UNIVERSITY OF CALIFORNIA,
IRVINE

Spatiotemporal Neural Activity Tracking and Continuous Transcriptional Variation of Neurons

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Mathematics

by

Kevin Johnston

Dissertation Committee:
Chancellor's Professor Qing Nie, Chair
Professor Xiangmin Xu
Professor Long Chen

2022

DEDICATION

To my wife for her patience in allowing me to finish, and to my grandparents who never got to see it.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

I would like to thank Dr. Qing Nie and Dr. Xiangmin Xu for supporting me during completion of this work, Dr. Zhaoxia Yu, Dr. Steven Grieco, Dr. John Guzowski, Dr. Zhiqun Tan, and Rachel Crary for their assistance in data production and analysis, The NIH (NS104897, R01 AG06715), the NSF (DMS1763272), the University of California, Irvine Mathematics department, and the Simons Foundation (594598) for funding assistance.

The text of portions of Chapters 2 and 3 of this dissertation are a reprint of the material as it appears in [19], used with permission from Elsevier. The co-authors listed for this publication are Steven Grieco, Hai Zhang, Suoqin Jin, Xiangmin Xu, and Qing Nie. The text of portions of Chapter 4 of this dissertation are a reprint of the material as it appears in [20], used with permission from SpringerNature. The co-authors listed for this publication are Yanjun Sun, Suoqin Jin, Xiaoxiao Lin, Lujia Chen, Xin Qiao, Li Jiang, Pengcheng Zhou, Peyman Golshani, Qing Nie, Todd Holmes, Douglas Nitz, and Xiangmin Xu. The co-author Xiangmin Xu listed in this publication directed and supervised research which forms the basis for Chapter 4 of this dissertation. The text of portions of Chapter 5 are a reprint of material from the text "Parvalbumin-cholecystokinin co-expressing interneurons and their implications in neuropsychiatric disorders". The co-authors listed for this manuscript are Steven Grieco, Bryan Tang, Pan Gao, Bernardo Garduno, Gregory Horowitz, Zhaoxia Yu, Qing Nie, Todd Holmes and Xiangmin Xu. The co-author Xiangmin Xu listed for this manuscript directed and supervised research which forms the basis for Chapter 5 of this dissertation. This manuscript has not yet been accepted for publication, and is included with the co-authors permission.

VITA

Kevin Johnston

EDUCATION

Doctor of Philosophy in Mathematics **2022**
University of California, Irvine *Irvine, California*

Masters of Science in Mathematics **2016**
Brigham Young University *Provo, Utah*

Bachelors of Science in Mathematics **2012**
Southern Utah University *Cedar City, Utah*

RESEARCH EXPERIENCE

Graduate Research Assistant **2020–2022**
University of California, Irvine *Irvine, California*

TEACHING EXPERIENCE

Teaching Assistant **2016–2020**
University of California, Irvine *Irvine, California*

Instructor **2016**
Brigham Young University *Provo, Utah*

REFEREED JOURNAL PUBLICATIONS

Johnston KG, Grieco SF, Zhang H, Jin S, Xu X, and Nie Q "Tracking longitudinal population dynamics of single neuronal calcium signal using SCOUT" **2022**
Cell Rep Meth 2(5)

Grieco SF, Qiao X, Johnston, KG, Chen L, Nelson RR, Lai C, Holmes TC, and Xu X "Neuregulin signaling mediates the acute and sustained antidepressant effects of subanesthetic ketamine" **2021**
Translational Psychiatry 11(1)

Sun Y, Jin S, Lin X, Chen L, Qiao X, Jiang L, Zhou P, Johnston KG, Golshani P, Nie Q, Holmes TC, Nitz DA, and Xu X "CA1-projecting subiculum neurons facilitate object-place learning" **2019**
Nature Neuroscience 22(11)

SUBMITTED PUBLICATIONS

Grieco SF, Johnston KG, Tang B, Gao P, Garduno BM, Sun Y, Horwitz G, Yu Z, Nie Q, Holmes TC, and Xu X "Parvalbumin-cholecystokinin co-expressing interneurons and their implications in neuropsychiatric disorders" **2022**
Molecular Psychiatry

Zhang H, Chen L, Johnston KG, Crapser J, Green K, Ha N, Tenner A, Holmes TC, Nitz D, and Xu X "Degenerate Mapping of Environmental Location Presages Deficits in Object-Location Encoding and Memory in the 5xFAD Mouse Model for Alzheimer's Disease" **2022**
Journal of Neuroscience

SOFTWARE

SCOUT `https://github.com/kgj1234/SCOUT`
Cell Extraction and Tracking Algorithms

Place Cell Decoding `https://github.com/kgj1234/Machine-Learning-for-Position-...`
Machine learning model for decoding position from place cell signaling.

