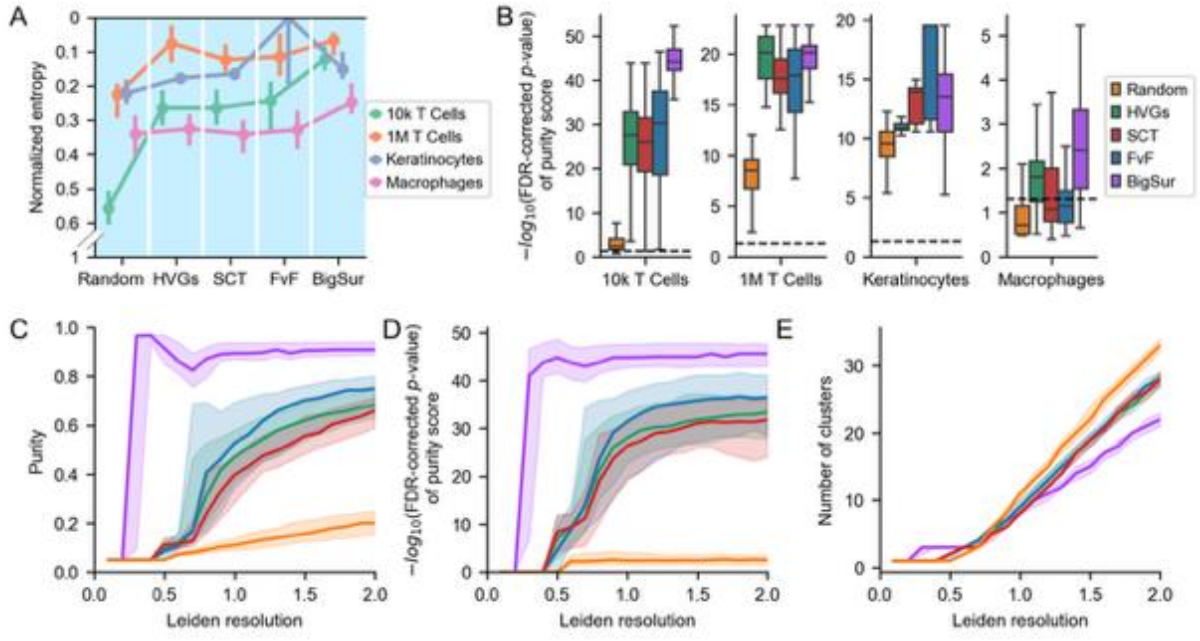


Supplementary Figure 13. Downsampling steps for the Tabula Sapiens datasets. UMAPs of downsampling steps of the liver, pancreas, eye and lymph node datasets from the Tabula Sapiens human cell atlas, see methods for details. For each UMAP , the top 10% of significant ϕ' were selected using BigSur and clustering was done using the Leiden algorithm (with resolution = 1 if not otherwise specified). A – C. UMAPs of the liver dataset. Cells are either colored by cluster assignments (panel A), donor IDs (panel B) or gene expression of PCK1 (panel C). D. UMAP of clusters 0, 3, 4, 12 and 14 from the liver dataset. E – G. UMAP of the pancreas dataset. Cells are either colored by cluster assignment (which was done using a resolution = 0.3; panel E), donor IDs (panel F) or gene expression of CLPS (panel G). H. UMAP of cluster 0 of the pancreas dataset. I – J. UMAPs of the eye dataset, with cells colored by either cluster assignment (panel I) or gene expression of GLUL (panel J). K – L. UMAP of

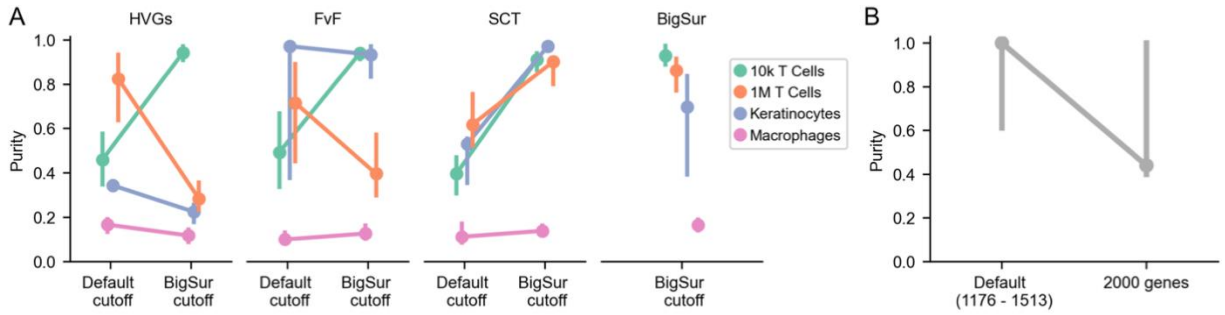
clusters 1, 5 and 11 of the eye dataset. Cells are colored by donor IDs (panel K) or cluster labels (panel L). M – N. UMAP of the lymph node dataset. Cells are colored by clusters (panel M) or expression of CD68. O – P. UMAP of cluster 19 from the lymph node dataset. Cells are colored by donor IDs (panel O) or cluster labels (panel P).



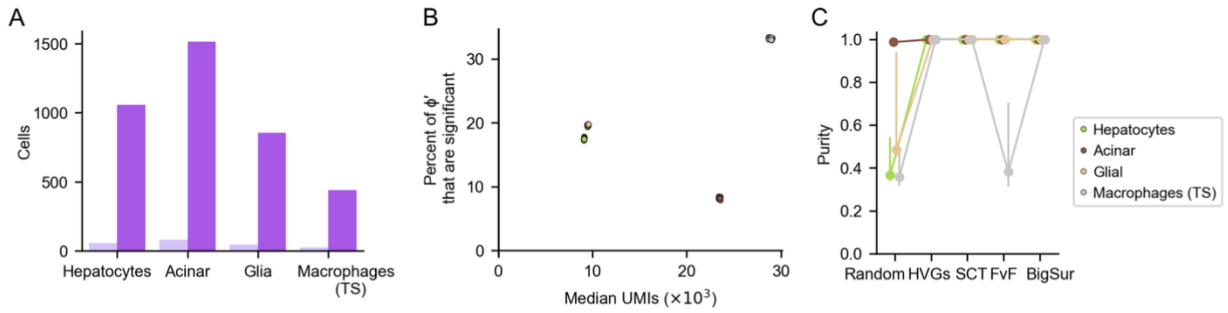
Supplementary Figure 14. Statistics and results of the Tabula Sapiens semi-synthetic datasets. The Tabula Sapiens (TS) datasets (the liver, pancreas, eye and lymph nodes datasets, see main text) were downsampled to only include two clusters of a cell type (hepatocytes, acinar cells, glia cells, and macrophages, respectively), detailed in the methods. A – D. UMAPs of the four downsampled datasets. For each dataset, features were selected, from which PCs and subsequently UMAP coordinates were calculated. The liver dataset clusters are shown in panel A; the pancreas dataset clusters are shown in panel B; the eye dataset clusters are shown in panel C; and the lymph node clusters are shown in panel D. E – F . Semi-synthetic datasets were generated from the TS datasets using the same procedure as in Figure 6 (see methods). For each semi-synthetic dataset, the number of rare and common cells for each semi-synthetic dataset is displayed in panel E, and the percent of ϕ' that are significant and median UMI/cell are displayed in panel F.



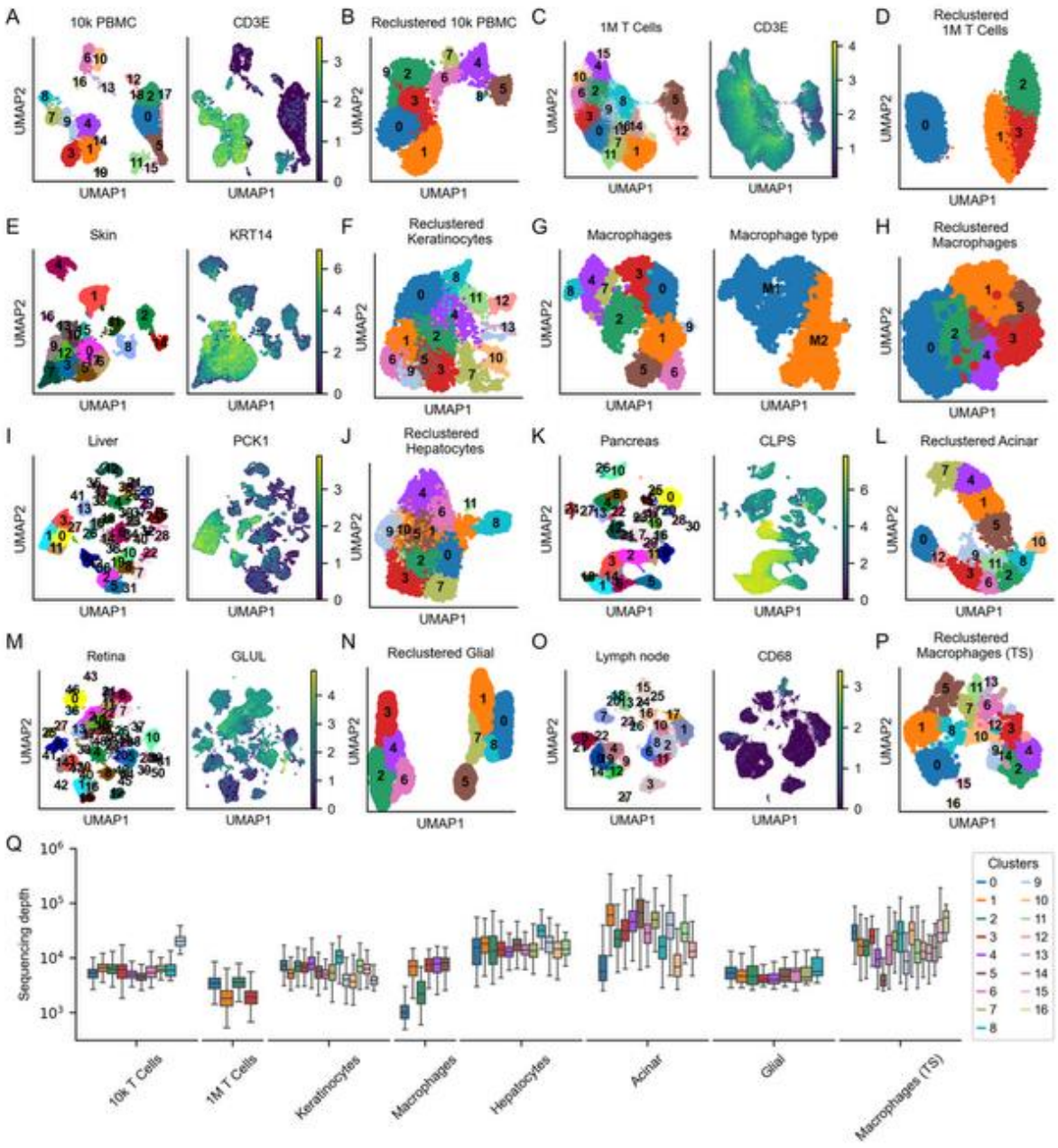
Supplementary Figure 15. Validation of purity score. For each semi-synthetic dataset whose purities are shown in Figure 6D-H, features were selected (using either HVGs, SCT, FvF or BigSur or at random) and clusters were assigned with 40 different Leiden starting seeds. A. The normalized entropy of the rare cells (see methods). B. p-values of the purity scores for each set of clusters (see methods). The dashed line represents p = 0.05. C-E. For each semi-synthetic generated from the 10k T cells dataset, features were selected using different methods and clustering was done using the Leiden algorithm, using 40 different starting seeds, with varying resolution. The resulting purity scores are shown in panel C, their p-values are shown in panel D and the number of clusters at each resolution is shown in panel E. Each plot displays the median and interquartile range (IQR) of the relevant measurement. The colors correspond to the feature selection method used, see panel B.



Supplementary Figure 16. Change in purity score when varying numbers of genes selected by other feature selection methods. A. Median and IQRs of purities of the semi-synthetic datasets discussed in Figure 6D-H, yielded by using each method's default cutoff (data from Figure 6H) or by limiting the number of genes selected to the number of genes BigSur selected ("BigSur cutoff"; see Figure 6C and main text). B. Purity scores of the macrophage (TS) semi-synthetic datasets with 10% rare to total cells yielded by using the default cutoffs chosen by BigSur (data from Figure 6J) or the 2000 genes with the highest ϕ' (without using p-value cutoff).

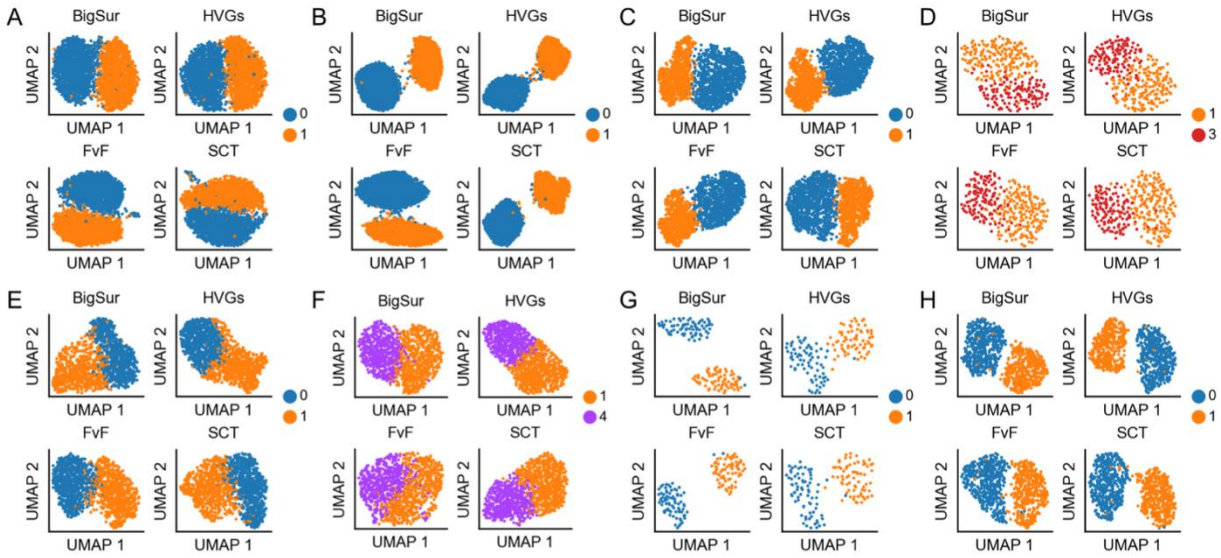


Supplementary Figure 17. Statistics and results of larger TS datasets. Large semi-synthetic datasets from the TS datasets were generated (see methods). A-B. The number of rare and common cells for each semi-synthetic dataset are shown in panel A, and the percent of ϕ' that are significant and median UMI/cell of these semi-synthetic datasets are shown in panel B. See panel C for legend. C. For each of the large semi-synthetic TS datasets, whose statistics are shown in panels H and I, features were selected, clusters were assigned using the Leiden algorithm, with 40 different Leiden starting seeds, and the purity scores were calculated, as in Figure 6. When using BigSur, the top 10% of significant ϕ' were selected.

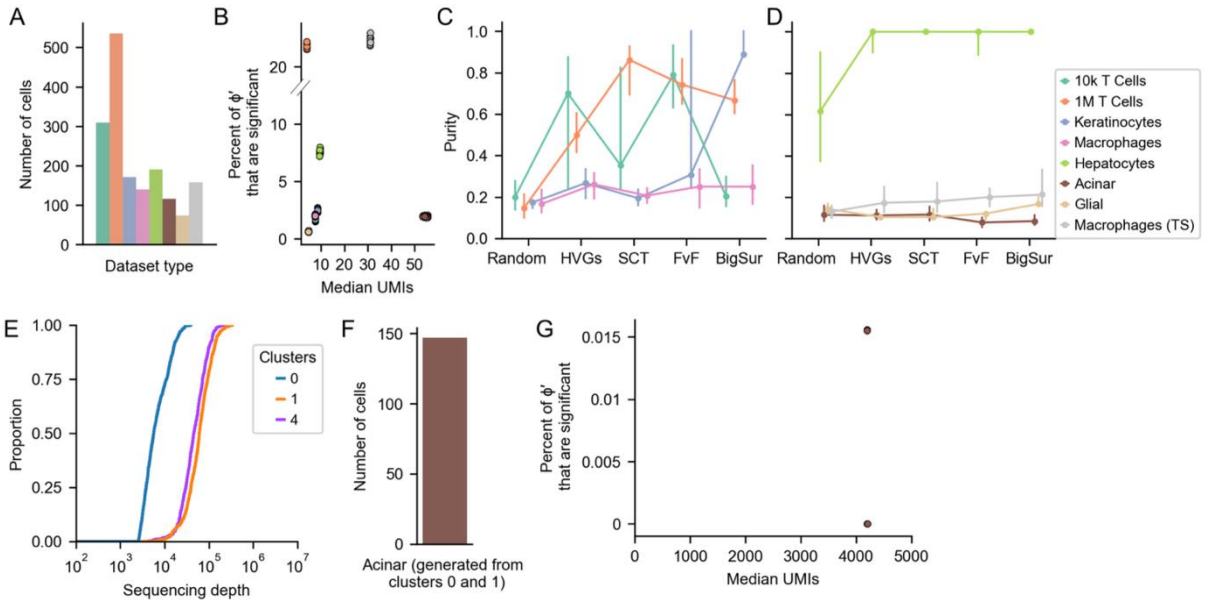


Supplementary Figure 18. Downsampling of eight datasets using HVGs. UMAPs of the eight datasets discussed in Figure 6 (the 10k T cells, 1M T cells, keratinocyte and macrophage datasets, along with the four datasets from the Tabula Sapiens atlas), along with UMAPs of subsets of cells from these datasets. For each UMAP, features were selected using HVGs (using defaults) and clustering was done using the Leiden algorithm (with resolution = 1). A. UMAPs of the 10k PBMC dataset, with cells colored by cluster assignment (left) or gene expression of CD3E (right). B. UMAP of cells assigned to clusters 1, 3, 4, 7, 8, 9 and 14 from the 10k PBMC dataset. C. UMAPs of the cells expressing CD3E in the 1M PBMC dataset. D. UMAP of cells assigned to clusters 4 and 10 displayed in panel C. E. UMAPs of the cells sequenced from patient ADSWT11 from the skin dataset. F. UMAP of cells assigned to clusters 0, 3, 5, 6, 7, 9, 10, 12, 13, 15 and 17 shown in panel E. G. UMAPs of macrophages from the macrophage dataset, excluding macrophages in the first stage of differentiation, progenitor cells, and repolarized macrophages (see methods for details). Cells are colored by cluster assignment (left) or macrophage polarization