ABSTRACT OF THE DISSERTATION

Spatiotemporal Neural Activity Tracking and Continuous Transcriptional Variation of Neurons

By

Kevin Johnston

Doctor of Philosophy in Mathematics

University of California, Irvine, 2022

Chancellor's Professor Qing Nie, Chair

Identifying and tracking cell location in long-term longitudinal studies is critical for identifying large-scale time-dependent neuronal activity variations. Population cell tracking across multiple sessions is complicated by non-rigid transformations induced by noise, cell movement and imaging field shifts. In this text we develop SCOUT (Single-Cell SpatiOtemporal LongitUdinal Tracking), a fast, robust cell tracking method utilizing multiple cell-cell similarity metrics, probabilistic inference, and an adaptive clustering methodology to perform cell identification across multiple calcium imaging sessions. We then apply SCOUT to study variations of firing activity coincident with contextual discrimination and neural circuit negation. Next, we investigate the relationship between firing activity and transcriptomics in a single cell type, showing that transcriptional gradients can be associated to subtle variations in neuronal firing activity, which then motivates the development of scGradient, a machine learning algorithm for identifying continuous transcriptional gradients across and within cell types.

Chapter 1

Introduction

In many ways, the past decade has been the decade of single-cell analysis, in which optical imaging methods [1] and RNA sequencing [2] have gone from bulk analysis methods to high throughput production lines ([3] among many others), enabling virtually any lab to begin interrogating functional activity and transcriptomics in most biological systems. Beyond transcriptomics, single-cell analysis of epigenomics [4], spatial transcriptomics [5], and DNA methylation [6], as well as "multiomics" [7] (simultaneous profiling of two or more of these cellular aspects) have become commonplace. With new data comes new challenges in analysis. In many ways, data production has outpaced our analysis capabilities. We have only scratched the surface of the information contained in the data we have already generated.

An enormous amount of energy and funding has been directed towards single-cell production and analysis of neural data[1]. This is unsurprising given the brain is the most complex biological system we have identified and is critical for human functionality. While great progress has been made in understanding functional processes and mitigating decline and failure in other organs, the brain defies characterization through its complexity [10].

[1] the Allen institute has obtained upwards of $1 billion in funding over the last decade [8], with the BICCN started in 2017 having publications consisting of nearly the entirety of the October 7, 2021 Nature publication [9].

At the single-cell level, neurobiology has focused on four main aspects of the neuron[2] [11]: morphology, connectivity, firing activity, and gene expression. Neuron morphology involves study of the physiology of individual neurons. Morphology has enormous impact on the functional properties of individual neurons, and partially governs neuron connectivity. Unlike many other cells, neurons make wide ranging connections with other neurons across large distances. This communication is primarily handled through the axons and dendrites of the neuron, with axons emitting signals, and dendrites receiving signals. Generally, axons and dendrites do not make physical contact with other neurons, instead communicating via neurotransmitters over small gaps (synaptic clefts) between axons and dendrites of communicating neurons.

Analysis of neuron firing activity, the primary communication method in the brain, has seen enormous progress in large scale investigations using calcium imaging technology [12]. Briefly, neurons (in a resting state) exhibit a negative electrical potential. When dendrites receive neurotransmitters across the synaptic cleft, ion channels open in the dendrite which depolarize (or repolarize in the case of inhibitory neurons) the dendrite. This electrical change is transmitted to the cell body (combining with signals from other dendrites) producing depolarization in the soma. If this reaches a critical threshold, an action potential is produced (a large, sudden depolarization event) which propagates down the axon, which releases neurotransmitters from the terminal button into synaptic clefts. This electromolecular signaling forms the fundamental basis for neuron communication.

Action potentials are associated with an influx of calcium (Ca^{2+}) ions, which can be measured with genetically encoded calcium indicators (GECIs) such as the GCaMP variants [1]. Most GCaMP variants respond on slower time scales than those on which action potentials occur, meaning that calcium signaling does not measure precise action potentials, but instead produces a reliable indicator of how much firing activity is occurring in each time

[2]the principal communicatory brain cell

frame, based on the signal intensity. The Miniscope [13] enables 1-photon Ca^{2+} imaging in freely moving mice and can typically image hundreds of neurons simultaneously. This contrasts with the standard electrophysiological approaches which measure individual action potentials, but are unable to record large numbers of neurons simultaneously.

At the single-cell level, some of the most dramatic changes in the understanding of neuron function have been due to advances in transcriptomic profiling. Single-nuclei RNA sequencing[3] has enabled systematic analysis of the transcriptome, identifying new cell types and their developmental origins[4].

While clearly both neuron communication and transcriptomics are of critical importance, single-cell methods for simultaneous interrogation of individual neuron activity, transcriptomics and morphology have not reached high-throughput levels [17], and most experiments consist of cells on the order of thousands [18].

Computational problems regarding longitudinal studies of firing activity, and the relations between neural activity present significant challenges that have yet to be adequately addressed. My primary contributions have been to develop analysis tools for disparate aspects of transcriptomics and longitudinal neural activity studies.

In Chapter 2, we begin by focusing on optical recording methods for neural activity, primarily on the technical challenges for extraction and tracking of neurons across multiple sessions. We develop SCOUT, a method for tracking neurons across multiple sessions. In Chapter 3, we will discuss results from testing the applications developed in Chapter 2, (Chapters 2 and 3 published in [19]). In Chapter 4, we discuss applications of calcium imaging in a long term contextual discrimination experiment, and develop a neural network algorithm used to identify changes in neural activity during experiments (portions published in [20]).

[3] snRNA-seq is frequently preferred to scRNA-seq due to the difficulty in dissociating neurons, and snRNA-seq's friendliness to frozen samples, [14], [15]

[4] perhaps the most well-known being classification of interneurons based on the ganglionic eminences from which they derive [16]

Next, we will discuss the relationship between the transcriptome and neuron firing activity. In Chapter 5 we motivate this discussion by analyzing Patch-seq data, which interrogates firing activity and gene expression in the same cell. In doing so we identify a gene expression gradient that strongly correlates with variations in firing activity (publication under review). In Chapter 6 we develop a modular neural network framework (scGradient) that enables probabilistic identification of celltypes, while simultaneously identifying transcriptional gradients among cells. we then posit how this framework can be modified to incorporate spatial coordinates for analysis of spatial transcriptomics.

Chapter 2

Improving automated optical recording extraction and cell tracking via SCOUT

2.1 Calcium imaging recordings

2.1.1 Optical recording extraction

Extracting longitudinal activity from large-scale neuronal ensembles is a fundamental first step to the analysis of neural circuit responses. Ca^{2+} imaging of population neurons allows the recording of larger neural ensembles than can be recorded using electrophysiology. *In vivo* calcium imaging using microendoscopic lenses enables imaging of previously inaccessible ensembles of neuronal populations at the single-cell level in freely moving mice as they perform neural transformations that underlie behavioral responses over both short and long timescales [21–23]. Microendoscopic *in vivo* brain imaging via head-mounted fluorescent miniature microscopes ("miniscopes") are used widely to study neural circuits in various brain regions [24–33].

Optical recordings take the form of grayscale videos (e.g. m x n x T matrices, where [m,n] is the image resolution in pixels, and T is the number of frames). These videos contain neuron spatial footprints (a subset of the pixels corresponding to individual neurons) with pixel intensity correlating to the associated neural activity. Calcium imaging extraction produces two outputs from each session: a set of spatial footprints (consisting of pixel intensity values corresponding to each neuron for a given recording session), and the temporal signal (extracted calcium traces; $\Delta F/F$).

One class of methods for signal extraction involves semi-manual ROI selection. Such methods include manual ROI selection of individual neuron footprints, and subsequent deconvolution of the neural trace, as well as methods such as convolutional neural networks (CNNs) which use a corpus of identified footprints to train a neural network to identify footprints in future experiments [34], followed by a second step in which temporal fluorescence traces are extracted based on the proposed footprints. However, such methods become computationally intractable when considering large cell population and become less accurate when considering neurons exhibiting strong spatial overlaps between footprints.

Another class of methods involves automated ROI construction, where both fluorescent traces, and spatial footprints are extracted simultaneously. The simplest such example is PCA/ICA [35], in which PCA and ICA are successively used to isolate and extract spatial footprints and calcium activity from optical recordings. These methods rely on linear demixing and can produce significant error when neuron footprints exhibit strong spatial overlaps [36].

The most recent class of methods uses non-negative matrix factorization. I first describe the basic algorithm and then present some variations on the theme that significantly improve results for one-photon imaging (which typically exhibits lower signal-to-noise ratios, with more complex spatiotemporal background).

Given a recording, let d represent the number of pixels in the field of view, T the number of frames observed, and K, the number of neurons in the field of view. Then let $Y \in R^{dT}$ represent the initial calcium fluorescence recording; let $A \in R^{dK}$, the spatial footprints of the neurons, with each column representing the footprint of a single neuron; let the rows of $C \in R^{KT}$ represent the fluorescent signal of each neuron at each frame. The goal is to find A *and* C such that $\|Y - AC\|_F$ is minimized, which can be interpreted as determining the optimal spatial footprints and fluorescence traces to reconstruct the recording.

However, the simplistic model is unable to account for noise (which can be semi-explicitly modeled), as well as the structure of the fluorescent traces. This motivates the inclusion of constraints on A and C, as well as inclusion of a constant background term (this model is called CNMF, constrained non-negative matrix factorization).