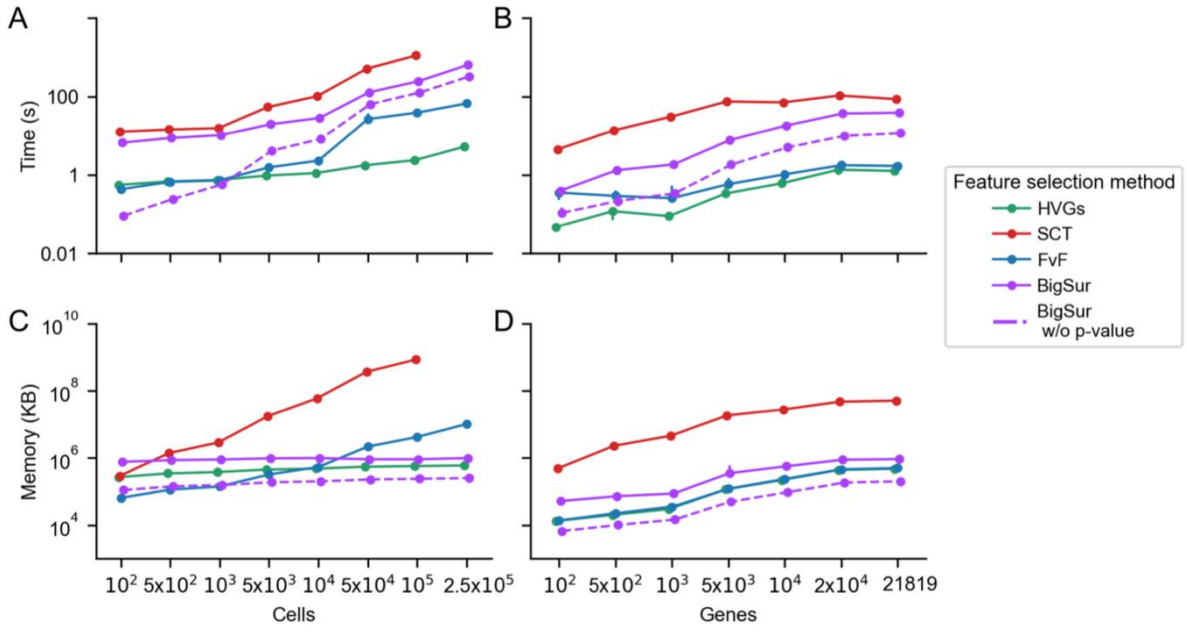
(right). H. UMAP of cells assigned to clusters 0 and 1 in the macrophage dataset. I. UMAPs of cells from the liver Tabula Sapiens (TS) dataset. J. UMAP of cells assigned to clusters 0, 1, 3, 11 and 27 in the liver dataset. K. UMAPs of cells from the pancreas TS dataset. L. UMAP of cells assigned to clusters 1, 2, 3, 5, 6, 14 and 18 in the pancreas dataset. M. UMAPs of cells from the retina TS dataset. N. UMAP of cells assigned to clusters 28 and 30 in the retina dataset. O. UMAPs of cells from the lymph node TS dataset. P. UMAP of cells assigned to cluster 15 in the lymph node TS dataset. Q. Sequencing depth (total UMI/cell) of all clusters in each dataset. Note the large differences in sequencing depth of clusters 0 and 1 in the macrophage and acinar datasets.



Supplementary Figure 19. Feature selection applied to eight downsampled datasets generated using HVGs. UMAPs of datasets which were used to generate semi-synthetic datasets. For each dataset, features were selected using BigSur (selecting the top 10% of significant ϕ'), HVGs, SCT or FvF. The cluster assignments for each dataset are displayed in Figure S15. A. UMAPs of T cells from the 10k PBMC dataset (cluster assignments shown in Figure S15B). B. UMAPs of T cells from the 1M PBMC dataset (cluster assignments in Figure S15D). C. UMAPs of keratinocytes from the skin dataset (cluster assignments in Figure S15F). D. UMAPs of macrophages from the macrophage dataset (cluster assignments in Figure S15H). E. UMAPs of hepatocytes from the liver TS dataset (cluster assignments in Figure S15J). F. UMAPs of acinar cells from the pancreas TS dataset (cluster assignments in Figure S15L). G. UMAPs of glial cells from the retina TS dataset (cluster assignments in Figure S15N). H. UMAPs of macrophages from the lymph node TS dataset (cluster assignments in Figure S15P).



Supplementary Figure 20. Results from semi-synthetic datasets generated using HVGs. For each of eight datasets, 20 semi-synthetic datasets were generated using HVGs (the intermediate steps are shown in Figure S20 and detailed in the methods). A. The number of cells in each semi-synthetic dataset (see panel D for legend). B. Percent of ϕ' that are significant plotted against the median UMI/cell of each semi-synthetic dataset. C – D. Purity scores of semi-synthetic datasets yielded using different feature selection methods. For each semi-synthetic dataset, either 2000 random features were selected or features were selected using HVGs, SCT, FvF or BigSur. Clusters were assigned using the Leiden algorithm (with resolution = 1) with 40 different random starting seeds and the purity was calculated. Points and bars denote median purities and interquartile range (IQRs) of purities, respectively. BigSur selected the top 10% of significant ϕ' for all semi-synthetic datasets except for those generated from the keratinocyte and 10k T cell datasets, for which the top 1% of significant ϕ' were selected. E. Empirical cumulative distribution function of the sequencing depths (total UMI/cell) of the acinar cells assigned to clusters 0, 1 and 4. F – G. For each of 20 semi-synthetic datasets generated from clusters 0 and 1 of the acinar dataset, the number of cells is shown in panel F, and the percent of ϕ' that are significant are plotted against the median UMI/cell are shown in panel G. For all except three datasets, in which two genes were found to have significant ϕ', no genes were found to have significant ϕ' (i.e. p-value < 0.05).



Supplementary Figure 21. Comparison of run time and memory usage of each method. The 1M PBMC dataset was downsampled to varying numbers of cells and genes and the total memory usage and run time of each feature selection was measured. For each plot, markers are medians and bars are interquartile ranges (IQRs).The time and memory usage of BigSur with (purple, solid line) and without (purple, dashed line) p-value calculation was measured. SCTransform was not run on datasets with 250,000 cells due to insufficient memory. A – B. Each method was timed for datasets with varying numbers of cells (panel A) and genes (panel B). C – D. The total memory usage of each method was measured for datasets with varying numbers of cells (panel C) and genes (panel D).

3.16 References

1. Das S, Rai A, Rai SN: Differential Expression Analysis of Single-Cell RNA-Seq Data: Current Statistical Approaches and Outstanding Challenges. Entropy (Basel) 2022, 24.

2. Junttila S, Smolander J, Elo LL: Benchmarking methods for detecting differential states between conditions from multi-subject single-cell RNA-seq data. Briefings in Bioinformatics 2022, 23.

3. Nguyen H, Tran D, Tran B, Pehlivan B, Nguyen T: A comprehensive survey of regulatory network inference methods using single cell RNA sequencing data. Brief Bioinform 2021, 22.

4. Simmons S: Cell Type Composition Analysis: Comparison of statistical methods. bioRxiv 2022:2022.2002.2004.479123.

5. Tam PPL, Ho JWK: Cellular diversity and lineage trajectory: insights from mouse single cell transcriptomes. Development 2020, 147.

6. Tritschler S, Büttner M, Fischer DS, Lange M, Bergen V, Lickert H, Theis FJ: Concepts and limitations for learning developmental trajectories from single cell genomics. Development 2019, 146.

7. Xie B, Jiang Q, Mora A, Li X: Automatic cell type identification methods for single-cell RNA sequencing. Comput Struct Biotechnol J 2021, 19:5874-5887.

8. Heumos L, Schaar AC, Lance C, Litinetskaya A, Drost F, Zappia L, Lucken MD, Strobl DC, Henao J, Curion F, et al: Best practices for single-cell analysis across modalities. Nat Rev Genet 2023, 24:550-572.

9. Li J, Cheng K, Wang S, Morstatter F, Trevino RP, Tang J, Liu H: Feature Selection: A Data Perspective. ACM Comput Surv 2017, 50:Article 94.

10. Jiang L, Chen H, Pinello L, Yuan GC: GiniClust: detecting rare cell types from single-cell gene expression data with Gini index. Genome Biol 2016, 17:144.

11. Brennecke P, Anders S, Kim JK, Kolodziejczyk AA, Zhang X, Proserpio V, Baying B, Benes V, Teichmann SA, Marioni JC, Heisler MG: Accounting for technical noise in single-cell RNA-seq experiments. Nat Methods 2013, 10:1093-1095.

12. Andrews TS, Hemberg M: M3Drop: dropout-based feature selection for scRNASeq. Bioinformatics 2018, 35:2865-2867.

13. Su K, Yu T, Wu H: Accurate feature selection improves single-cell RNA-seq cell clustering. Brief Bioinform 2021, 22.

14. Townes FW, Hicks SC, Aryee MJ, Irizarry RA: Feature selection and dimension reduction for single-cell RNA-Seq based on a multinomial model. Genome Biol 2019, 20:295.

15. Tyler SR, Lozano-Ojalvo D, Guccione E, Schadt EE: Anti-correlated Feature Selection Prevents False Discovery of Subpopulations in scRNAseq. Nature Communications 2024, 15.

16. Lall S, Ghosh A, Ray S, Bandyopadhyay S: sc-REnF: An entropy guided robust feature selection for single-cell RNA-seq data. Brief Bioinform 2022, 23.

17. Stuart T, Butler A, Hoffman P, Hafemeister C, Papalexi E, Mauck WM, 3rd, Hao Y, Stoeckius M, Smibert P, Satija R: Comprehensive Integration of Single-Cell Data. Cell 2019, 177:1888-1902 e1821.

18. Satija R, Farrell JA, Gennert D, Schier AF, Regev A: Spatial reconstruction of single-cell gene expression data. Nat Biotechnol 2015, 33:495-502.

19. Hafemeister C, Satija R: Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. Genome Biol 2019, 20:296.

20. Germain PL, Sonrel A, Robinson MD: pipeComp, a general framework for the evaluation of computational pipelines, reveals performant single cell RNA-seq preprocessing tools. Genome Biol 2020, 21:227.

21. Zhao R, Lu J, Zhou W, Zhao N, Ji H: A systematic evaluation of highly variable gene selection methods for single-cell RNA-sequencing. bioRxiv 2024.

22. Silkwood K, Dollinger E, Gervin J, Atwood S, Nie Q, Lander AD: Leveraging gene correlations in single cell transcriptomic data. BMC Bioinformatics 2024, 25:305.

23. Hao Y, Hao S, Andersen-Nissen E, Mauck WM, 3rd, Zheng S, Butler A, Lee MJ, Wilk AJ, Darby C, Zager M, et al: Integrated analysis of multimodal single-cell data. Cell 2021, 184:3573-3587 e3529.

24. Wolf FA, Angerer P, Theis FJ: SCANPY: large-scale single-cell gene expression data analysis. Genome Biol 2018, 19:15.

25. Choudhary S, Satija R: Comparison and evaluation of statistical error models for scRNAseq. Genome Biology 2022, 23:27.

26. Heimberg G, Bhatnagar R, El-Samad H, Thomson M: Low Dimensionality in Gene Expression Data Enables the Accurate Extraction of Transcriptional Programs from Shallow Sequencing. Cell Syst 2016, 2:239-250.

27. Lenz M, Muller FJ, Zenke M, Schuppert A: Principal components analysis and the reported low intrinsic dimensionality of gene expression microarray data. Sci Rep 2016, 6:25696.

28. Lukk M, Kapushesky M, Nikkila J, Parkinson H, Goncalves A, Huber W, Ukkonen E, Brazma A: A global map of human gene expression. Nat Biotechnol 2010, 28:322-324.

29. Miragaia RJ, Gomes T, Chomka A, Jardine L, Riedel A, Hegazy AN, Whibley N, Tucci A, Chen X, Lindeman I, et al: Single-Cell Transcriptomics of Regulatory T Cells Reveals Trajectories of Tissue Adaptation. Immunity 2019, 50:493-504 e497.

30. Hughes G: On the mean accuracy of statistical pattern recognizers. IEEE Transactions on Information Theory 1968, 14:55-63.

31. Zimek AS, E.; Kriegel, H. P.;: A survey on unsupervised outlier detection in high-dimensional numerical data. Statistical Analysis and Data Mining 2012, 5:363-387.

32. Menon M, Mohammadi S, Davila-Velderrain J, Goods BA, Cadwell TD, Xing Y, Stemmer-Rachamimov A, Shalek AK, Love JC, Kellis M, Hafler BP: Single-cell transcriptomic

atlas of the human retina identifies cell types associated with age-related macular degeneration. Nat Commun 2019, 10:4902.

33. Traag VA, Waltman L, van Eck NJ: From Louvain to Leiden: guaranteeing well-connected communities. Sci Rep 2019, 9:5233.

34. Bahar Halpern K, Tanami S, Landen S, Chapal M, Szlak L, Hutzler A, Nizhberg A, Itzkovitz S: Bursty gene expression in the intact mammalian liver. Mol Cell 2015, 58:147-156.

35. Beal J: Biochemical complexity drives log-normal variation in genetic expression. Engineering Biology 2017, 1:55-60.

36. Zimek A, Schubert E, Kriegel H-P: A survey on unsupervised outlier detection in high-dimensional numerical data. Statistical Analysis and Data Mining: An ASA Data Science Journal 2012, 5:363-387.

37. Lause J, Berens P, Kobak D: Analytic Pearson residuals for normalization of single-cell RNA-seq UMI data. Genome Biol 2021, 22:258.

38. Cornish EA, Fisher RA: Moments and Cumulants in the Specification of Distributions. Revue de l'Institut International de Statistique / Review of the International Statistical Institute 1938, 5:307-320.

39. Benjamini Y, Hochberg Y: Controlling the False Discovery Rate - a Practical and Powerful Approach to Multiple Testing. Journal of the Royal Statistical Society Series B-Statistical Methodology 1995, 57:289-300.

40. Guerrero-Juarez CF, Lee GH, Liu Y, Wang S, Karikomi M, Sha Y, Chow RY, Nguyen TTL, Iglesias VS, Aasi S, et al: Single-cell analysis of human basal cell carcinoma reveals novel regulators of tumor growth and the tumor microenvironment. Sci Adv 2022, 8:eabm7981.

41. Tabula Sapiens C, Jones RC, Karkanias J, Krasnow MA, Pisco AO, Quake SR, Salzman J, Yosef N, Bulthaup B, Brown P, et al: The Tabula Sapiens: A multiple-organ, single-cell transcriptomic atlas of humans. Science 2022, 376:eabl4896.

42. Kumar U, V. K, and Kapur JN: NORMALIZED MEASURES OF ENTROPY. International Journal of General Systems 1986, 12:55-69.

43. Bulmer MG: On Fitting the Poisson Lognormal Distribution to Species-Abundance Data. Biometrics 1974, 30:101-110.

44. Wolock SL, Lopez R, Klein AM: Scrublet: Computational Identification of Cell Doublets in Single-Cell Transcriptomic Data. Cell Syst 2019, 8:281-291 e289.

45. Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, et al: Array programming with NumPy. Nature 2020, 585:357-362.

46. Pedregosa FV, G; Gramfort, A; Michel, V; Thirion, B; Grisel, O; Blondel, M; Prettenhofer, P; Weiss, R; Dubourg, V; Vanderplas, J; Passos, A; Cournapeau, D; Brucher, M; Perrot, M; Duchesnay, E;: Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 2011, 12:2825-2830.

47. Mann HB, Whitney DR: On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. The Annals of Mathematical Statistics 1947, 18:50-60.

48. Bengtsson H: profmem: Simple Memory Profiling for R. 2020.

49. Carvalho K, Rebboah E, Jansen C, Williams K, Dowey A, McGill C, Mortazavi A: Uncovering the Gene Regulatory Networks Underlying Macrophage Polarization Through Comparative Analysis of Bulk and Single-Cell Data. bioRxiv 2021.

CHAPTER 4: A NOTE ON NOISE ESTIMATION IN SINGLE CELL RNA-SEQUENCING DATA

BigSur relies heavily on a gene's estimated coefficient of variation (CV; *c* in previous chapters), both in the calculation of the modified corrected Pearson residuals and in the assignment of *p*-values for either the modified corrected Fano factor or correlation coefficients. In the original manuscript, we estimated a single value of CV that we applied to every gene. This allowed us to obtain meaningful results, but there is no reason to assume that the noise in every gene's expression ought to be the same. To the contrary, we might generally expect that genes which are expressed in high amounts and transcribed often (so-called "housekeeping" genes) would have lower noise in expression than those that are only occasionally expressed.

As such, we wanted to see if there was a more appropriate way to estimate CV. While the Fano factor is expected to have a value of one from any Poisson-distributed data, this is not the case for the CV, which is expected to have an inversely proportional relationship with the mean. We figured, then, that we may be able to empirically estimate a null relationship between the mean expression of a gene and the CV. To test this, we analyzed scRNAseq data from mouse embryonic stem cells (mESCs) which were treated with 5'-iodo-2'-deoxyurine (IdU), a drug noted to globally increase the noise in gene expression[1], to a DMSO-treated control sample. This dataset is relatively homogeneous, making it a strong candidate for testing. After isolating clusters of cells from each dataset, for each gene in each sample, we computed the value of CV which would produce a modified-corrected Fano factor equal to one. From there, we used. The rationale is that most genes are not differentially expressed across the cell states present in the system. Their behavior, therefore, mostly reflects intrinsic variability, and the median across genes provides an empirical estimate of the null relationship between mean expression and CV.

Additionally, most technical noise differences should be accounted for by fitting to the modified corrected Fano factor (i.e., using Pearson residuals).

We found that the mean-CV relationship was well fit by:

$$CV_{fitted} = \sqrt{\frac{\eta}{\mu} + \theta} \quad (Eq. 1)$$

where μ and both η and θ are parameters of the fit (Fig. 1). Fitting the null mean-CV relationship in this way correctly predicts that the noise in the IdU treated cells is higher than those treated with DMSO (Fig. 2).

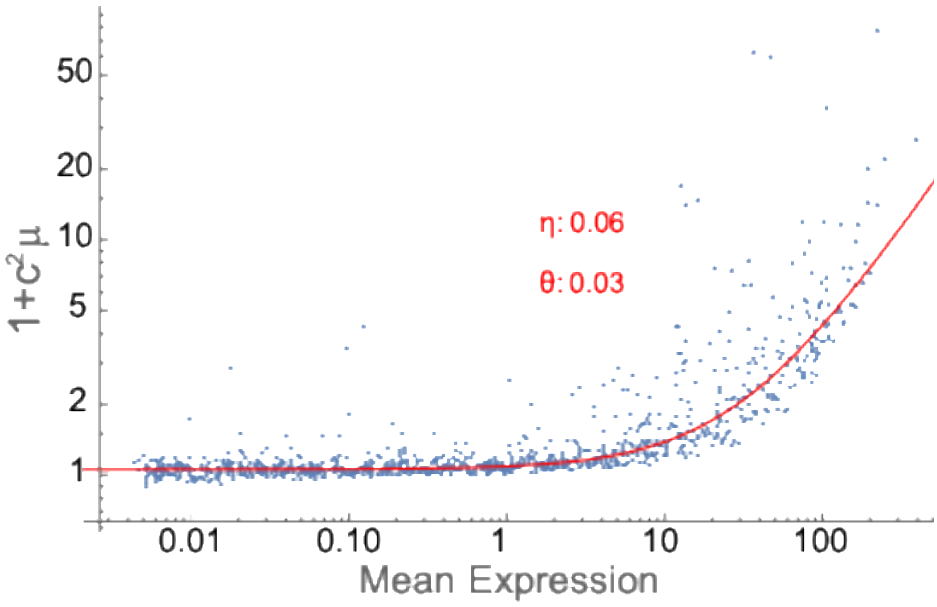


Figure 1. The null relationship between CV and mean expression is well modeled by $\sqrt{\frac{\eta}{\mu} + \theta}$. Blue points are sampled genes (see Methods). Red shows the fitted curve and parameter values. The y-axis shows the expected per-gene Fano factor.