The ith row of C is represented as an autoregressive process, where $c_i(t) = \sum_{j=1}^{p} \gamma_j^{(i)} c_i(t-j) + s_i(t)$, and $s_i(t)$ represents the number of spikes fired by the i-th neuron in the tth frame, and S, the matrix of spikes, is constrained to be sparse. The footprint matrix A is also constrained to be sparse, and the authors introduce background $B = \boldsymbol{b_0} \cdot \boldsymbol{1}^T$ which models a constant background for each pixel. At this point, we attempt to minimize the quantity $\|Y - (AC + B)\|_F$ in order to identify spatial footprints A and temporal traces C.

The final advancements of this model are the inclusion of a variable fluctuating background term. In this model, the background is decomposed as $B = B^f + B^c$ where B^c models the constant baseline background, and B^f models fluctuating background activity. To model the fluorescent background, the authors assume that background sources (typically due to blurred and out of focus fluorescence) exhibit a coarser spatial profile than neurons. If l is the typical neuron diameter, B^f for each pixel is implicitly modeled as a linear combination of the background of nearby pixels excluding those within Euclidean distance l of the pixel. In practice, this model is overly complicated, and is replaced by the closed form formula $B^f = W(Y - A\dot{C} - B^c)$, where W is a matrix of weights in which nearby pixels have weight

0 (this model is called CNMF-E, CNMF for microendoscopic data).

Initialization for neuron centers uses a greedy algorithm, such that a proposed pixel satisfies two criteria: a minimum threshold on peak-to-noise ratio (calculated as peak signal strength divided the standard deviation of the noise), and a sufficiently high temporal local correlation (implying strong similarities in temporal signal for pixels surrounding the proposed center) (**Figure 2.1 A-C**, [37]). Initialization of variables C and B, as well as updates for the background B and iterative fitting algorithms are discussed in the original paper [38].

2.1.2 Optical recording alignment

In addition to extraction of the optical recordings, there are technical difficulties relating to both and between session alignment. Regarding within session alignment, animal movement during recording causes FOV movement on a frame by frame basis throughout individual recordings. The most common adjustment algorithm used for motion correction is NoRM-Corre [39], which splits the field of view into overlapping spatial patches in each direction, registers them using a rigid translation to a template (which is sequentially updated), and then up-samples to create a smooth motion field that approximates non-rigid artifacts in a piecewise rigid manner.

The second alignment issue is alignment between sessions. This process is complicated, as neurons exhibit periods of inactivity (e.g. they may be inactive in some sessions) and FOVs may exhibit significant variation between sessions (depending on the setup and the consistency of the individuals who set up the recordings).

2.2 SCOUT overview

We now introduce SCOUT (Single-Cell SpatiOtemporal LongitUdinal Tracking) which consists of three components. First, a preprocessing module for alignment and motion correction of optical recordings, second, an extraction module modified version of CNMF-E that includes a new spatial filter based on expected spatial footprint shapes, and third, a cell tracking module utilizing a novel probabilistic algorithm that incorporates both spatial and temporal metrics to identify neurons across multiple sessions.

2.3 Preprocessing and session registration

The preprocessing module is built on NoRMCorre, MATLAB, and the fordanic image registration library. This module motion corrects and semi-automatically aligns sessions. After motion correction, an initial registration using matlab build in code is checked by the user (if this registration is flawed, the user may manually select points of similarity between sessions), followed by non-rigid intensity registration using the optical-flow algorithm.

2.4 Correcting and filtering spatial footprints with SCOUT

One major issue for current extraction methods is the prevalence of false discoveries, which consist of extracted footprints and temporal traces that do not correspond to ground truth neurons in the recording (**Figure 2.1 D**). These false discoveries can be caused by background noise, inaccurate initialization of neuron footprints, or errors in the estimation of

footprints and corresponding temporal traces. For *in vivo* recordings, depending on recording quality and initialization parameters, false discovery rates may reach 45% of detected neurons.

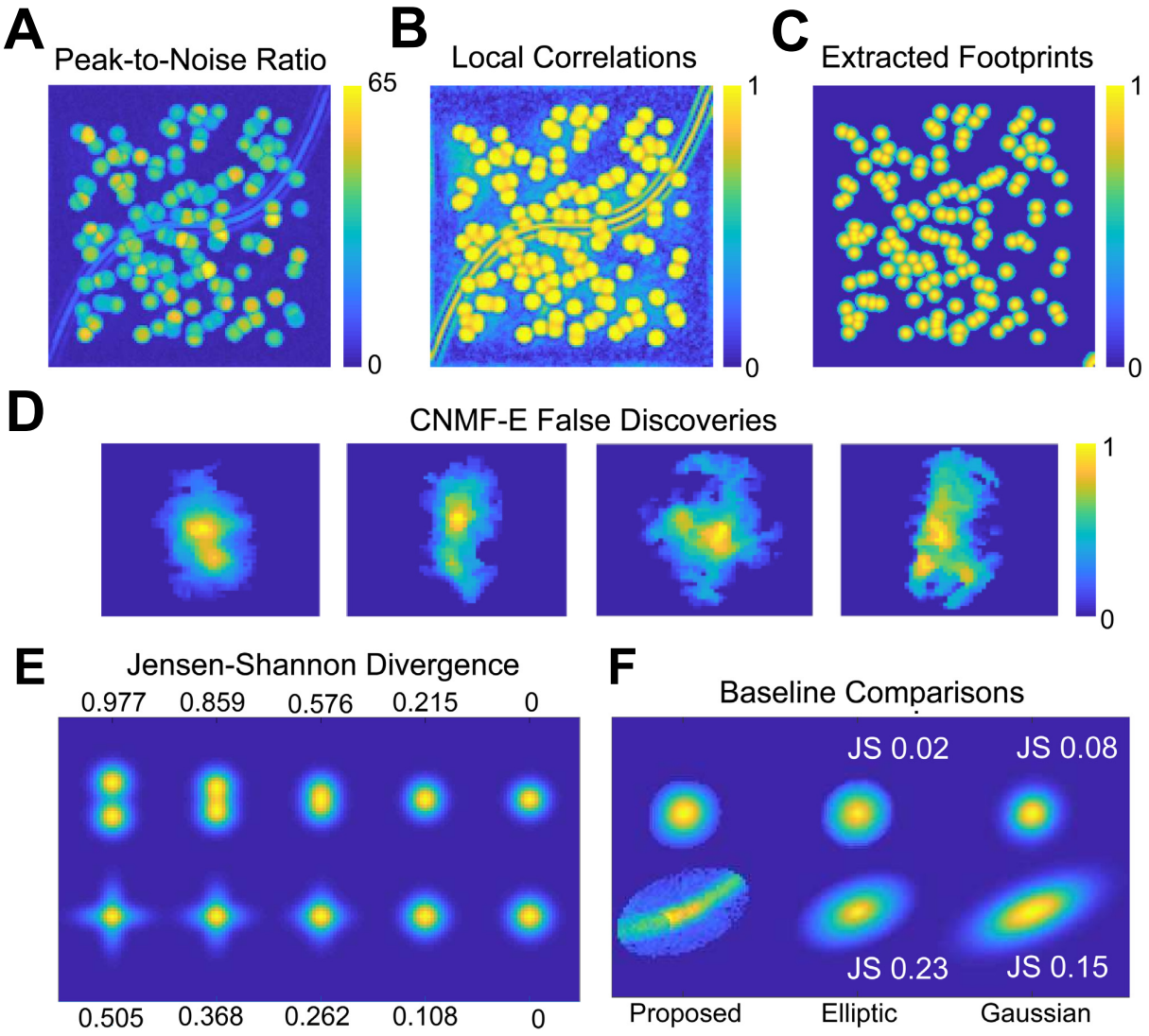

Figure 2.1: **A-B**: The local correlation image (**B**) and peak-to-noise ratios (**A**) are used as initialization points for neurons in CNMF-E and SCOUT. Extracted footprints (using SCOUT, **C**) show strong fidelity to the initialization values, and filter out most of the background activity. **D**: Examples of false discoveries detected by CNMF-E on in vivo recordings of the CA1 hippocampus. **E**: Examples of Jensen-Shannon divergence between Gaussian probability distributions which are (1st row) gradually translated apart or (2nd row) stretched in orthogonal directions. **F**: A true (top row) and false (bottom row) footprint discovery is shown compared with the elliptic comparison (2nd column) and a Gaussian comparison (3rd column). The elliptic comparison shows a stronger differentiation between the false and true discovery.

Some approaches have been suggested to address this issue, such as manual false discov-

ery identification [38], and convolutional neural network classification [40], performed at intermediate stages of the extraction. However, for experiments involving large numbers of recordings, manual identification of false discoveries becomes untenable. Convolutional neural networks have been used to detect false discoveries, but they lack interpretability, and struggle to generalize to neurons with different footprint profiles than those in the training set.

In SCOUT, after each iteration of CNMF-E, an initial pre-processing begins in which proposed spatial footprints are thresholded based on maximum pixel intensity, removing low intensity ($<$10% of maximum intensity) pixels. Each footprint is normalized so that the sum of pixel intensities in each footprint equals 1, allowing us to view each spatial footprint as a discrete probability distribution. Each footprint is then compared with a baseline distribution, using Jensen-Shannon (JS) divergence as a metric [41]. Subsequently, the footprints with similarity exceeding the baseline of a specified JS threshold are removed, and the remaining footprints are updated either by averaging with the baseline, or by setting the pixel intensities of all points not in the support of the baseline to 0. Using an iterative process employed by the CNMF-E, the spatiotemporal traces corresponding to each footprint are then updated by removing any non-zero intensity levels not in the support of the comparison footprint, after which the remaining intensity values are rescaled to their original magnitude. Note that varying the JS threshold changes the sensitivity of the algorithm (**Figure 2.1 E**), allowing the user to optimize the tradeoff between false discoveries and true positives. Baseline neurons footprints are sampled from a user determined probability distribution, with parameters sampled from the proposed distribution.