We arrived at this fit empirically, but it is not a novel model of gene expression noise. With parameters treated as constants, telegraph models with bursty gene expression can be written in this form [2–4]. Note, these models may be used to describe either protein or mRNA

expression as, although the biology differs, the mathematical structure is shared. In both cases, the system alternates between an inactive and active state. In the active state, the arrival of production machinery (i.e., transcription or translation proteins) is described by a Poisson process, after which production occurs in discrete bursts, the size of which is drawn from some distribution (commonly geometric) [5]. The rates of activation and deactivation are tied to promoter binding in transcription models and mRNA lifetime in protein models. These models would support η and θ encompassing transcriptional bursting parameters and promoter noise respectively [4].

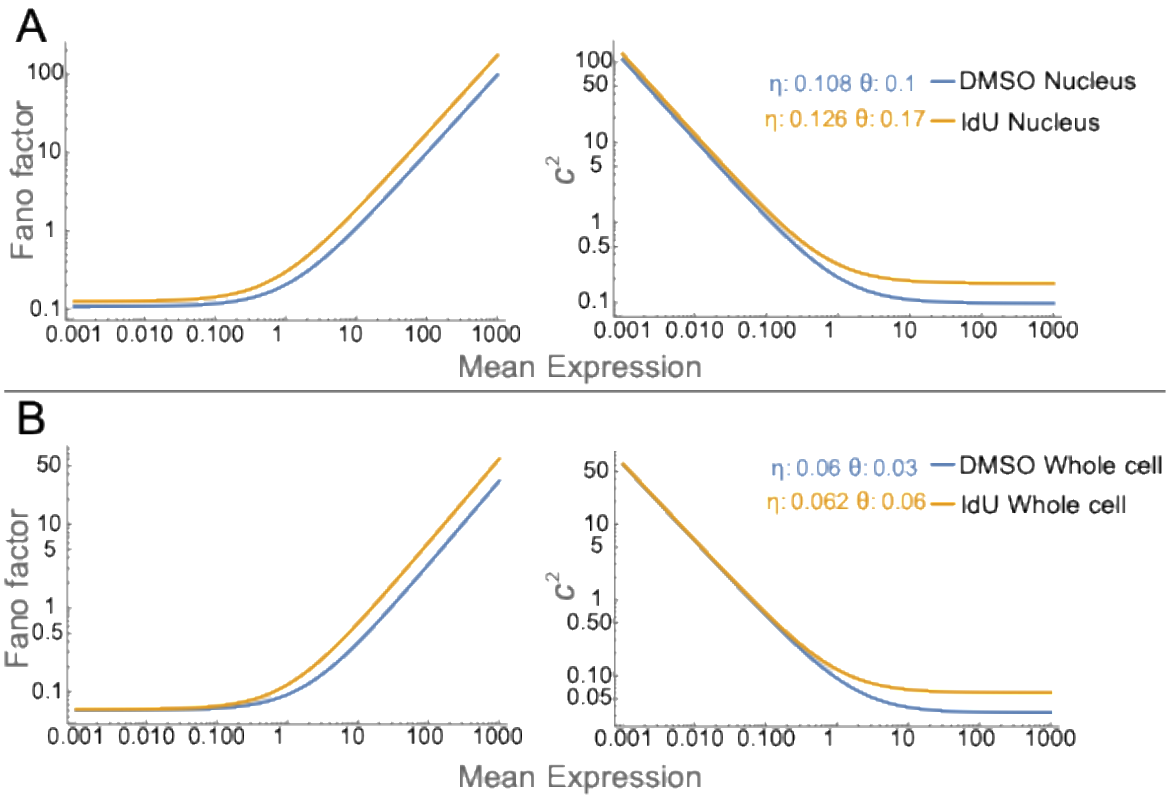


Figure 2. The empirical null model relating CV to mean expression predicts higher noise in IdU-treated cells than in controls.
A. Models fit on sequenced nuclei. B. Models fit on sequenced whole cells. η and θ represent the parameters to the fitted model (Eq.1).

In the next chapter and all future works, we employ an updated version of BigSur which estimates the null CV for each gene from its mean expression. Anecdotally, this appears to filter out more lowly-expressed genes than before.

4.1 Method

4.1.1 Data Preparation

Counts matrices created through either whole cell (WC) or nucleus (Nuc) sequencing for the IdU and DMSO treated mESCs were retrieved directly from the authors of Desai *et al*[1].

Separately in each of the four datasets, pseudogenes were removed using a self-curated list. Genes that were expressed in fewer than 50 cells were then removed from analysis.

Although these cells were expected to be largely homogeneous, clustering was performed and certain clusters removed to further reduce heterogeneity. BigSur was used to select features and calculate the modified corrected Pearson residuals matrix (Benjamini Hochberg $p < 0.05$ for all datasets; Fano thresholds - DMSO WC:1.7, IdU WC:1.7, DMSO Nuc: 1.3, IDU Nuc: 1.44). The top 20 principal components of those selected features (calculated on the modified corrected Pearson residuals matrix) were used to construct a shared nearest neighbor graph, which was then partitioned by Leiden clustering (resolution - WC datasets: 0.2, Nuc datasets: 0.15). WC clusters containing with under 10,000 total UMIs on average were removed from further analysis. Nuc clusters containing fewer than 500 cells were also removed.

4.1.2 Fitting the mean-CV relationship

We would like to find the values of CV such that the modified corrected Fano factor equals its null expectation of one. To do this, after calculating the cell-specific expectation matrix (E_{ij}) [6], we find the value c such that:

$$f_i(c) = \sum_j \frac{\left(\mathrm{O_{ij}} - E_{ij}\right)^2}{E_{ij}(1 + c^2 E_{ij})} - (n-1) = 0$$

where O_{ij} is the original counts matrix and n is the number of cells. This is solved numerically by the secant method.

The relationship between per-gene expected CV and mean expression was modeled as:

$$c^2(\mu) = \frac{\eta}{\mu} + \theta$$

Equivalently, fitting was performed on $1 + c^2(\mu)$ against $1 + \eta + \theta\ \mu$. Parameters were estimated in log space using L1 regression.

For computational efficiency, fits were performed on a subsample of up to 1,000 genes drawn from each 0.5-unit bin of log mean expression

4.2 References

1. Desai RV, Chen X, Martin B, Chaturvedi S, Hwang DW, Li W, et al. A DNA repair pathway can regulate transcriptional noise to promote cell fate transitions. Science. American Association for the Advancement of Science; 2021;373:eabc6506. https://doi.org/10.1126/science.abc6506

2. Pedraza JM, Paulsson J. Effects of Molecular Memory and Bursting on Fluctuations in Gene Expression. Science. American Association for the Advancement of Science; 2008;319:339–43. https://doi.org/10.1126/science.1144331

3. Elgart V, Jia T, Fenley AT, Kulkarni R. Connecting protein and mRNA burst distributions for stochastic models of gene expression. Phys Biol. 2011;8:046001. https://doi.org/10.1088/1478-3975/8/4/046001

4. Chen M, Luo S, Cao M, Guo C, Zhou T, Zhang J. Exact distributions for stochastic gene expression models with arbitrary promoter architecture and translational bursting. Phys Rev E. American Physical Society; 2022;105:014405. https://doi.org/10.1103/PhysRevE.105.014405

5. Kumar N, Singh A, Kulkarni RV. Transcriptional Bursting in Gene Expression: Analytical Results for General Stochastic Models. PLOS Comput Biol. Public Library of Science; 2015;11:e1004292. https://doi.org/10.1371/journal.pcbi.1004292

6. Silkwood K, Dollinger E, Gervin J, Atwood S, Nie Q, Lander AD. Leveraging gene correlations in single cell transcriptomic data. BMC Bioinformatics. 2024;25:305. https://doi.org/10.1186/s12859-024-05926-z

CHAPTER 5: GENE CORRELATION STRUCTURE REVEALS CELL STATE TRANSITION DRIVERS IN SINGLE-CELL RNA SEQUENCING DATA

5.0 Foreword

Now, it is finally time to discuss how modified corrected Pearson correlation coefficients may be used to detect cell state transition driver genes. This text is prepared for future submission for publication, so elements are repeated from earlier parts of this thesis (i.e., the introduction). Here, I performed all data analysis and developed the technique. The read alignment, quality control, and original cell filtering steps were performed by Chan Hee Mok in the laboratory of Ralph Marcucio (UCSF). Diane Hu, also in the Marcucio lab, performed the experiments on *Gallus gallus* and wrote the section on their methods.

5.1 Introduction

As this text is prepared for future submission for publication, the following is an abridged version of the Introduction of this thesis, comprised of duplicated text from that section.

After the discovery of the lac operon by Jacob and Monod[1], it became increasingly evident that a gene's expression could itself regulate the expression of other genes. Further, it was noted that these genes could form arbitrarily complex, feedback loop-enriched networks with one another i.e., gene regulatory networks (GRN)[2]. In the steady state, high-dimensional dynamical networks such as these were predicted to give rise to relatively few stable attractors in gene expression space[2,3]. In other words, boundaries imposed by the GRN constrain the available gene expression space in which a cell can explore by its intrinsic stochastic fluctuations in gene expression alone. These stable attractors were proposed to define cell states.[2,3]

Under this model, for a cell to transition in state, it would first need to overcome the barriers placed by the GRN. The expression of any given gene is not constant even among cells of the same state but instead fluctuates stochastically leading to heavy-tailed distributions of transcripts and proteins[4–7]. Indeed, it has been demonstrated that this "intrinsic noise" can lead to stochastic differentiation events[8–12]. Of course, cell state transitions are not completely random but can be influenced by external stimuli[13–15].

As such, intrinsic noise and external stimuli, individually or in coordination, provide the means for the cell to transition from one attractor to another. These signals are thought to directly alter the GRN landscape itself, destabilizing the prior attractor state and facilitating the transition into another[16,17]. This is known as a critical transition: an attractor in multi-stable state space destabilizes until the cell passes the tipping point.

Critical transitions have been studied in a variety of different fields to better understand that lead to and follow catastrophic phenomena in nature and social situations[18–32]. Ecologists study tipping points to forecast how and when changes in environmental conditions, often due to human activity, will lead to major ecological shifts or collapse [23,28,29]; including studies to predict the point in which deforestation of the Amazon rainforest will lead to the destabilization of the hydrological cycle[19], the conditions which lead to the irreversible desertification of previously fertile biomes[18], and the point where rapidly changing climate conditions will lead to loss of ocean ecosystems[32]. Climatologists study tipping points to predict when catastrophic environmental conditions, such as the loss or reversal of ocean currents[21] or abrupt shifts in global climate[24–27] will occur. Economists study tipping points to determine conditions that lead to economic collapse[30] and epidemiologists study the conditions that lead to epidemics and disease re-emergence[31].

Though these systems are quite different conceptually, the early warning signs of their approach to the tipping point are often consistent[33]. First is the observation of increasing variance as the alternative attractor state becomes more accessible. Second is a "critical slowing down" in which the system becomes restricted from returning to its prior state after a perturbation. In single cell transcriptomics, where measurements are collected from a static timepoint, these early warning signs manifest as increased heterogeneity among cells and increased correlation between a subset of genes[16].

Measuring the change in correlations across a transition trajectory (either using time-series collection or trajectory inference) would, in theory, allow for the identification of transition driver genes; we would expect that these genes would "peak" in correlation strength with another subset of genes immediately preceding the tipping point. However, conventional correlation methods perform poorly on scRNAseq data, leading to many false positive results.

We previously developed BigSur[34] to obtain statistically calibrated gene-gene correlations from scRNAseq data. Here we asked whether these correlations, tracked across a transition trajectory, could identify candidate cell state transition driver genes. In this work, we provide evidence that correlation peaks can identify transition driver genes in both linear and binary differentiation events and utilize that methodology to identify novel transition driver genes in craniofacial development.

5.2 Results

5.2.1 Genes with correlation peaks form modules enriched for epithelial-mesenchymal transition annotation

Although scRNAseq measures all cells at a single timepoint, the captured population of cells is distributed asynchronously along a set of expressional trajectories. A sample collected

when cells are, on average, near the tipping point will, therefore, also contain cells which have already undergone the critical transition and settled into a new stable state. We would expect such a sample to show elevated correlation, not only among the transition driver genes, but also between those drivers and markers of the new cell state and between the various new cell state markers (Figure 1A). At a later timepoint, once most cells have entered the new state, driver-driver and driver-marker correlations are expected to decay since these fluctuations are alone present about the transition. State-specific correlations (i.e., marker-marker correlations), on the other hand, should persist, sustained by ordinary co-fluctuation within a stable expression program.

Under these expectations, a set of time series scRNAseq data which encapsulates the course of a cell state transition before and after the critical transition should enable us to uncover modules of candidate cell state transition driver genes, coupled with the markers of the state that the cells are transitioning towards. To test this, we analyzed scRNAseq data of MCF10A cells undergoing TGF-β induced epithelial-mesenchymal transition (EMT) across eight days (D1-D8) of treatment with the mesenchymal state appearing at D8[35]. BigSur was used to calculate modified-corrected correlation coefficients, hereafter referred to as “correlation coefficient” or simply “correlation”, on three different datasets representing different timepoints (D0, D4, D8) of the TGF-β treatment.

Following this, the correlation coefficient trajectory for each gene-gene pair on the subset of human transcription factors[36] was calculated and 1,325 peaks(332 unique genes) which had a prominence (the correlation at the peak minus the larger of the two endpoint correlations) above a correlation threshold of 0.05 at day 4 were collected. An adjacency graph describing the

peaks shared between collected genes was created and the walk trap algorithm[37] was used to identify more tightly connected modules of peaking genes (see Methods).

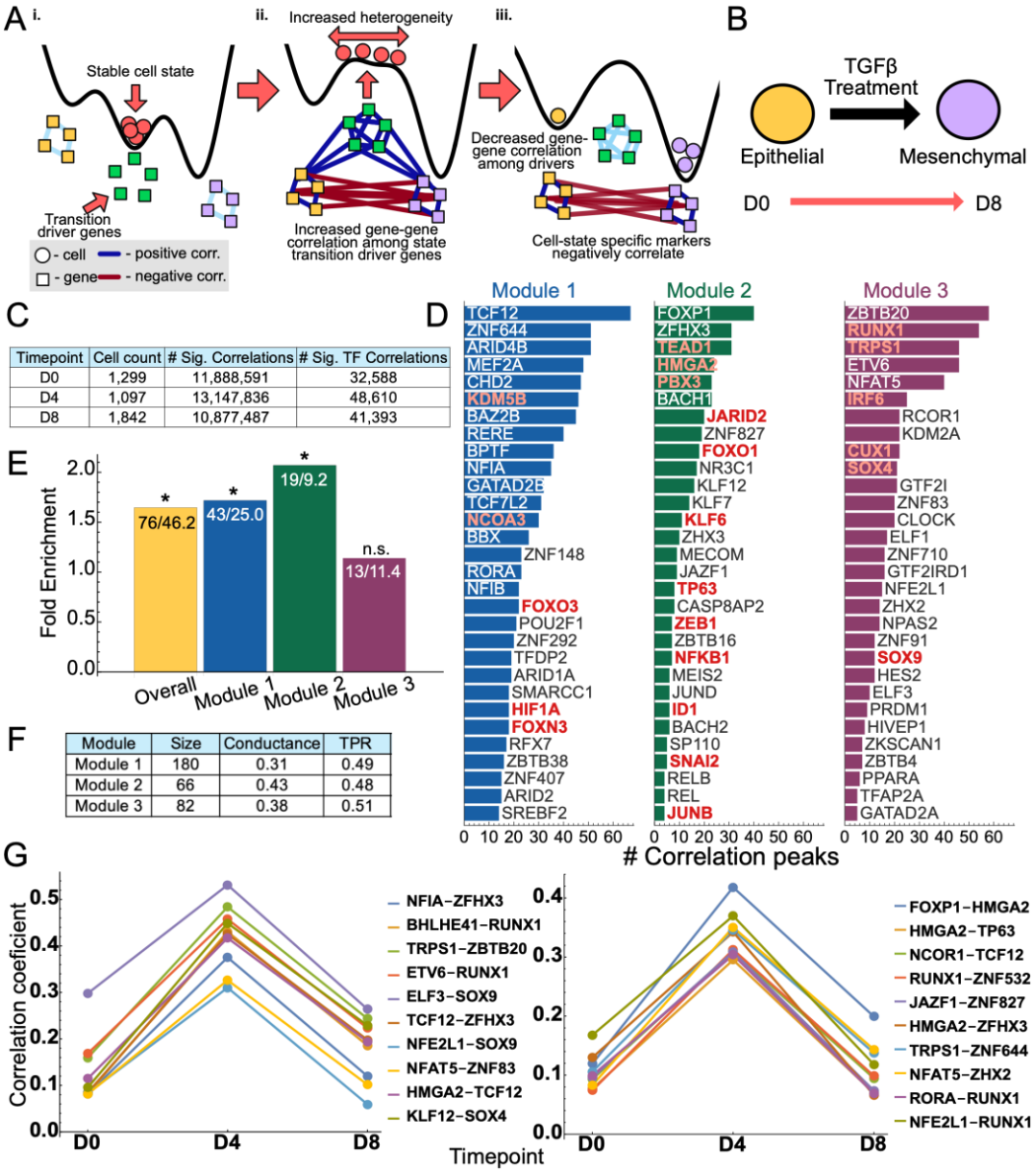


Figure 1. Gene-gene correlation peaks identify modules of transcription factors enriched for known EMT genes.

A. Tipping point theory and its relationship to gene-gene correlations from scRNAseq data. B. Schematic of EMT. C. Descriptive statistics describing the dataset and output from BigSur correlation analysis. D. Modules containing more than 10 genes. Genes annotated in dbEMT 2.0 are shown in red/salmon. Bar length gives the number of correlation peaks per gene. Only the 40 genes with the most peaks in each module are shown. E. Fold enrichment for annotated EMT genes. Star indicates $p < 0.05$. Numbers on bars show the ratio of annotated EMT genes to number expected from a random draw. F. Module-level descriptive statistics (TPR, triangle participation ratio). G. The 20 gene pairs with the largest correlation

peak prominence overall, split across two panels and ordered by decreasing prominence within each. Pairs in the left panel have higher prominence than those in the right.