While the construction of the baseline is customizable, we consider several options for 1-photon data (**Figure 2.1 F**). One option is a Gaussian model. Mean and covariance are sampled from the normalized, thresholded footprint P, and used to construct a comparison footprint Q. Gaussian models overestimated the rate of decrease in signal intensity when

moving from the center of the proposed neuron to its boundary. Another is an elliptical comparison created by calculating the centroid from the footprint P and the rate of signal strength intensity decrease along the major axis of the footprint. The intensity values of the footprint are interpolated along the major axis using a fractional polynomial model, and the fractional polynomial is rotated around the centroid of the footprint, linearly scaled so as to decrease to the width of the minor axis after a rotation of 90^{o}, to create an elliptical model for the neurons spatial footprint. This is the method used in the experimental results discussed in later, as it appeared to show greater differentiation between true and false discoveries, than the Gaussian model.

For 2-photon data, we note that neuronal footprints have characteristic ring shapes which were modeled using the same Gaussian/polynomial models as for 1-photon data, followed by removing the regions with intensity greater than 0.5 (on a scale of 0-1), essentially removing the centers. This aspect of SCOUT is not as relevant as the signal-to-noise ratio (SNR) in 2-photon data is significantly higher than 1-photon data, and comparably fewer false positive emerge.

2.5 Tracking neurons across multiple sessions with SCOUT

Experiments that require the accurate identification of neurons across multiple recording sessions have proven difficult, as cell movement, shifts in field of view, and inaccuracies in the extraction of neural activity from session recordings complicate this task. Most previous attempts to track the activity of neurons over long term experiments have taken one of three forms (**Figure 2.2 A-B**). 1) Initial concatenation of registered recordings followed by extraction of fluorescence traces and spatial footprints from the concatenated recording [28]. 2) Concatenation followed by splitting the spatial dimension into overlapping patches.

Extraction is performed on each patch separately, and neurons are merged across the patches, giving extracted footprints and neural signals through the full recording [38]. 3) Extraction of neural signals from each session independently, followed by tracking cells across recordings via spatial similarities in the extracted neuron footprints [24**?**].

Concatenation and patch methods can be resource intensive in terms of both computational power and time while requiring neuron spatial footprints to remain in constant position over time. In terms of scalability, cell tracking methods may provide the best option for long-term neural ensemble analysis, but several factors complicate the cell tracking task.

First, imperfect motion correction or low signal-to-noise ratio can reduce the quality of neuron extractions leading to false discoveries (neurons identified by extraction algorithms corresponding to noise or motion artifacts). Second, global registration can lead to varying centroid distances and overlap between identified neurons across the field of view. Third, variability in neuron position and/or FOV changes can reduce tracking accuracy across sessions. Finally, some analyses require analysis of neural signals across all sessions, in which case lowering the detection threshold for neuron extraction may allow for the identification of lower signal neurons at the cost of an increased false discovery rate (Video S1). These factors are compounded when experiments take place over extended time periods (>30 sessions). An ideal cell tracking algorithm should therefore be robust to changes in neuron position, false discoveries, and missing neurons. To address these issues, spatial metrics alone are insufficient.

SCOUT (Single-Cell SpatiOtemporal LongitUdinal Tracking), is a method for tracking individual neurons across multiple sessions using both spatial and temporal metrics. SCOUT uses the temporal similarity metrics of SNR (signal-to-noise ratio) and fluorescence decay rate, as well as a new correlation metric which uses connecting recording segments to verify the neuron identification (**Figure 2.2 C**) in addition to standard spatial metrics such as centroid distance, footprint overlap, and Jensen-Shannon divergence, to improve neuron