This resulted in the identification of three modules which contained more than ten genes (Figure 1D). Each of the three modules contained genes annotated in the dbEMT 2.0 database[38], with two of them (Module 1 and Module 2) being significantly enriched for annotated genes (fold changes of 1.72, 2.07 and *p*-values of 4.6×10^{-4} , 8.4×10^{-4} respectively). The combined set of genes was also significantly enriched for annotated genes (fold change = 1.64, $p = 2.4 \times 10^{-8}$), indicating that correlation analysis alone can identify transition-relevant genes. Modules were assessed using conductance and triangle participation ratio (TPR) measurements (Figure 1F)[39]. Conductance is a measure of module interconnectivity, describing the fraction of edges for each gene which leave the module. Modules are expected to have a small conductance if they describe multiple orthogonal transitions in the data. TPR is a measure of module shape, describing the fraction of genes which form a triangle of interconnectivity (i.e., A←→B←→C←→A). The conductance measurements indicate that the modules are highly interconnected (i.e., are likely derived from related or identical transitional trajectories).

This method is best suited to identifying candidate transition driver genes for experimental testing. We suggest two complementary rankings.

The first ranks genes by the number of correlation peaks they exhibit (Figure 1D). A gene peaking with many partners may be implicated in the expression of many other genes relevant to the transition. We would further expect driver genes to accumulate more peaks than marker genes, since marker–marker correlations reflect a stable expression program and should persist, whereas driver–driver correlations are confined to the neighborhood of the tipping point and decay once cells enter the new state. The limitation is that the significance of any gene–gene

correlation depends on the expression levels of both genes, so the most highly connected genes may simply be the most highly expressed rather than the most important to the transition. Additionally, the number of peaks per gene is directly related to the prominence threshold used to identify peaks in correlation, an arbitrarily selected value which will vary from dataset-to-dataset.

The second method ranks gene pairs by peak prominence and identifies genes recurring among the top-ranked pairs (Figure 1G). The rationale parallels methods for detecting transitioning cells: pairs showing the largest transient rise in correlation carry the strongest evidence of tipping-point behavior. Ranking at the pair level rather than aggregating per gene (for example, by summing prominences across a gene's peaks) avoids the expression dependence described above, since aggregate measures scale with peak count. Correlated expression between a pair may also imply a regulatory relationship such as a shared upstream regulators or positive feedback. As such, the pairs themselves are interpretable as hypotheses.

Key EMT genes occupy the top ranks of both lists (Figure 1D,G). For example, *HMGA2* is ranked fourth for number of peaks in module two and four times in the list of overall top twenty prominence gene-gene pairs. *HMGA2* is a known upstream regulator of TGF-β induced EMT, providing expressional control over *SNAI1*, *SLUG*, and *TWIST1*[40]. Additionally, *HMGA2* is shown to have a high prominence pairing with *TP63*. *HMGA2* has been found within *TP63* immuno-precipitated complexes by immunoprecipitation mass spectrometry in ESCC cells[41]. *SOX4* and *SOX9*, genes with observed TGF-β-coupled expression[42], are both found with high peak number rankings in module three and within the top ten prominence gene-gene pairs. *SOX4* has been described as a highly upstream, master regulator of EMT[42]. The strong *SOX4-KLF12* peak is itself interesting. *SOX4* has been noted to regulate the expression of

EZH2[42], a gene which has been hypothesized to repress *KLF12* activity in mouse prostate cancer models[43]. A peak in correlation lends some credence to a model of negative feedback between *KLF12* and *SOX4 (i.e., KLF12* activates *SOX4* which creates negative feedback on *KLF12* through *EZH2* activation), though, to the best of our knowledge, there is no known direct relationship between the two. *SOX9* and *ELF3*, molecules which have been shown to directly interact with one another[44], have the fifth highest prominence among transcription factor pairs. Finally, *RUNX1* is ranked third overall in peak number and appears five times in the list of top twenty prominence pairs. *RUNX1* is known to stabilize the epithelial phenotype in MCF10A cells and its expression is lost during TGF-β-induced EMT[45].

5.2.2 Correlation peak analysis identifies distinct cell state transition modules in bifurcative neuromesodermal progenitor cell differentiation

EMT is a cell state transition that travels in a single direction: epithelial cells become mesenchymal cells or vice versa. Oftentimes trajectories are more complex, and a set of progenitor cells have the capability to enter multiple differentiated states.

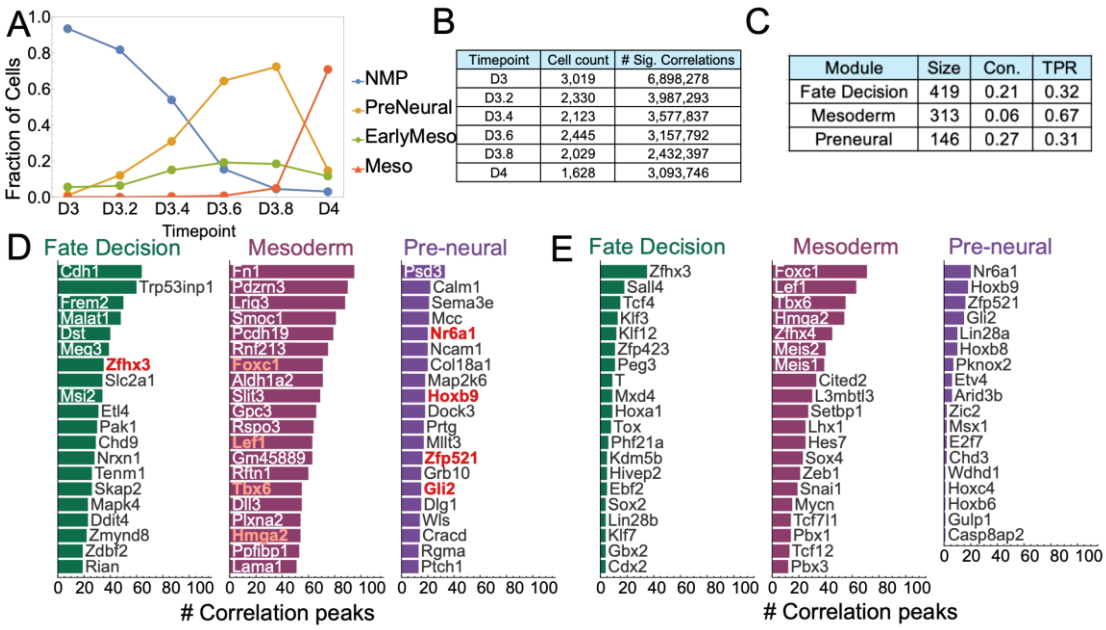
To test whether correlation peaks can identify transition drivers in a multi-directional cell state transition, we analyzed ESC-derived mouse mesodermal progenitors (NMPs) differentiating toward either a pre-neural or mesodermal fate[46]. Cells had been sampled at six equally spaced timepoints between days 3 and 4 (D3, D4) of treatment, and the analysis was restricted to cells annotated by the authors of Fontaine *et al* [46] as NMP, pre-neural, early mesoderm, or mesoderm. The timeframe of the transition differs between the two cell fates, with pre-neural cells peaking in population at D3.8 of treatment while mesoderm cells do not substantially appear until D4 (Figure 2A). BigSur was used to analyze correlations, and the correlation trajectory for each gene-gene pair was calculated. Using a prominence threshold of 0.03, we

collected 7,176 peaks composed of 1,999 unique genes. Seven modules above size 10 were identified in the manner previously described (Fig. 2D,E; Supp. Fig. 1).

g:Profiler gene ontology analysis[47] revealed that three of these modules appear to have considerable relevance to NMP differentiation. Module 3, hereafter referred to as the mesoderm module, was most strongly enriched for the paraxial mesoderm development GO BP term (18.90-fold enrichment, g:SCS-adjusted *p*-value of 1.9×10^{-3}), driven by *Dll3*, *Foxc1*, *Lef1*, *Nup133*, *Smad3*, *Wnt5a*, and *Zic3*. *Tbx6*, whose expression has been reported to be required for a cell to enter the mesodermal fate[48], appears as the third most highly ranked transcription factor in terms of peak number (15^{th} overall) in the mesoderm module. *Lef1* is also highly ranked in both lists, as the second highest ranked transcription factor for peak count (12^{th} overall) and appearing four times in the top ten highest prominence mesoderm pairs, is also heavily implicated in mesodermal fate determination (Fig. 2D,E ; Fig. 3B). Loss of *Lef1* has been shown to cause defects in paraxial mesoderm differentiation and leads to the formation of surplus neural tubes[49]. *Foxc1* appears as the most highly ranked transcription factor in terms of peak number and five times within the top ten largest transcription factor-transcription factor prominences. *Foxc1* is a regulator of mesoderm commitment to either an intermediate or paraxial fate[50], representing a downstream process from *Lef1* and *Tbx6*. Taken together, this module may represent candidate regulators of mesoderm across two separate fate decisions.

Module 4, hereafter referred to as the pre-neural module, was most strongly enriched for anterior/posterior pattern specification (fold = 7.41, $p = 0.049$), driven by posterior Hox genes (*Hoxb6, Hoxb8, Hoxb9, Hoxc4)* and several others (*Gli2*, *Mllt3*, *Msx1*, *Prickle1*, *Prkdc*, *Wls*), alongside a more modest enrichment for neuron development (fold = 3.28, $p = 3 \times 10^{-3}$). Enriched neural terms were restricted to broad progenitor categories (e.g., nervous system

development, neurogenesis, neuron differentiation) while, to the best of our knowledge, terms marking terminal neuronal differentiation were not significantly enriched. This is consistent with reports that cells commit to a regional identity before acquiring neural identity[51] and aligns with our sample, which precedes neural commitment and lacks any final state cells. The module may therefore comprise candidate regulators of the spatial fate that cells acquire prior to neural differentiation.



B

Timepoint	Cell count	# Sig. Correlations
D3	3,019	6,898,278
D3.2	2,330	3,987,293
D3.4	2,123	3,577,837
D3.6	2,445	3,157,792
D3.8	2,029	2,432,397
D4	1,628	3,093,746

C

Module	Size	Con.	TPR
Fate Decision	419	0.21	0.32
Mesoderm	313	0.06	0.67
Preneural	146	0.27	0.31

Figure 2. Gene-gene correlation peaks identify separate modules of transition drivers in NMP differentiation.
A. Fraction of cells with the indicated state label at each timepoint. B. Descriptive statistics describing the dataset and output from BigSur correlation analysis. C. Module-level descriptive statistics (Con., conductance; TPR, triangle participation ratio). D. Top 20 genes with the most peaks in modules 1-3 (red, transcription factors). E. Top 20 transcription factors with the most peaks in modules 1-3.

Of the three, Module 2 had the weakest enrichments for any of the GO BP terms, being most strongly enriched for regulation of cellular response to growth factor stimulus (fold = 3.64, $p = 0.03$). Despite this, we hypothesize that this module captures the most upstream candidate drivers: those that directly regulate the decision between a neural and mesodermal fate. This is largely motivated by the presence of two key players in fate determination: *Sox2* and *T*

(Brachyury). These two genes are co-expressed in NMPs, but work antagonistically with one another to control the choice of either a neural or mesodermal fate[52]. *T* directly controls the expression of *Tbx6*[52]. Once expressed, *Tbx6* suppresses *Sox2* to inhibit neural development and reinforce a mesodermal fate[48,52]. Additionally, *Cdx1* and *Cdx2* are both present in module 2, genes which are known to directly regulate the Hox genes in both the mesoderm and neurectoderm[53,54]. This provides further evidence that module 2, which will now be referred to as the fate decision module, is representative of processes taking place upstream of those in the mesoderm and pre-neural modules.

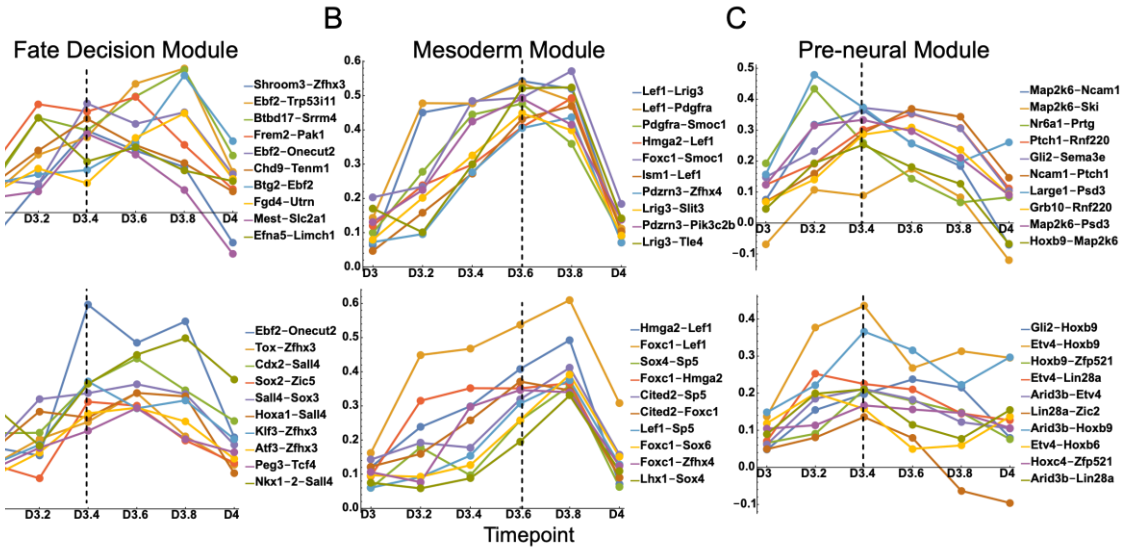


Figure 3. Most prominent gene-gene correlation trajectories in modules.
The top ten highest prominence overall gene-gene (top) and transcription factor pairs (bottom) are shown for each module (A, Fate Decision; B, Mesoderm; C, Pre-neural). Median peak timepoint for gene pairs in each module is indicated with a dashed line.

Module assignments in the NMP dataset were substantially better separated than those in the EMT dataset, with the mesoderm module reaching a conductance of 0.06 (Fig. 2C). Few edges cross its boundary, indicating that the candidate mesodermal driver genes form a largely self-contained set. This is expected if the two fates are driven by distinct regulatory programs. The fate decision and pre-neural modules had higher conductance, consistent with greater

sharing of genes between them. This is unsurprising as the fate decision modules contains state-transition drivers for the early progenitors and the pre-neural cells have not yet terminally differentiated.

The advantage of analyzing more than three timepoints is that the time at which each correlation peaks carries dynamic information. The median peak timepoint was later for mesoderm module genes than for the pre-neural or fate decision modules (D3.6 vs. D3.4; Fig. 3), consistent with the earlier appearance of pre-neural cells in this dataset.

Correlation peak analysis therefore recovered several features of NMP differentiation. It identified three modules of candidate driver genes, each containing known regulators of the process. Network metrics indicated that the neural and mesodermal programs are largely separable. Finally, peak timing was consistent with the two fates being specified at different points in the window.

5.2.3 Genes identified through correlation peaks are not easily identified through differential gene expression analysis

Differential gene expression analysis is commonly used to observe differences between groups of cells (e.g., between cell types or states). Correlation is a function of co-expression, so it is natural to think that the genes observed with correlation peaks may simply be more highly expressed at the timepoint closest to the tipping point. To test this, we identified differentially expressed genes (DEGs) in both the EMT and NMP datasets. These were calculated by performing a one-versus-all Mann-Whitney test on the putative critical transition timepoint against all other timepoints. For the NMP datasets, the median peak time per module was used as the critical transition timepoint.

Module	Gene	Fold change (Log2)	Adjusted *p*-value	Enrichment rank (*, highest rank)
EMT 1	*ZEB2*	0.78	1×10^{-37}	157 *
EMT 2	*HMGB3*	0.98	6×10^{-116}	461 *
EMT 3	*TSHZ2*	2.63	2×10^{-133}	80 *
Mesoderm	*Hes7*	0.96	4×10^{-60}	42 *
Mesoderm	*Tbx6*	0.91	4×10^{-15}	46
Pre-neural	*5730508A11Rik*	0.89	2×10^{-107}	37 *
Pre-neural	*Prickle2*	0.71	8×10^{-40}	128
Fate decision	*Mt1*	1.10	8×10^{-115}	14 *
Fate decision	*T*	-1.33	3×10^{-13}	-

Table 1. Mann-Whitney test results for a set of example genes from each correlation peak module. p-values were adjusted using Benjamini-Hochberg. An asterisk in the Enrichment rank column indicates that that gene had the highest fold change in expression of any gene in their respective module.

Many genes identified by correlation peak analysis were significantly differentially expressed (Supplementary Fig. 1,2; Supplementary Table; Table 1), but their fold changes were low relative to other genes. Apart from EMT module 3 and the NMP fate decision module, the most strongly differentially expressed member of each module showed less than a two-fold difference in expression at its putative transition timepoint. Many genes belonging to a correlation peak module actually have negative fold changes (Figure 4), including the NMP fate regulator *T* (Table 1). This is particularly true for the mesoderm module where, of the 222 genes found to have significant differential expression at D3.6, 162 have a negative fold change in expression. Correlation peak detection therefore identifies genes that would rank low in a conventional differential expression analysis, despite their coordinated behavior at the transition.

Notably, these genes would also be difficult to pick out in feature plots, where, oftentimes, the different timepoints do not clearly separate in UMAP space (Supp. Fig. 2,3).

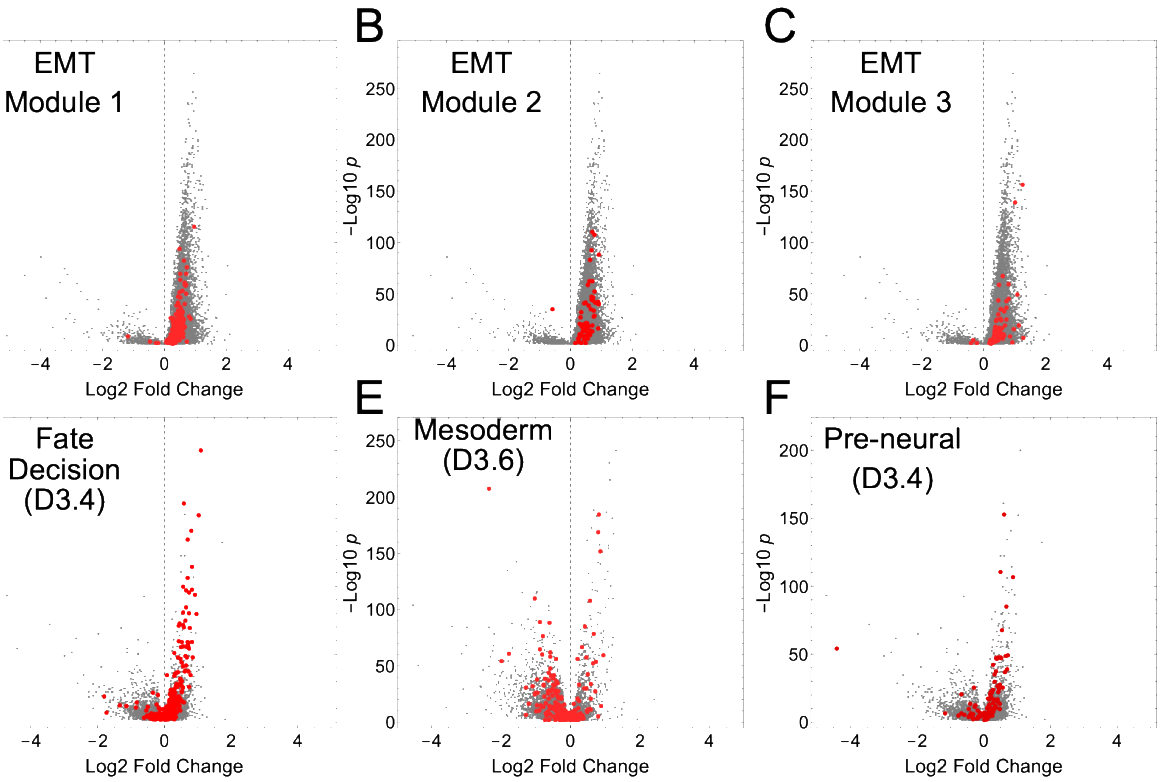


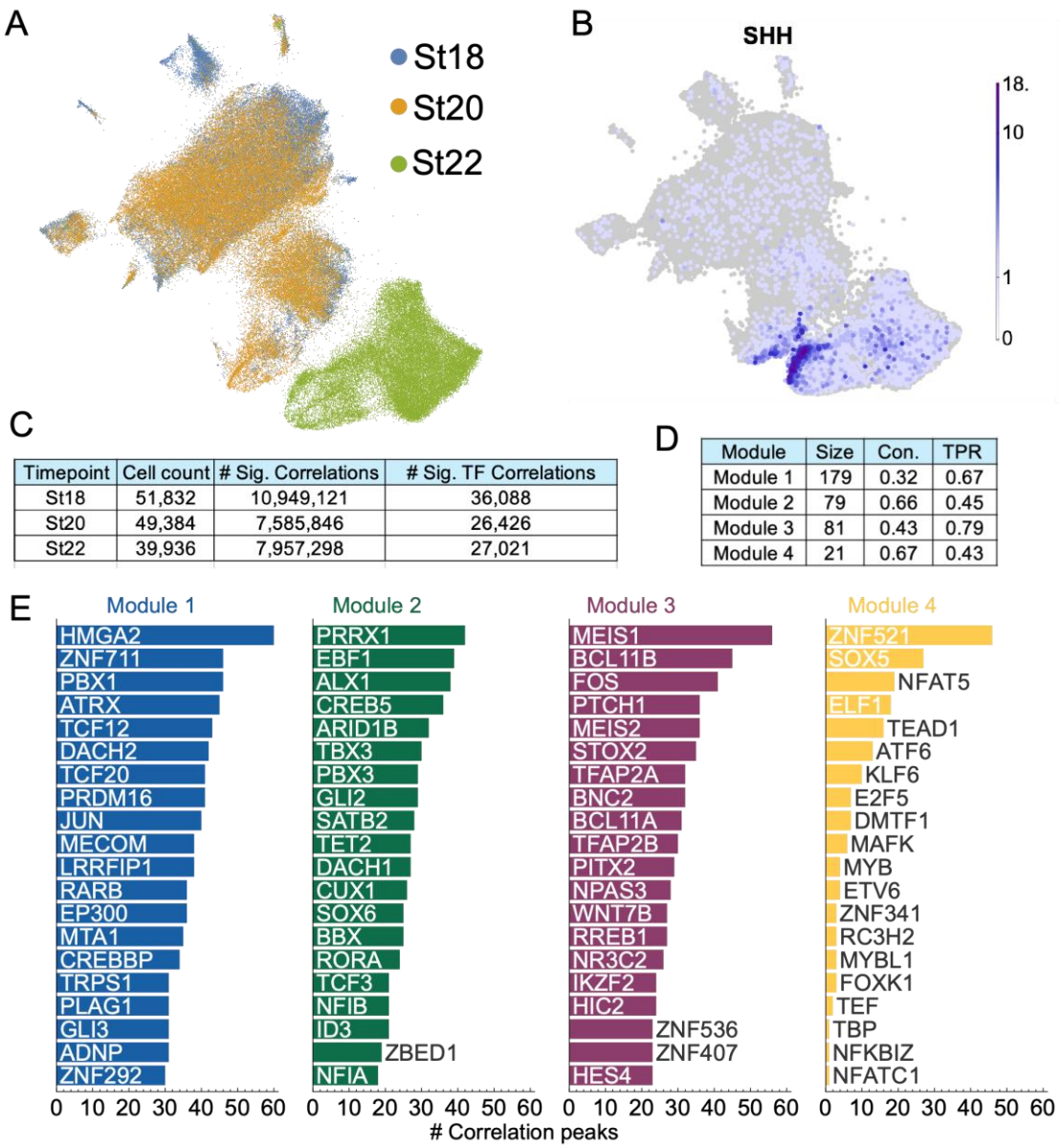
Figure 4. Volcano plots of significant DEGs in each module
(A, EMT Module 1; B, EMT Module 2; C, EMT Module 3; D, NMP Fate decision; E, NMP mesoderm; F, NMP pre-neural). Red indicates that the gene is a member of the module.

5.2.4 PITX2 and NPAS3 drive SHH-expression in the FEZ

We next applied this process to craniofacial development in *Gallus gallus*, where specific transitions are less well characterized. During craniofacial development, a small region called the frontonasal ectodermal zone (FEZ), at a boundary with *FGF8* expressing epithelia, controls patterning and regulates growth of the frontonasal process in *Gallus gallus* through *SHH* signaling[55]. The drivers which lead these cells to a *SHH*-expressing state are not well understood.

To see if we could identify drivers with correlation peaks, we first performed scRNAseq on isolated FEZ epithelial cells at three different stages of development (Hamburger-Hamilton Stages (St) 18, 20, and 22). After removing a small subset of cells expressing neuron-related genes (e.g., *NEUROG1, MYT1L*) from each sample, we plotted the cells in UMAP space. St 18 and St 20 cells occupy overlapping regions of UMAP space with a majority of St 22 cells (Fig. 5A). A relatively small number of cells express *SHH* in each of the three timepoints (Fig. 5B, Supp. Fig 4). Despite this, because we do not know what the ground truth is regarding cell states, we chose not to remove any additional cells.

We identified candidate drivers by identifying correlation peaks in a similar manner as before. First, we performed BigSur correlation analysis on each of the three stages. We elected to determine the correlation trajectory only between pairs of genes belonging to the subset including the *Gallus gallus* transcription factors, *SHH, PTCH1, PTCH2,* and *WNT7B*. From this, 2,425 peaks (360 unique genes) were collected without thresholding for prominence. Four modules were identified from these peaks (Figure 5E). Notably, the genes in these modules are generally more lowly expressed at St20 than the other two stages (Supp. Fig. 5).



Timepoint	Cell count	# Sig. Correlations	# Sig. TF Correlations
St18	51,832	10,949,121	36,088
St20	49,384	7,585,846	26,426
St22	39,936	7,957,298	27,021

Module	Size	Con.	TPR
Module 1	179	0.32	0.67
Module 2	79	0.66	0.45
Module 3	81	0.43	0.79
Module 4	21	0.67	0.43

Figure 5. Correlation trajectory analysis reveals four modules of genes sharing correlation peaks among the subset of transcription factors, SHH, PTCH1, PTCH2, and WNT7B.
A. Plot of cells in UMAP space colored by timepoint. B. Feature plot of SHH expression in UMAP space. Color indicates expression level. C. Overview of Descriptive statistics describing the dataset and output from BigSur correlation analysis. “TF correlations” encompasses the subset of genes indicated in the text. D. Module-level descriptive statistics (Con., conductance; TPR, triangle participation ratio). E. The four modules constructed from correlation peak analysis. The 20 genes with the largest number of correlation peaks in each module are shown.

The conductance values for each of these modules were high relative to either the EMT or NMP datasets, indicating that the module assignments were loose. Despite this, module 3 was

of particular interest as it contained both *SHH* and its receptors *PTCH1* and *PTCH2* (Fig. 6A) with many of the genes being tightly positively correlated with one another (Fig. 6C). *WNT7B* also appeared in this module, which is known to work in tandem with *SHH* to regulate ectodermal boundaries in mammalian tooth development[56]. Though this list contained many interesting candidate drivers, we decided to first look more deeply at *PITX2* or *NPAS3*. Each of these genes ranked highly in number of correlation peaks for module 3 (11th and 12th respectively) (Fig. 6A) and appeared multiple times in the list of top ten highest prominence pairs (Fig. 6B). They each exhibited peaks in correlation with the *SHH* receptor *PTCH1*, but increasing correlation with both *SHH* and each other (Fig 6D).

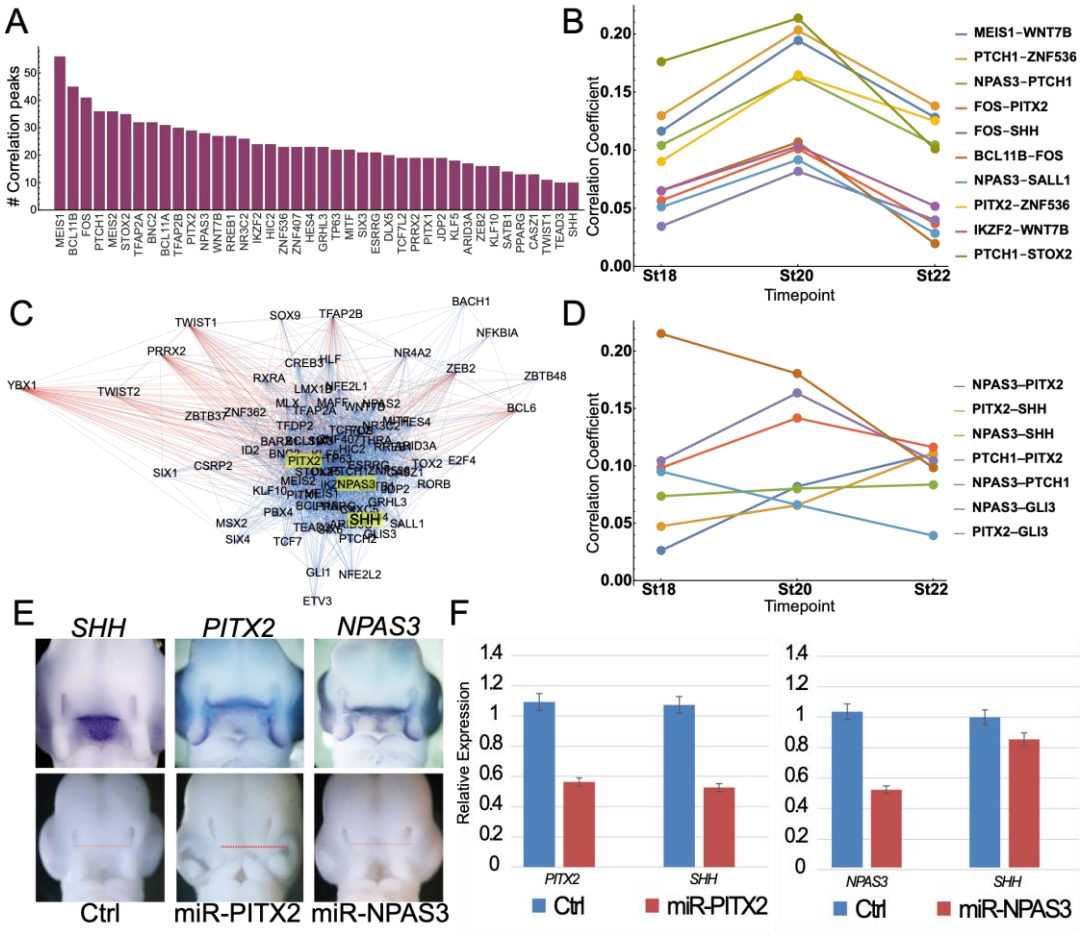


Figure 6. Correlation peak analysis reveals SHH expression drivers PITX2 and NPAS3.

A. Top 40 genes with the most correlation peaks in module 3. B. Top ten highest prominence gene-gene pairs in module 3. C. Network plot of correlations between members of module 3. (Blue edges, positive correlation; red edges, negative). SHH, NPAS3, and PITX2 are highlighted in yellow. D. Selected trajectories between pairs containing PITX2 or NPAS3. E. St22 Gallus gallus. Top: FISH imaging of SHH, PITX2, and NPAS3 in the FEZ. Bottom: Facial structure after knockout experiments. A red line is drawn between the two nasal pits. F. Relative expression of PITX2, SHH, and NPAS3 between experimental conditions.

FISH imaging confirmed that *PITX2, NPAS3,* and SHH were co-expressed in the FEZ at St 22 (Fig. 6E, top). To test whether *PITX2* or *NPAS3* influenced *SHH* expression, knockout experiments using miR-PITX2 and miR-NPAS3 were performed. Loss of either of these molecules, particularly in the case of *PITX2*, led to deformities in the developing face (Fig. 6E) and significant loss in *SHH* expression (Fig. 6F). Together these results indicate that both *PITX2* and *NPAS3* are required for proper facial development and *SHH* expression in *Gallus gallus*.

5.3 Discussion

The above results suggest that novel drivers of cell state transitions can be identified through a dynamic correlation analysis spanning multiple timepoints. Peaks in correlation appear to identify relevant driver genes in EMT and NMP, and novel drivers in the developing FEZ. Many of these genes that appear highly ranked for either number of peaks or in prominence would be difficult to pinpoint through differential gene expression analysis alone. Still, it is important to recognize this as a proof of concept; this method can, and should, be iterated on further.

From a theoretical perspective, this method observes only one early warning signal of a critical transition: an increase in gene-gene correlations. It does not incorporate the second early warning signal, increased cell heterogeneity. Methods for identifying cells in a transition state commonly combine both, but we chose to omit the heterogeneity term because it cannot be interpreted unambiguously in snapshot data. This is because heterogeneity has multiple sources

that are hard to distinguish in scRNAseq data without ground truth. Suppose at timepoint *t* we observe both a rise in correlation among a subset of genes and a rise in heterogeneity across cells. Tipping point theory predicts that cells approaching the transition will become more variable, but high variability could also result from a mixture effect. A sample near the transition could contain cells at different points along the trajectory, and a greater proportion of the post-transition cells at *t* than at *t-1* would inflate the apparent heterogeneity of the transition cell population. Without a means of separating within-state variability from between-state mixing, the heterogeneity signal cannot solely be attributed to critical slowing down.

However, this problem isn't completely solved by only looking at gene-gene correlations. In fact, heterogeneity of the system has direct effects on the gene-gene correlations as well. A mixed population of cells will have strong negative correlations between genes that are markers of cell state, however they will also show increased positive correlation among their own state's markers. If, over the time course, the population becomes more heterogeneous but becomes less heterogeneous by the end, you are likely to see correlation peaks among cell-state defining genes. In this case, we cannot expect genes which peak in this way to have fewer peaks than the drivers.

Pseudotime or RNA velocity trajectory inference methods could help to solve this problem, but they come at the cost of having to already know which genes are changing across the transition to be accurate. Using a set of highly variable features would be subject to the same problems we assert above: heterogeneity caused by many present cell states would add noise to the highly variable features and could lead to incoherent trajectories being drawn. Additionally, the number of cells at each point in the trajectory is not likely to be equal and is expected to be

particularly low among transition cells. Still, if some ground truth is known, this may be a powerful solution to the problems caused by heterogeneity.

The prominence threshold that was applied may have a large effect on which genes are ranked highly for number of peaks or for total prominence, particularly if there are genes with many small peaks in correlation. It is not clear whether these should or should not be included and, if included, how normalization can be implemented. Relying on the highest prominence pairs of genes avoids this problem, at the expense of being more difficult to analyze for individual genes.

5.4 Conclusions

Critical transition theory predicts that a set of driver genes will increase in correlation as they approach a tipping point. Here, we have provided evidence that these peaks in correlation can be leveraged to identify cell state transition driver genes using time-series scRNAseq data that differential gene expression analysis would miss and have used this methodology to identify novel driver genes in craniofacial development.

5.5 Methods

5.5.1 Calculating Correlation Trajectories

A "correlation trajectory" is the sequence of correlation coefficient values that a gene-gene pairs holds over the course of a time series. Modified-corrected correlation coefficients and their respective *p*-values were calculated using BigSur[34]. A 0.02 Benjamini-Hochberg corrected *p*-value was used to determine statistical significance.

Because a correlation coefficient of exactly 0 cannot be determined to stem from a lack of statistical power or true lack of correlation, a gene-gene pair which had a correlation coefficient of 0 at any of the time points was removed from analysis.

For each gene pair, the peak timepoint was taken as the timepoint of maximum correlation. Prominence was then defined as the correlation coefficient at that timepoint minus the larger of the correlation coefficients at the first and last timepoints. Gene pairs which had prominence values below a specified value were then filtered out.

5.5.2 Creating Correlation Peak Modules

From the list of correlation peak pairs, a gene x gene adjacency matrix was created with entries valued at one if the genes shared a correlation peak and a zero if they did not. The walktrap algorithm[37] with a step parameter of 4 was used to define gene modules.

Walktrap frequently returns very small modules, in some cases ones which contain only a single gene. These were merged into larger modules after initial community detection. At each step, the smallest module under twenty genes was identified, and the total number of edges between it and every other module was computed from the adjacency matrix. That module was then merged into the one with which it shared the greatest number of edges. Modules with no edges to any other module were left unmerged. This procedure repeated until every module either exceeded the size threshold or had no connections to merge across.

5.5.3 Calculation of Module Measurements

Modules were evaluated using two separate measurements: conductance and triangle participation ratio (TPR)[39,57,58]. For a given module, conductance measures the fraction of total edge volume that points outside of the cluster. To calculate it, you sum over all edge endpoints belonging to module members and calculate the proportion which point to genes outside of the module. Values range from 0 to 1, with lower values indicating that the genes within a module are mainly connected to one another rather than to others. It is commonly defined in the form:

$$\phi(S) = \frac{e(S,\ S^-)}{\min(vol(S), vol(S^-)}$$

Where S is the module being evaluated, S^- is the remainder of the graph, $e(S,\ S^-)$ calculates the cut (i.e., the total weight of edges with one endpoint in S and the other in S^-), and vol(S) and vol(S^-) are the volumes of S (i.e., the sum of degrees of its members, counting internal edges twice as both endpoints are inside) and S^- respectively.

TPR measures the fraction of nodes in a module that belong to a triangle (i.e., node A connects to node B, node B connects to node C, and node C connects to node A) within the module. It is defined as:

$$TPR(S) = \frac{\#\{u: u \in S, \{(v,w): v,w \in S, (u,v) \in E, (u,w) \in E, (v,w) \in E\} \neq \emptyset\}}{n_s}$$

Where S is the module, n_s is the number of nodes in S, u is the node being checked, v and w are any other two nodes in S, and E is the edge set of the graph. A node counts in the numerator if at least one such pair (v,w) that participates in a triangle with u exists.

5.5.4 Analysis of EMT data

Raw sequencing MCF10A reads from three separate timepoints across TGF-β-induced EMT (before treatment, 4 days after treatment, and 8 days after treatment)[35] were obtained from the NCBI Sequence Read Archive (BioProject PRJNA698642; SRA accessions SRR13606306, SRR13606302, SRR13606301) using the SRA Toolkit, and aligned to GRCh38 using the 10x Genomics prebuilt reference refdata-gex-GRCh38-2020-A with CellRanger v10.0.0[59] (default parameters). Genes with mean expression under 8 x 10^{-3} counts per cell and cells with under 10,000 total UMI were filtered before analysis. BigSur correlation analysis and correlation peak analysis (prominence threshold = 0.05) was then performed as written above.