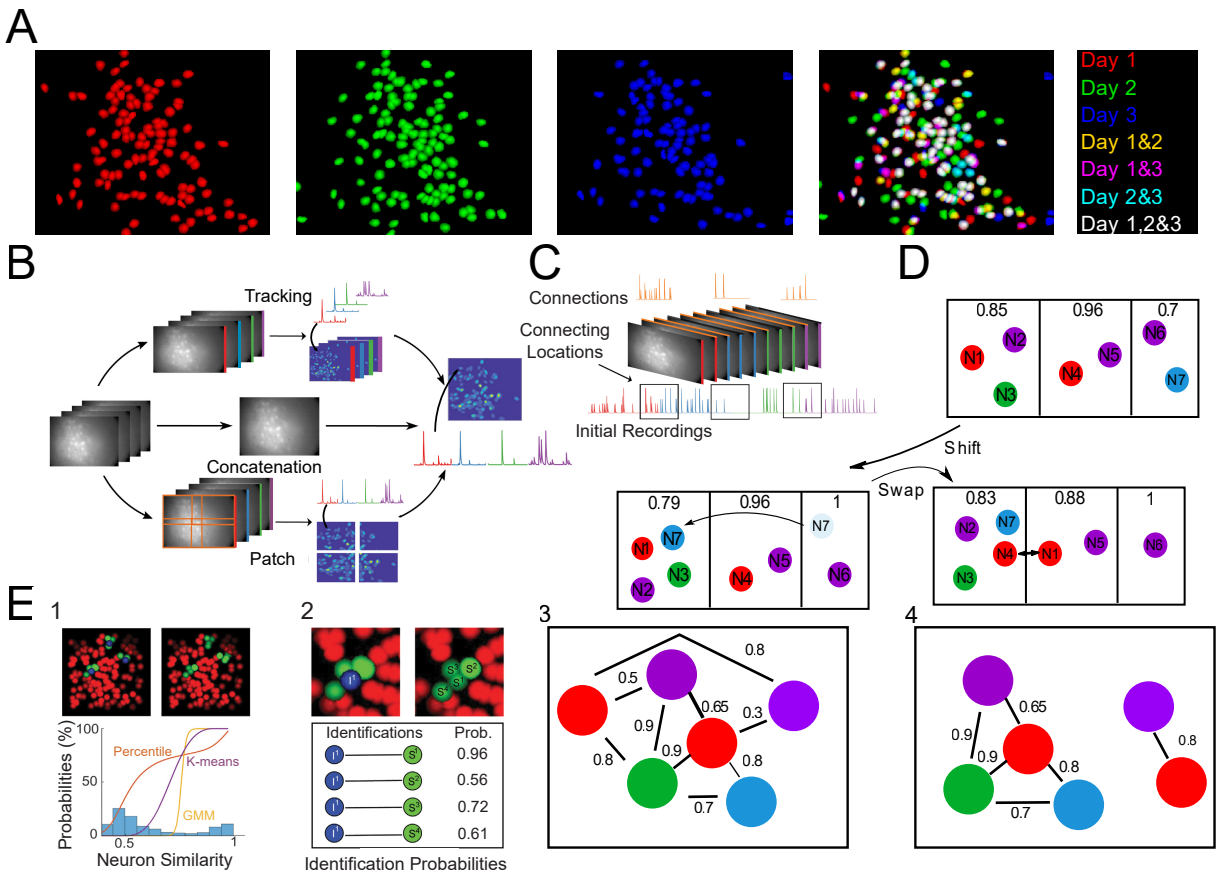

Figure 2.2: SCOUT: a method for single cell tracking incorporates spatial and temporal metrics into a probabilistic consensus clustering framework. **A**: Neuron spatial footprints from three sessions obtained from a 1-photon recording of the prefrontal cortex with neurons colored by session (rectangle 1-3). Overlaid results in the fourth rectangle. Cell tracking seeks to identify the same cell across multiple days. **B**: Long-term study of neural activity requires computation of fluorescence traces for identified neurons across sessions (right) from individual sessions (max projections, left). Approaches include concatenation (middle track), patch methods (bottom track) and tracking methods (top track). Concatenation involves global registration of sessions and concatenation (middle step) followed by fluorescence extraction. Patch methods divide each session into overlapping patches in the spatial domain (orange rectangles, first step), which are concatenated, and fluorescence activity extracted for each neuron (black arrow, second step) followed by merging patches. Tracking methods extract traces for all neurons in each session (first step), followed by identification of neurons across sessions (second step). **C**: Temporal correlation involves a link session (orange) between consecutive pairs of recordings. High quality neurons result in a corresponding neuron in the link session with matching neural signals used to identify cells across sessions. **D**: The SCOUT clustering algorithm groups cells from different sessions into clusters. Boxes indicate separate clusters, with the color of each circle indicating the session. The associated numbers indicate the within cluster similarity for the given cluster. **E**: Demonstration of the SCOUT algorithm. (1) Several cells (blue) and their neighbors (green) within session (top left), and between sessions (top right). Histogram of cell-cell similarity between sessions for a metric (bottom) with overlaid identification likelihood using several models. (2) A single cell (blue), and its neighbors (top) within and between sessions with sample aggregate (across metrics) identification probabilities (bottom). (3) A sample graph in which nodes indicate neurons, and edges between nodes denote identification probability exceeding a minimum threshold (min_ prob). Colors correspond to sessions. (4) A possible graph clustering.

identification between sessions. SCOUT also provides the option of allowing the user to define additional metrics according to their use case. SCOUT then uses a combination of probabilistic models, a novel clustering algorithm, and consensus clustering to perform cell tracking over multiple sessions (**Figure 2.2 D-E**). This combination of features makes SCOUT unique among cell tracking methods.