Module enrichment for EMT genes was performed by comparing the fraction of genes in the module present in the dbEMT 2.0[38] database to the number expected to be in a module of that size. The number expected is calculated with:

$$E_k = \frac{n\,K}{m}$$

Where E_k is the expected number of annotated EMT genes in the module, n is the size of the module, m is the number of transcription factors with a significant correlation at any timepoint, and K is the total number of EMT genes in m. *p*-values were determine through a hypergeometric test.

5.5.5 Analysis of NMP data

Processed counts data and cell-type annotations of ESC-derived NMPs at 6 timepoints between 3 and 4 days after treatment were obtained directly from the authors of Fontaine *et al* 2025[46]. Cells with labels other than "NMP", "EarlyMeso", "Meso", or "PreNeural" were filtered out. Genes were filtered out if they had fewer than 40 total UMI counts in any of the 6 datasets. Gene correlation analysis, correlation trajectory calculation, and module inference were performed as previously written.

Gene ontology enrichment was performed with g:Profiler[47] for *Mus musculus*, restricted to the GO biological process database. The statistical domain was set to the default of all known genes (15,161). Significance was assessed at a threshold of 0.05 with g:SCS multiple-testing correction.

5.5.6 Analysis of FEZ data

scRNAseq Preparation

Embryos were incubated to the appropriate developmental stages for gene single cell RNA sequencing (HH18, HH20, and HH22[60]). Embryos were removed from the eggs and the

heads were removed. The frontonasal prominences (FNP) were collected, then incubated in 3mg/ml dispase in DMEM for 20 minutes on ice. Following incubation, the FNP was washed with DMEM and the facial ectoderm containing the frontonasal ectodermal zone (FEZ) was carefully dissected away from the underlying mesenchyme using sharpened tungsten needles. The isolated facial ectoderm was then digested with 1 mg/mL Liberase in DMEM for 10 min at 37°C to generate a single-cell suspension. Nuclei were subsequently isolated according to the 10x Genomics nuclei isolation protocol (CG000366). The isolated nuclei were transported to the UCSF Genomics Core for 10x Genomics Multiome sequencing.

qPCR

The surface ectoderm containing the FEZ was isolated from treated and control embryos (miRNA, see "Gene knock-down"). Total RNA was prepared using the Phenol-Chloroform / TRIzol Extraction mothod, and qPCR was used to determine the relative quantity of each transcript using the SYBR Green method.

In situ hybridization

Embryos were incubated to the correct stage for gene expression analysis, removed from eggs, rinsed in ice-cold PBS, fixed in 4% paraformaldehyde over-night at 4°C, and taken through a graded ethanol series for dehydration. These samples were prepared for whole mount *in situ* hybridization.

Probes were prepared by cloning unique regions (see below) of *NPAS3* and *PITX2* into pBluescript SK vector. Sequencing was used to confirm the sequence cloned and orientation. A SHH probe was already available and was also used. Following plasmid linearization, digoxigenin (DIG)-labeled cRNA probes were synthesized and purified. In situ hybridization was performed on whole embryos at HH18, HH20, HH22, and HH25, as well as on embryos

following experimental manipulation. In situ hybridization was performed on whole embryos as described [61].

Sequences of probes

SHH:

forward 5'-GCTGACAGACTGATGACTCA-3'

reverse 5'-TCGTAGTGCAGCGATTCCTC-3'

PITX2:

forward 5'-GCCGGGAATAATAAGCTGTG-3'

reverse 5'-TCTCTTTCTCTGCGGCCT-3'

NPAS3:

forward 5'-ATGATGGGAACAGCTCCAG-3'

reverse 5'-GCTCCACCTTGATCTGCA-3'

Preparation of experimental embryos

Fertilized chicken eggs (*Gallus gallus*) were incubated until embryos reached Hamburger and Hamilton stage 10 [HH 10 (approximately 36 h)] and then a small hole was made in the shell directly over the embryo and 1.0 ml of albumin was removed. Tape was placed over the hole and eggs were returned to the incubator until the appropriate stage for experimental manipulation (HH14/15).

Genetic knock-down

To knock-down PITX2 and NPAS3, microRNAs targeting each transcript were generated and cloned into the RCAS-Viral expression vector. A plasmid DNA solution (concentration 1mg/ml) containing the respective RCAS-miRNA construct was prepared, and 0.3-0.5 μL was injected adjacent to the frontonasal process (FNP) ectoderm. Electroporation was performed at

HH14–HH15 using five pulses of 50 ms each at approximately 10 V, with 100 ms intervals between pulses, to transfect the ectodermal cells comprising the FEZ. Following electroporation, eggs were returned to the incubator and embryos were collected at HH22 and prepared for microscopy and either in situ hybridization, or qPCR.

Sequences of miRNAs

PITX2:

Top: 5'-

TGCTGTTGGCTTTGAGTCTCAGGCTGGTTTTGGCCACTGACTGACCAGCCTGACTCA

AAGCCAA-3'

Bottom: 5'-

CCTGTTGGCTTTGAGTCAGGCTGGTCAGTCAGTGGCCAAAACCAGCCTGAGACTCAA

AGCCAAC-3'

NPAS3:

Top: 5'-

TGCTGAAATGGTCCCGACATATTGAGGTTTTGGCCACTGACTGACCTCAATATCGGG

ACCATTT-3'

Bottom: 5'-

CCTGAAATGGTCCCGATATTGAGGTCAGTCAGTGGCCAAAACCTCAATATGTCGGGA

CCATTTC-3'

scRNAseq Data Analysis

Raw sequencing reads were aligned to the chicken genome (galGal6, ENSEMBL) using 10X Genomic's Cell Ranger ARC v2.0.2 with default settings. Doublets were detected in both

the scRNAseq and scATAC-seq data using scDblFinder v1.24.10 [62]. Cells which were labeled as doublets in both scRNAseq and scATAC-seq data were removed from analysis.

From there, genes with a mean expression under 1 x 10^{-3} counts per cell in any of the timepoints and cell with under 1,300 total UMI were removed before analysis.

For each of the three datasets, features were selected using BigSur (modified corrected Fano factor >1.0, Benjamini Hochberg $p<0.05$). The top 20 principal components of those selected features (calculated on the modified corrected Pearson residuals matrix) were used to construct a shared nearest neighbor graph, which was then partitioned by Leiden clustering (resolution = 0.5). Cells belonging to clusters expressing relatively high amounts of *NEUROG1*, *MYT1L*, *LHX2*, and *OLIG2* were filtered from further analyses. Gene correlation analysis, correlation trajectory calculation, and module inference were performed as previously written.

5.5.7 UMAP Representations of Timepoints

EMT and NMP datasets

Counts matrices for all relevant timepoints were concatenated. From there, BigSur was used to select features and create the modified corrected Pearson residuals matrix (EMT: modified corrected Fano > 1.0, Benjamini Hochberg $p<0.05$; NMP: modified corrected Fano > 1.3, Benjamini Hochberg $p<0.05$). From the modified corrected Pearson residuals matrix, the top 20 principal components of selected features were used to construct a shared nearest neighbor graph, which was then projected into two-dimensional UMAP space.

Gallus gallus dataset

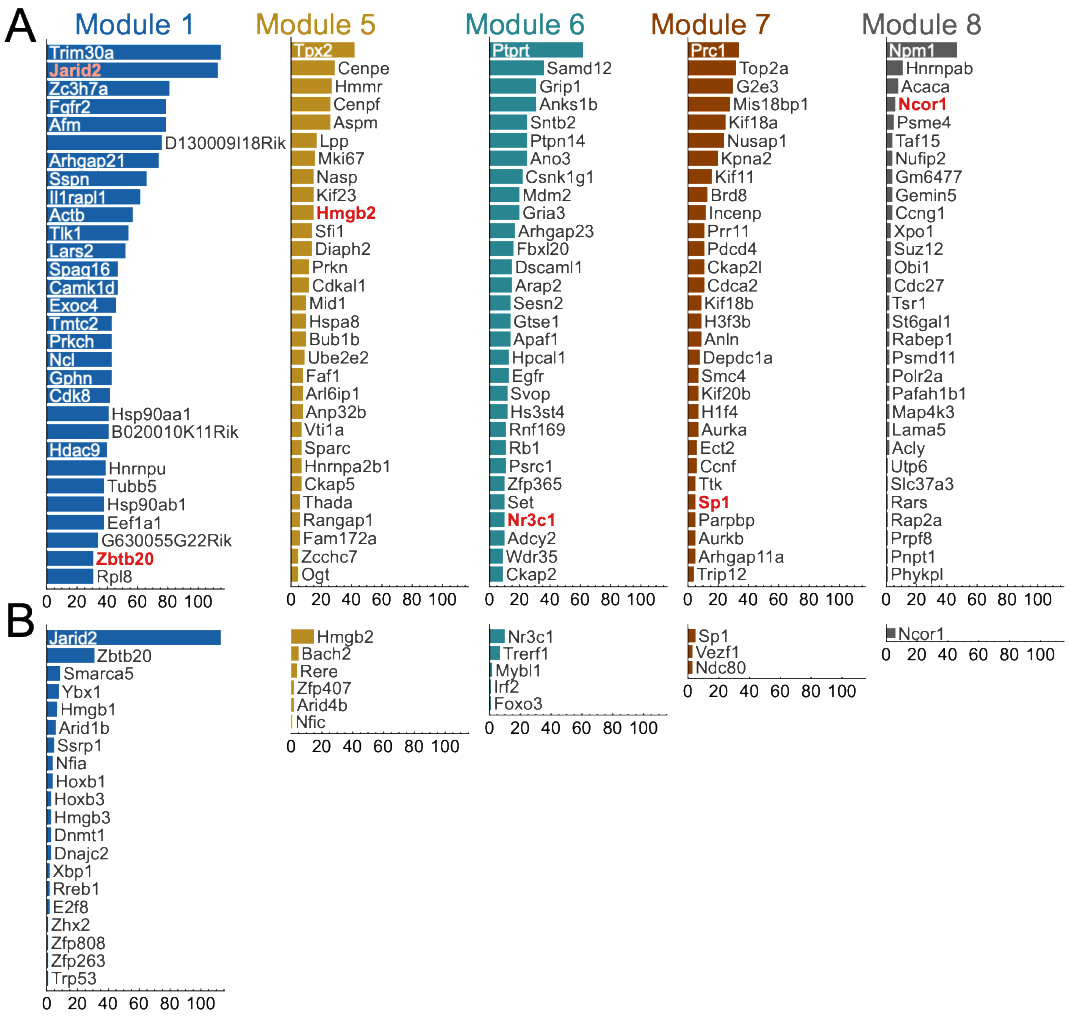
BigSur was used to select features from only the St22 dataset (modified corrected Fano > 1.0, Benjamini Hochberg $p<0.05$). Counts matrices for the three timepoints were then concatenated and the modified corrected Pearson residuals matrix was calculated. From the

modified corrected Pearson residuals matrix, the top 20 principal components of the St22 selected features were used to construct a shared nearest neighbor graph, which was then projected into two-dimensional UMAP space.

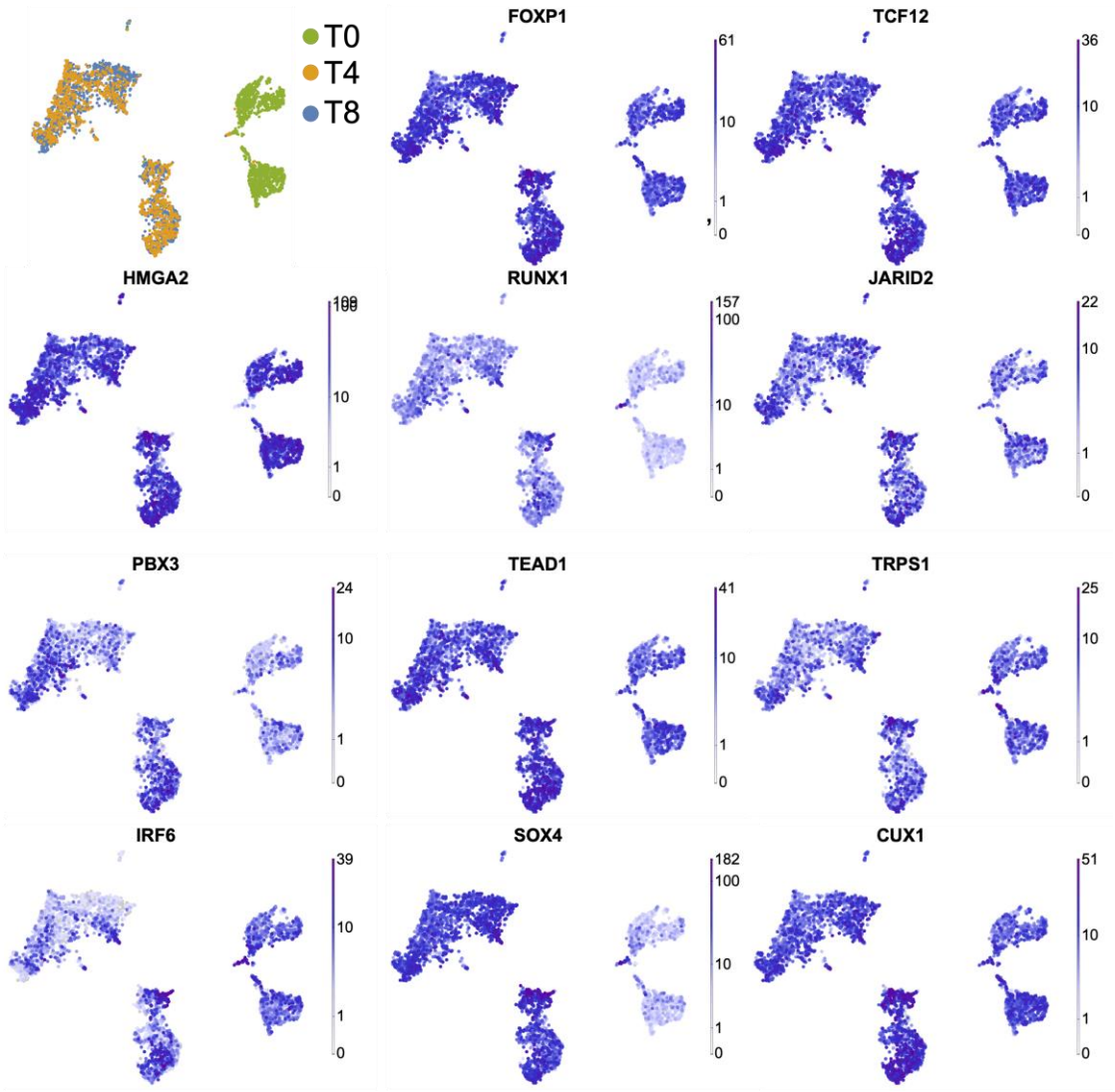
5.5.8 Differential gene expression

Significant DEGs were determined by the Mann-Whitney test as implemented in Mathematica with a threshold of a Benjamini Hochberg corrected p-value = 0.05. The putative tipping point timepoint was tested against the concatenated matrices of all other timepoints.

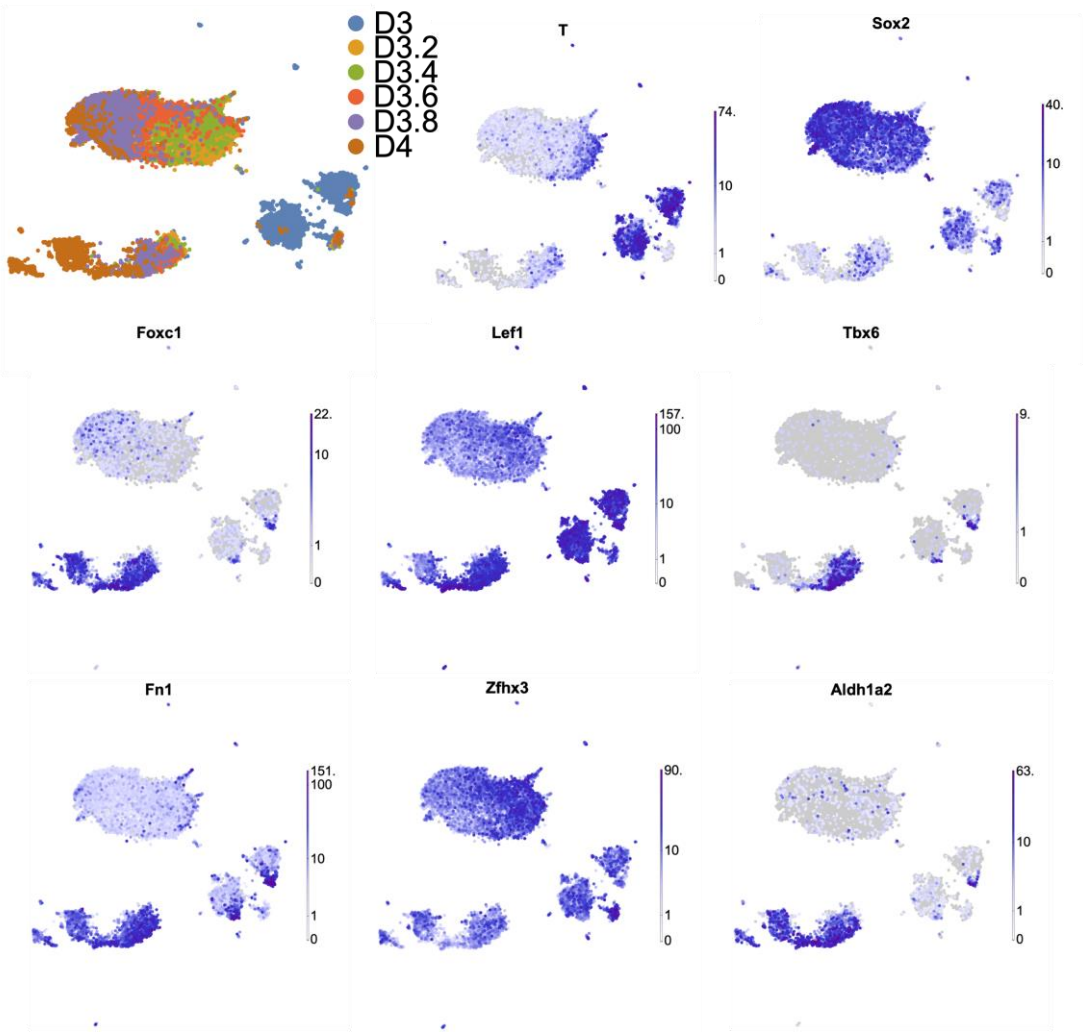
5.6 Supplementary Figures



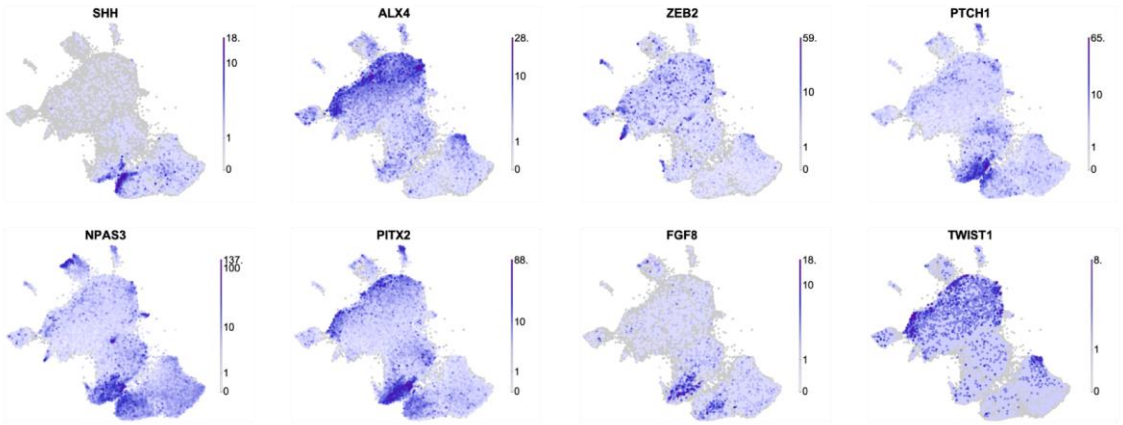
Supplementary Figure 1. Correlation peak modules 1, 5-7 from the NMP dataset analysis. Bar length indicates the number of correlation peaks a gene participates in. A. The 30 genes with the largest number of peaks per module. B. Transcription factors belonging to each module.



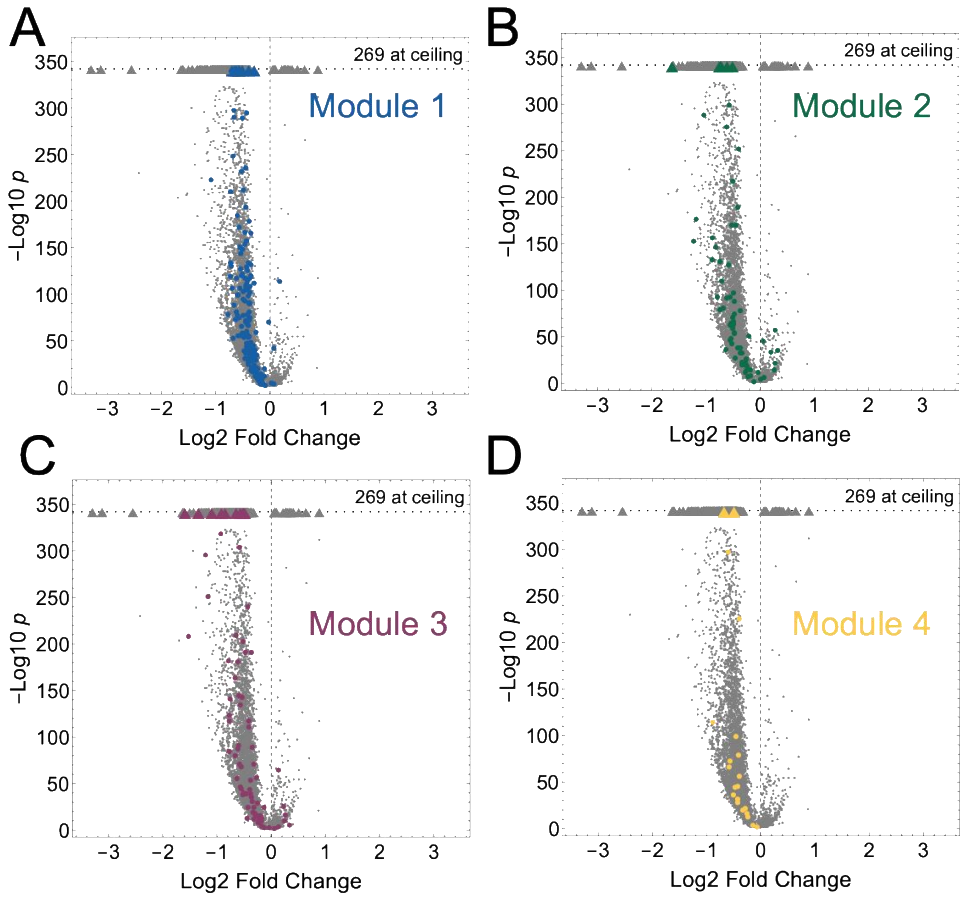
Supplementary Figure 2. Feature plots of EMT dataset in UMAP space. The top left shows the relative positions of the cells at the three different timepoints. Color indicates expression level, going from zero (gray) to the maximum number of counts (indigo).



Supplementary Figure 3. Feature plots of NMP dataset in UMAP space. The top left shows the relative positions of the cells at the six different timepoints. Color indicates expression level, going from zero (gray) to the maximum number of counts (indigo).



Supplementary Figure 4. Feature plots of *Gallus gallus* dataset in UMAP space. Color indicates expression level, going from zero (gray) to the maximum number of counts (indigo).



Supplementary Figure 5. Volcano plots showing the statistically significant DEGs for each module from St20 *Gallus gallus* FEZ. DEGs were considered significant if they met a Benjamini-Hochberg corrected p-value of 0.05. Colored points indicate a gene belonging to the module (A: Module 1, blue; B: Module 2,

green; C: Module 3, purple; D: Module 4, yellow). Genes which were assigned a p-value less than 2 x 10-323 are marked with a triangle and placed at 342.6 on the y-axis (dotted line). The number of genes that met that criteria are shown in the top right of the plots.

5.7 References

1. Jacob F, Monod J. Genetic regulatory mechanisms in the synthesis of proteins. J Mol Biol. 1961;3:318–56. https://doi.org/10.1016/S0022-2836(61)80072-7

2. Kauffman SA. The Origins of Order: Self-Organization and Selection in Evolution. Oxford University Press; 2023. https://doi.org/10.1093/oso/9780195079517.001.0001

3. Kauffman S. Homeostasis and Differentiation in Random Genetic Control Networks. Nature. 1969;224:177–8. https://doi.org/10.1038/224177a0

4. McAdams HH, Arkin A. Stochastic mechanisms in gene expression. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 1997;94:814–9. https://doi.org/10.1073/pnas.94.3.814

5. Swain PS, Elowitz MB, Siggia ED. Intrinsic and extrinsic contributions to stochasticity in gene expression. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 2002;99:12795–800. https://doi.org/10.1073/pnas.162041399

6. Bahar Halpern K, Tanami S, Landen S, Chapal M, Szlak L, Hutzler A, et al. Bursty Gene Expression in the Intact Mammalian Liver. Mol Cell. 2015;58:147–56. https://doi.org/https://doi.org/10.1016/j.molcel.2015.01.027

7. Beal J. Biochemical complexity drives log-normal variation in genetic expression. Eng Biol. John Wiley & Sons, Ltd; 2017;1:55–60. https://doi.org/https://doi.org/10.1049/enb.2017.0004

8. Till JE, McCulloch EA, Siminovitch L. A stochastic model of stem cell proliferation, based on the growth of spleen colony-forming cells*. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 1964;51:29–36. https://doi.org/10.1073/pnas.51.1.29

9. Suda T, Suda J, Ogawa M. Disparate differentiation in mouse hemopoietic colonies derived from paired progenitors. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 1984;81:2520–4. https://doi.org/10.1073/pnas.81.8.2520

10. Chang HH, Hemberg M, Barahona M, Ingber DE, Huang S. Transcriptome-wide noise controls lineage choice in mammalian progenitor cells. Nature. 2008;453:544–7. https://doi.org/10.1038/nature06965

11. Piras V, Tomita M, Selvarajoo K. Transcriptome-wide Variability in Single Embryonic Development Cells. Sci Rep. 2014;4:7137. https://doi.org/10.1038/srep07137

12. Pina C, Fugazza C, Tipping AJ, Brown J, Soneji S, Teles J, et al. Inferring rules of lineage commitment in haematopoiesis. Nat Cell Biol. 2012;14:287–94. https://doi.org/10.1038/ncb2442

13. Krantz SB. Erythropoietin. Blood. 1991;77:419–34. https://doi.org/10.1182/blood.V77.3.419.419

14. Yamanaka S. Induction of pluripotent stem cells from mouse fibroblasts by four transcription factors. Cell Prolif. 2008;41 Suppl 1:51–6. https://doi.org/10.1111/j.1365-2184.2008.00493.x

15. Xu J, Lamouille S, Derynck R. TGF-β-induced epithelial to mesenchymal transition. Cell Res. Nature Publishing Group; 2009;19:156–72. https://doi.org/10.1038/cr.2009.5

16. Mojtahedi M, Skupin A, Zhou J, Castaño IG, Leong-Quong RYY, Chang H, et al. Cell Fate Decision as High-Dimensional Critical State Transition. PLOS Biol. Public Library of Science; 2016;14:e2000640. https://doi.org/10.1371/journal.pbio.2000640

17. Huang S, Guo Y-P, May G, Enver T. Bifurcation dynamics in lineage-commitment in bipotent progenitor cells. Dev Biol. 2007;305:695–713. https://doi.org/10.1016/j.ydbio.2007.02.036

18. Reynolds JF, Smith DMS, Lambin EF, Turner BL, Mortimore M, Batterbury SPJ, et al. Global Desertification: Building a Science for Dryland Development. Science. American Association for the Advancement of Science; 2007;316:847–51. https://doi.org/10.1126/science.1131634

19. Lovejoy TE, Nobre C. Amazon Tipping Point. Sci Adv. American Association for the Advancement of Science; 4:eaat2340. https://doi.org/10.1126/sciadv.aat2340

20. Centola D, Becker J, Brackbill D, Baronchelli A. Experimental evidence for tipping points in social convention. Science. American Association for the Advancement of Science; 2018;360:1116–9. https://doi.org/10.1126/science.aas8827

21. Held H, Kleinen T. Detection of climate system bifurcations by degenerate fingerprinting. Geophys Res Lett. John Wiley & Sons, Ltd; 2004;31. https://doi.org/https://doi.org/10.1029/2004GL020972

22. Winkelmann R, Donges JF, Smith EK, Milkoreit M, Eder C, Heitzig J, et al. Social tipping processes towards climate action: A conceptual framework. Ecol Econ. 2022;192:107242. https://doi.org/https://doi.org/10.1016/j.ecolecon.2021.107242

23. Dakos V, Matthews B, Hendry AP, Levine J, Loeuille N, Norberg J, et al. Ecosystem tipping points in an evolving world. Nat Ecol Evol. 2019;3:355–62. https://doi.org/10.1038/s41559-019-0797-2

24. Dakos V, Scheffer M, van Nes EH, Brovkin V, Petoukhov V, Held H. Slowing down as an early warning signal for abrupt climate change. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 2008;105:14308–12. https://doi.org/10.1073/pnas.0802430105

25. Lenton TM. Early warning of climate tipping points. Nat Clim Change. 2011;1:201–9. https://doi.org/10.1038/nclimate1143

26. Lenton TM, Held H, Kriegler E, Hall JW, Lucht W, Rahmstorf S, et al. Tipping elements in the Earth's climate system. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 2008;105:1786–93. https://doi.org/10.1073/pnas.0705414105

27. Lenton TM, Livina VN, Dakos V, van Nes EH, Scheffer M. Early warning of climate tipping points from critical slowing down: comparing methods to improve robustness. Philos Trans R Soc Math Phys Eng Sci. Royal Society; 2012;370:1185–204. https://doi.org/10.1098/rsta.2011.0304

28. Folke C, Carpenter S, Walker B, Scheffer M, Elmqvist T, Gunderson L, et al. Regime Shifts, Resilience, and Biodiversity in Ecosystem Management. Annu Rev Ecol Evol Syst. 2004;35:557–81. https://doi.org/https://doi.org/10.1146/annurev.ecolsys.35.021103.105711

29. Scheffer M, Carpenter S, Foley JA, Folke C, Walker B. Catastrophic shifts in ecosystems. Nature. 2001;413:591–6. https://doi.org/10.1038/35098000

30. Smug D, Ashwin P, Sornette D. Predicting financial market crashes using ghost singularities. PLoS One. Public Library of Science; 2018;13:e0195265. https://doi.org/10.1371/journal.pone.0195265

31. O'Regan SM, O'Dea EB, Rohani P, Drake JM. Transient indicators of tipping points in infectious diseases. J R Soc Interface. 2020;17:20200094. https://doi.org/doi:10.1098/rsif.2020.0094

32. Heinze C, Blenckner T, Martins H, Rusiecka D, Döscher R, Gehlen M, et al. The quiet crossing of ocean tipping points. Proc Natl Acad Sci. Proceedings of the National Academy of Sciences; 2021;118:e2008478118. https://doi.org/10.1073/pnas.2008478118

33. Scheffer M, Carpenter SR, Lenton TM, Bascompte J, Brock W, Dakos V, et al. Anticipating Critical Transitions. Science. American Association for the Advancement of Science; 2012;338:344–8. https://doi.org/10.1126/science.1225244

34. Silkwood K, Dollinger E, Gervin J, Atwood S, Nie Q, Lander AD. Leveraging gene correlations in single cell transcriptomic data. BMC Bioinformatics. 2024;25:305. https://doi.org/10.1186/s12859-024-05926-z

35. Cook DP, Vanderhyden BC. Context specificity of the EMT transcriptional response. Nat Commun. Nature Publishing Group; 2020;11:2142. https://doi.org/10.1038/s41467-020-16066-2

36. Weirauch MT, Yang A, Albu M, Cote AG, Montenegro-Montero A, Drewe P, et al. Determination and inference of eukaryotic transcription factor sequence specificity. Cell. 2014;158:1431–43. https://doi.org/10.1016/j.cell.2014.08.009

37. Pons P, Latapy M. Computing Communities in Large Networks Using Random Walks. In: Yolum pInar, Güngör T, Gürgen F, Özturan C, editors. Comput Inf Sci - ISCIS 2005. Berlin, Heidelberg: Springer; 2005. p. 284–93. https://doi.org/10.1007/11569596_31

38. Zhao M, Liu Y, Zheng C, Qu H. dbEMT 2.0: An updated database for epithelial-mesenchymal transition genes with experimentally verified information and precalculated regulation information for cancer metastasis. J Genet Genomics Yi Chuan Xue Bao. 2019;46:595–7. https://doi.org/10.1016/j.jgg.2019.11.010

39. Yang J, Leskovec J. Defining and Evaluating Network Communities based on Ground-truth [Internet]. arXiv; 2012 [cited 2026 Aug 9]. https://doi.org/10.48550/arXiv.1205.6233

40. Thuault S, Valcourt U, Petersen M, Manfioletti G, Heldin C-H, Moustakas A. Transforming growth factor-β employs HMGA2 to elicit epithelial–mesenchymal transition. J Cell Biol. 2006;174:175–83. https://doi.org/10.1083/jcb.200512110

41. Jiang Y-Y, Jiang Y, Li C-Q, Zhang Y, Dakle P, Kaur H, et al. TP63, SOX2, and KLF5 Establish a Core Regulatory Circuitry That Controls Epigenetic and Transcription Patterns in Esophageal Squamous Cell Carcinoma Cell Lines. Gastroenterology. Elsevier; 2020;159:1311-1327.e19. https://doi.org/10.1053/j.gastro.2020.06.050

42. Tiwari N, Tiwari VK, Waldmeier L, Balwierz PJ, Arnold P, Pachkov M, et al. Sox4 Is a Master Regulator of Epithelial-Mesenchymal Transition by Controlling Ezh2 Expression and Epigenetic Reprogramming. Cancer Cell. Elsevier; 2013;23:768–83. https://doi.org/10.1016/j.ccr.2013.04.020

43. Jacobi JJ, Wadosky KM, Jaiswal N, Zhang X, Wang Y, Singh PK, et al. EZH2 Suppression Diversifies Prostate Cancer Lineage Variant Evolution and Lacks Efficacy in Inhibiting Disease Progression. Cancer Res. 2026;86:889–908. https://doi.org/10.1158/0008-5472.CAN-25-2747

44. Otero M, Peng H, Hachem KE, Culley KL, Wondimu EB, Quinn J, et al. ELF3 modulates type II collagen gene (COL2A1) transcription in chondrocytes by inhibiting SOX9-CBP/p300-driven histone acetyltransferase activity. Connect Tissue Res. 2017;58:15–26. https://doi.org/10.1080/03008207.2016.1200566

45. Hong D, Messier TL, Tye CE, Dobson JR, Fritz AJ, Sikora KR, et al. Runx1 stabilizes the mammary epithelial cell phenotype and prevents epithelial to mesenchymal transition. Oncotarget. 2017;8:17610–27. https://doi.org/10.18632/oncotarget.15381

46. Fontaine M, Delas MJ, Saez M, Maizels RJ, Finnie E, Briscoe J, et al. Dynamic Landscape Analysis of Cell Fate Decisions: Predictive Models of Neural Development From Single-Cell Data [Internet]. bioRxiv; 2025 [cited 2026 Apr 27]. p. 2025.05.28.656648. https://doi.org/10.1101/2025.05.28.656648

47. Kolberg L, Raudvere U, Kuzmin I, Adler P, Vilo J, Peterson H. g:Profiler-interoperable web service for functional enrichment analysis and gene identifier mapping (2023 update). Nucleic Acids Res. 2023;51:W207–12. https://doi.org/10.1093/nar/gkad347

48. Takemoto T, Uchikawa M, Yoshida M, Bell DM, Lovell-Badge R, Papaioannou VE, et al. Tbx6-dependent Sox2 regulation determines neural vs mesodermal fate in axial stem cells. Nature. 2011;470:394–8. https://doi.org/10.1038/nature09729

49. Galceran J, Fariñas I, Depew MJ, Clevers H, Grosschedl R. Wnt3a−/− -like phenotype and limb deficiency in Lef1−/−Tcf1−/− mice. Genes Dev. Cold Spring Harbor Lab; 1999;13:709–17.

50. Wilm B, James RG, Schultheiss TM, Hogan BLM. The forkhead genes, *Foxc1* and *Foxc2*, regulate paraxial versus intermediate mesoderm cell fate. Dev Biol. 2004;271:176–89. https://doi.org/10.1016/j.ydbio.2004.03.034

51. Metzis V, Steinhauser S, Pakanavicius E, Gouti M, Stamataki D, Ivanovitch K, et al. Nervous System Regionalization Entails Axial Allocation before Neural Differentiation. Cell. Elsevier; 2018;175:1105-1118.e17. https://doi.org/10.1016/j.cell.2018.09.040

52. Koch F, Scholze M, Wittler L, Schifferl D, Sudheer S, Grote P, et al. Antagonistic Activities of Sox2 and Brachyury Control the Fate Choice of Neuro-Mesodermal Progenitors. Dev Cell. Elsevier; 2017;42:514-526.e7. https://doi.org/10.1016/j.devcel.2017.07.021

53. Charité J, de Graaff W, Consten D, Reijnen MJ, Korving J, Deschamps J. Transducing positional information to the Hox genes: critical interaction of cdx gene products with position-sensitive regulatory elements. Development. Cambridge, England; 1998;125:4349–58. https://doi.org/10.1242/dev.125.22.4349

54. Deschamps J, van Nes J. Developmental regulation of the Hox genes during axial morphogenesis in the mouse. Development. Cambridge, England; 2005;132:2931–42. https://doi.org/10.1242/dev.01897

55. Hu D, Marcucio RS, Helms JA. A zone of frontonasal ectoderm regulates patterning and growth in the face. Development. 2003;130:1749–58. https://doi.org/10.1242/dev.00397

56. Sarkar L, Cobourne M, Naylor S, Smalley M, Dale T, Sharpe PT. Wnt/Shh interactions regulate ectodermal boundary formation during mammalian tooth development. Proc Natl Acad Sci U S A. 2000;97:4520–4. https://doi.org/10.1073/pnas.97.9.4520

57. Shi J, Malik J. Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell. 2000;22:888–905. https://doi.org/10.1109/34.868688

58. Kannan R, Vempala S, Vetta A. On clusterings: Good, bad and spectral. J ACM JACM. 2004;51:497–515. https://doi.org/10.1145/990308.990313

59. Zheng GXY, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, et al. Massively parallel digital transcriptional profiling of single cells. Nat Commun. Nature Publishing Group; 2017;8:14049. https://doi.org/10.1038/ncomms14049

60. Hamburger V, Hamilton HL. A series of normal stages in the development of the chick embryo. 1951. Dev Dyn Off Publ Am Assoc Anat. 1992;195:231–72. https://doi.org/10.1002/aja.1001950404

61. Albrecht U, Eichele G, Helms J, Lu H, Daston G. Molecular and cellular methods in developmental toxicology. N Y CRC Inc. 1997;23–48.

62. Germain P-L, Lun A, Garcia Meixide C, Macnair W, Robinson MD. Doublet identification in single-cell sequencing data using scDblFinder. F1000Research. 2022;10:979. https://doi.org/10.12688/f1000research.73600.2

CHAPTER 6: DISCUSSION

This thesis makes two primary contributions to the field of scRNAseq analysis: a framework for developing statistics which accounts for the peculiarities of scRNAseq data and a proof of concept for leveraging gene-gene correlation dynamics to infer cell state transition driver genes.

6.1 BigSur Correlations

BigSur provides the means to calculate and statistically evaluate gene-gene correlations from scRNAseq data. When applied to correlations, it is primarily a tool which prevents the development of false hypotheses (i.e., false positive correlations). In scRNAseq data, this is incredibly impactful as standard correlation measurements are essentially meaningless due to the number of false positives. The results of BigSur are particularly interesting at a module level. There, we can develop macro-level hypotheses about gene co-regulation. Internally, we have also discussed the use of correlation modules as a new form of gene ontology, an idea that I believe holds some promise but has remained unexplored.

It is important to understand that the correlation modules that come out of BigSur are not gene regulatory networks, and BigSur is not a gene regulatory network inference method. Primarily, this is because gene regulatory networks have directed edges; that is, one gene *causes* an increase or decrease in another gene's expression. Causal relationships cannot be inferred from correlations alone — we are only able to hypothesize co-regulation. For example, a correlation may arise because a pair of genes are both downstream of the activation of some third gene. In this case, while you might predict that all three of the genes would correlate with one another, the observed relationship between them is impossible to guess. Time delays between upstream genes and their targets may lead to very weak or, if the delay is long enough, even

negative correlations. A correlation indicates co-regulation, not a true gene regulatory link. This distinction must be drawn if correlations are to be leveraged meaningfully.

Though I focus on the utility of correlation trajectories toward the end of the thesis, single timepoint correlations can also be useful in analyzing cell state transitions. A sample which contains cells in both the start and end stable states is likely to show negative correlations between the cell state markers. Both positively correlated modules on either end give hints at the regulatory processes defining the stable states and may even contain some of the genes that drove the transition. For example, in the St22 *Gallus gallus* analyses, we noticed strong negative correlation between ALX4 and SHH related genes (Fig. 1). These genes are known to have complex interactions with each other in the developing limb bud: ALX4 and SHH are mutually repressive, but ALX4 expression is necessary for proper SHH activation timing[1]. From the correlations of a single timepoint, we were able to form a hypothesis that similar interactions may be driving state changes in the frontonasal process.

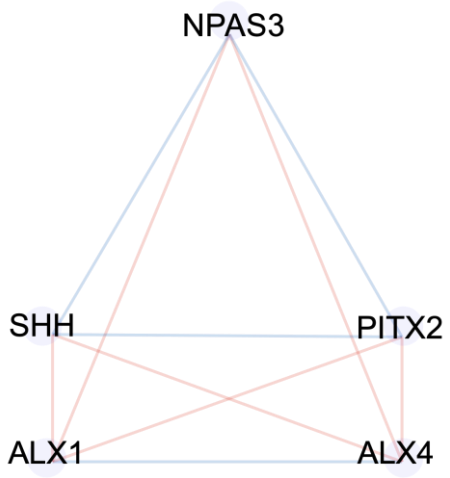


Figure 1. Correlation network showing the relationship between SHH, NPAS3, PITX2, ALX1, and ALX4.
Negative correlations are shown in red, positive in blue. Node size is not indicative of expression level.

Because they often represent the separation of distinct cell states, negative correlations present an interesting alternative feature selection method. This concept has been suggested previously by Tyler *et al*[2] but is made possible by BigSur's improved ability to detect negative correlations in scRNAseq data. A particularly interesting idea would be to replace principal components with gene correlation modules, with weights assigned based on the gene's number of edges or centrality. Because a positively correlated module indicates co-regulation, each cell's position in each module-defined axis should better represent its relative underlying cell state than in principal component space, which simply maximizes variance. We have tested a version of this where, similar to PCA, we create linear combinations of gene expression values weighted by their eigenvector centrality or edge count. Anecdotally, it appears to work well, though more work needs to be done to validate the method.

The BigSur method itself is rigorous and should work well for most scRNAseq data, provided it follows the expected structure of a compound Poisson. We have developed a negative binomial version of BigSur as well, which was not used or discussed in this work. Anecdotally, the differences between the Poisson log-normal and negative binomial versions are minor and the recovered gene-gene correlations are mostly consistent. Qualitatively, these distributions are quite similar until you reach the upper moments.

The main concern with the program is a practical one: both the modified corrected Pearson residual and cell-specific expectation matrices are dense. This makes the analysis of large datasets very expensive, a problem that we are constantly looking for solutions to. Deeply sequenced data and large numbers of cells also add to the computational costs of *p*-value estimation as fewer pairs are pre-filtered before the root finding step.

6.2 BigSur Feature Selection

The use of modified corrected Pearson residuals enabled us to develop the modified corrected Fano factor for use in feature selection. In this work, we were able to demonstrate that many clustering problems are easy and feature selection is mostly meaningless. However, we also showed that it is very important to carefully choose clustering features when differences between cell states in the data are subtle. The standardly used set of feature selectio n software tends to select a very large number of genes by default, often in the thousands — adding too much noise for the weaker signal of rare or similar cell states to overcome.

While the modified corrected Fano factor provides a statistical framework by which to select features in a more principled manner, its use in clustering is not without flaws. As a measure of variance, genes which are expressed very highly in a small number of cells (e.g., hemoglobin genes from contamination) are often assigned the highest Fano factors. While this is correct, the use of these genes in clustering is often undesired as they are mostly invariant for the vast number of cells. While we could simply limit the selected genes to those with Fano factors below a certain level, a more interesting solution would employ population-level statistics such as the Gini coefficient. Thresholding the genes by Gini coefficient would allow the user to control the scale at which they want to cluster their cells.

Further, it is unclear that measures of variance are the best way to separate cell states. Many of the highly central genes that we find by looking at negatively correlated modules have relatively low modified corrected Fano factors. Despite this, using these genes as features clusters the cells relatively well. There are many opportunities for feature selection techniques to be further optimized, and measurements other than variance should be taken into further consideration.

6.3 Correlation trajectory analysis

Finally, we were able to demonstrate that applying critical transition theory to scRNAseq analysis enables the identification of cell state transition drivers. This becomes evident even in a method as crude as the one applied: a simple correlation trajectory analysis across multiple sequenced timepoints recovered known drivers in well-studied cell state transitions and identified novel drivers in the developing chick. Particularly exciting was the level of separation between the different modules of genes observed in the NMP data. This indicates that we may be able to infer novel instances of fate divergence by assessing correlation network statistics such as the conductance.

As was discussed previously, this method, by design, ignores the problem of differentiating between intra- and extra-cell state heterogeneity. Heterogeneity is a tricky concept in the analysis of cell state transitions. Cells are expected to become more heterogeneous as they transition, but observing an increase in heterogeneity is only useful in analysis if it derives only from that set of transitioning cells. This is almost oxymoronic – we would ideally want to observe a set of heterogeneous homogeneous cells. In a true time-series scRNAseq experiment, we are guaranteed to have an increase in cell state heterogeneity at some point across the transition. Necessarily, some cells will have entered the new stable state while others are still transitioning. It is likely that some proportion of candidate driver genes will simply be cell state markers, even among those participating in many peaking pairs. However, it is unclear how the prominence of peaks driven by heterogeneity would compare to those between true driver genes.

On the surface, pseudotime or RNA velocity trajectory construction appears to be a strong solution to this problem. But how do you choose which genes to use to construct the trajectory? Oftentimes, the list of highly variable features is used for this task, which we have

shown tend to add noise by containing too many genes. But even if you use the modified corrected Fano factor and threshold the *p*-value, the problem is not necessarily solved as it is unclear that variance measurements would predict an expression trajectory well.

When applied to heterogeneous cells, trajectory inference can generate spurious correlation peaks. Consider a gene regulatory module that is off in the initial cell state and activates during a transition. Ordering cells along this module would correctly recover the transition dynamics. However, if the data also contains an unrelated cell type that expresses the same genes at more moderate levels, those cells will be placed near the midpoint of the inferred trajectory. Their markers would appear to rise and fall across the trajectory, producing correlation peaks which simply reflect the misordering of cells rather than any real signal of cell state transitions.

Additionally, in an inferred trajectory, a window size must be selected to perform correlation analysis. If each window has the same number of cells, a small window will increase homogeneity among the cells and time-scale resolution but would also negatively impact the number of correlations that are found significant. However, because transitioning cells are often traversing across an unstable part of gene expression space, they will be fewer in number than those in any given stable state in typical scRNAseq datasets (e.g., McKellar *et al*[3]). Choosing a window which is too large may capture too many heterogeneous cell states. The alternative, creating equal sized pseudotime windows, runs into the issue of variable sample sizes with fewer cells in each bin across the transition (middle) timepoints.

Thus, it is imperative to carefully consider the conditions under which trajectory inference is applied before using it in conjunction with tipping point theory.

However, the presence of heterogeneity does not make this method useless. We were able to pick up the signal of both *PITX2* and *NPAS3* in the *Gallus gallus* data despite clear heterogeneity among the cells. Heterogeneity will add noise (i.e., false positives) to the candidate list, but real driver genes should continue to follow the expected dynamics as well. Of course, there is a limit to that statement: in the case where stable state cells vastly outnumber transitioning cells at every timepoint, noise may prevent the observation of correlation peaks.

Time-scale resolution of the data is clearly important in correlation peak analysis. In the NMP data, we were able to, albeit weakly, discern that the two differentiation arms occurred at different times. This would not have been possible to observe with only three timepoints, the minimum number necessary to use this method. Perhaps more importantly, though, is the effect that it might have the inferred gene modules. If the NMP dataset consisted only of D3, D3.6, and D4, it is possible that many of the pairs which peaked at D3.4 (the median peak timepoint for both the pre-neural and fate decision modules) would not have been detected. Particularly for multi-directional cell state transitions, the analysis quality will improve with the inclusion of more sequenced timepoints.

However, the transitions that we studied here are slow, occurring over the course of days. It may not be feasible to sequence, say, a group of cells transitioning into a stressed state driven by the rapid expression of immediate early response genes at meaningful timepoint intervals. In these situations, one might expect the unstable transition states of the cells to be extremely short lived, such that, at any sequenced timepoint, the number of cells actively transitioning cells is much fewer in number than those in stable states. Again, trajectory inference is not necessarily a solution in these cases. It has been noted that these methods fail to work well on systems such as macrophage stimulus response[4].

The best metric for ranking candidate driver genes is still an open question. The two presented here, ranking individual pairs by prominence or individual genes by the number of peaks, work but can be optimized further. A correlation's statistical significance is tied to the expression levels of the two participating genes. As such, any summary statistic for an individual gene will be, in part, a function of its expression, including the number of peaks it participates in. It should be possible to normalize for this, but it is not a simple task. A normalization technique that simply scales the number of peaks by the gene's expression level will inevitably cause lowly expressed genes with very low numbers of peaks to be ranked highly. This can be partially addressed by penalizing genes with few peaks, but any such penalty would be chosen *post hoc* rather than derived, thus introducing an arbitrary parameter into an otherwise rather simple statistic and distorting the ranking in an unjustified manner.

Ranking gene pairs rather than individually is robust to the expression level of the participating genes, and the pairs themselves create interesting hypotheses about co-regulation. Despite this, it is unclear how to compare the candidacy strength between individual genes using the paired ranking. Is a gene which appears once among the top ten highest prominence pairs more likely to be a driver gene than one that appears in every pair between rank ten and fifteen? The inability to assess genes individually is a real weakness of this ranking.

The two ranking methods complement one another by addressing each other's primary weakness. However, it remains unclear how these two metrics can be combined to form a more objective ranking of gene importance. In the *Gallus gallus* data, we arbitrarily chose to observe *PITX2* and *NPAS3* rather than other genes which appear highly ranked in both lists (e.g., *FOS*, *MEIS1*). For an unbiased list, ideally these metrics would be combined into a single ranking, removing the need to choose which candidates to test first *ad hoc*.

This brings me to experimental testing of candidate driver genes, which is not optional. Gene pairs can peak in correlation for many reasons, only one of which is an underlying critical transition. scRNAseq analyses are great for forming hypotheses but cannot establish causal relationships.

6.4 The Utility of Applied Bias in scRNAseq Analysis

In each of these sections, I have discussed the use and limitations of the described methods in producing unbiased results and measurements. The pursuit of unbiased analysis has become a dominant value in the field, and I would argue this has both limited the utility that has been drawn from scRNAseq data and introduced new forms of bias which are less visible than those that they were meant to overcome.

Standard scRNAseq analysis pipelines make multiple arbitrary choices to ensure unbiased analysis. For example, even if we selected the perfect set of features for clustering, those features are then typically packaged into principal components. While the use of principal components is an unbiased method of removing noise, an arbitrary number of them, often 20-50, are the actual input into the clustering software. Just as with selecting features, choosing the wrong number of principal components can lead to the addition of too much noise or the omission of too much signal to achieve good clustering.

Further upstream, read alignment must contend with the question of where to assign reads that score equally for two different sequences, as is often the case when pseudogenes are present. Some programs will choose to randomly assign the count to one of the equally scoring genes[5,6] while others will choose to drop the read completely[7]. Either way, noise has been introduced, irrevocably altering the data. Noise caused by these arbitrary decisions compounds and, just as with feature selection, can lead to major differences in results.

It is important for the field to continue to push further forward with new, and exciting downstream analysis techniques; it is equally, if not more, important that we continue to ask questions about where our processed data comes from and to understand how upstream decisions bias results.

Prominence Threshold	*SHH* Module Size	# *SHH* peaks	*SHH* in module with *NPAS3/PITX2?*
None	952	75	Yes
0.01	1322	9	Yes
0.015	924	3	Yes
0.02	277	2	No
0.025	154	1	No
None, selected gene set	81	10	Yes

Table 1. No threshold both reduced the SHH-containing module to a testable size and retained its association with NPAS3 and PITX2. Only modules above a size of 10 are reported.

On the other hand, purposely biased choices in analysis can often prevent the need for arbitrary decision-making in analytical techniques. In the *Gallus gallus* data, the correlation peak analysis was performed only on the set of all transcription factors plus hand-selected genes *SHH*, *PTCH1*, *PTCH2*, and *WNT7B*. Separately, I ran the same analysis on the set of all genes. The module holding *SHH* contained 951 other genes, too many to reasonably consider testing experimentally (though still appearing in a module with both *PITX2* and *NPAS3*). One solution here is to tune the prominence threshold to remove more genes from the analysis, but even a threshold as low as 0.02 was enough to remove all but two peaks shared with *SHH* from consideration and place it in a separate module from both *NPAS3* and *PITX2*. By only

considering a hand-selected set of genes, we were able to avoid the threshold entirely and still identify a promising module of *SHH*-related genes. If a threshold exists which both reduces the module to a testable size and captures the *SHH/NPAS3/PITX2* relationship, it is not easily found, and there would be no way to know what real relationships are discarded through its use.

Biased choices are useful in other situations as well. If a gene is known to be differentially expressed between cell states in the data, but does not appear in in the list of highly variable genes, there is little reason to exclude it. We already do implement biased decisions in standard pipelines e.g., removing cells expressing high levels of mitochondrial genes. Should we include those cells in the pursuit of true unbiased analysis?

I am not suggesting that we should begin leveraging scRNAseq analysis as a tool to practice confirmation bias. Instead, I am encouraging those who have deep biological understanding to make use of their expertise. Employing prior knowledge can lead to discoveries that unbiased analysis would miss[8].

6.5 References

1. Kuijper S, Feitsma H, Sheth R, Korving J, Reijnen M, Meijlink F. Function and regulation of *Alx4* in limb development: Complex genetic interactions with *Gli3* and *Shh*. Dev Biol. 2005;285:533–44. https://doi.org/10.1016/j.ydbio.2005.06.017

2. Tyler SR, Lozano-Ojalvo D, Guccione E, Schadt EE. Anti-correlated feature selection prevents false discovery of subpopulations in scRNAseq. Nat Commun. Nature Publishing Group; 2024;15:699. https://doi.org/10.1038/s41467-023-43406-9

3. McKellar DW, Walter LD, Song LT, Mantri M, Wang MFZ, De Vlaminck I, et al. Large-scale integration of single-cell transcriptomic data captures transitional progenitor states in mouse

skeletal muscle regeneration. Commun Biol. Nature Publishing Group; 2021;4:1280. https://doi.org/10.1038/s42003-021-02810-x

4. Sheu KM, Pimplaskar A, Hoffmann A. Single-cell stimulus-response gene expression trajectories reveal the stimulus specificities of dynamic responses by single macrophages. Mol Cell. 2024;84:4095-4110.e6. https://doi.org/10.1016/j.molcel.2024.09.023

5. Langmead B, Salzberg SL. Fast gapped-read alignment with Bowtie 2. Nat Methods. 2012;9:357–9. https://doi.org/10.1038/nmeth.1923

6. Langmead B, Trapnell C, Pop M, Salzberg SL. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. Genome Biol. 2009;10:R25. https://doi.org/10.1186/gb-2009-10-3-r25

7. Zheng GXY, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, et al. Massively parallel digital transcriptional profiling of single cells. Nat Commun. Nature Publishing Group; 2017;8:14049. https://doi.org/10.1038/ncomms14049

8. Ideker T, Dutkowski J, Hood L. Boosting Signal-to-Noise in Complex Biology: Prior Knowledge is Power. Cell. 2011;144:860-3. https://doi.org/10.1016/j.cell.2011.03.007