In total, SCOUT cell tracking consists of four steps: 1) cell-cell identification probability computation for session pairs, 2) creation of cell-cell similarity matrices, 3) clustering of the cell identification matrices and consensus clustering of the resulting cell identifications, 4) creation of a cell register defining indices of identified cells between sessions and associated neural signals. As input, SCOUT accepts the spatial footprints and temporal traces of the neurons in each session (as well as some computed statistics of the temporal traces) and returns a cell register, a num_neurons x num_sessions matrix consisting of all neurons tracked between sessions. Some entries may be empty.

2.5.1 Spatial similarity measures for calculating neuron similarity across sessions

Currently, three methods for spatial similarity are included with SCOUT: centroid distance, spatial overlap, and Jensen-Shannon divergence. Centroids of neuron spatial footprints are calculated using the usual formulae $\overline{x} = \sum_{i,j} x_i a_{ij}$, $\overline{y} = \sum_{i,j} y_j a_{ij}$, where i,j range across the number of pixels in the field of view, in the horizontal and vertical directions respectively, and a_{ij} is the footprint intensity at the ith horizontal pixel, and the jth vertical pixel. Centroid distance between to footprints is calculated as the Euclidean distance between their centroids. Spatial overlap between footprints a, b is calculated as $\frac{a \cdot b}{||a||_2 ||b||_2}$, where a and b, are binarized column vectors representing whether each footprint has positive pixel in-

tensity. Jensen-Shannon divergence between two (normalized) footprints P, Q, is calculated as $\frac{1}{2}(\mathrm{D_{KL}}(P||M) + \mathrm{D_{KL}}(Q||M))$, where $M = \frac{1}{2}(P + Q)$, and $\mathrm{D_{KL}}$ is the Kullback-Liebler divergence: $\mathrm{D_{KL}}(P||Q)$=E(log[dP/dQ]), where dP/dQ is the radon-nikodym derivative of P with respect to Q.

2.5.2 Temporal similarity measures for calculating neuron similarity across sessions

SCOUT has implemented temporal similarities based on signal-to-noise ratio

$$SNR = Var(Signal)/Var(Noise),$$

and the fluorescence trace decay rate for each neuron. Signal decay rate is computed automatically in CNMF-E, by fitting exponential models (among other options) to the neural signal. We have provided a similar algorithm for computing signal decay rate directly from the raw neural signal to ensure SCOUT works with other pipelines. This algorithm identifies peak locations, normalizes the signal height at each peak, and fits an exponential decay function to the average signal. This function can fail if fewer than three peaks were detected for a given neuron.

For SNR, similarity between neurons is calculated as $|(\log(SNR_1) - \log(SNR_2))|$ where SNR_i is the signal-to-noise ratio for neuron i (taking the logarithm produces a more centralized distribution of values). Signal decay similarity is calculated as $|(dec_1 - dec_2)|$, where dec_i is the signal decay rate for neuron i.

2.5.3 Calculation of temporal correlation similarity metric across sessions

Given two preprocessed optical recording sessions S_1 and S_2, we construct a connecting recording S_c by concatenating the last n frames of the first recording with the first n frames of the second, where n is some number less than the minimum number of frames in S_1 and S_2. Next, we extract spatial and fluorescence traces from S_1, S_2, and S_c.

Given N_1, a neuron from S_1, and N_2, a neuron from S_2, we start by setting a maximal distance threshold m, which defines neighboring neurons. If the distance between the centroids N_1 and N_2 exceeds m, N_1 and N_2 are not considered neighbors. Only neighboring neurons can be identified as the same between sessions. We eliminate from our calculations any neuron in the connecting session exhibiting neural activity in frames overlapping only one of the sessions, as such neurons will not allow comparison between sessions.

For temporal correlation similarity, a similarity score is obtained for each neighboring neuron pair (N_1 and N_2) in the two recording sessions, by ranging over the full set of neighboring neurons (N_c) in the connecting recording (i.e. across the set of N_c coming from S_c such that N_1 is a neighbor to N_c, and N_c is a neighbor to N_2). The choice N_c that maximizes the average of the correlation between N_1 and N_c, and N_c and N_2, is considered the connecting neuron, and the correlation similarity between N_1 and N_2 is the mean of the maximal correlation across choice of connecting neuron N_c (**Figure 2.2 C**).

2.5.4 Assigning pairwise identification probabilities with SCOUT

To assign probability scores between sessions for a given metric, we detail two approaches. First, we can simply assign the percentile as the probability score for each metric. If the

distance between N_1 and N_2 for a given a metric, is less than $p\%$ of distances between all possible neighbor pairs, then p is the percentile assigned to the pairing. This method has several drawbacks. First, it is sensitive to the choice of maximum distance parameter. If the parameter governing the maximum distance between neighbors is increased, the probability assigned to any neighboring pair will increase. Second, when few neuron pairs exist, similarity metric values can accumulate near 0, so that even relatively small metric values can be associated to low probabilities.

Another paradigm is to assume that for each metric, the distances between neighboring pairs come from a mixture of distributions: a distribution of distances corresponding the neurons that should be identified between sessions, and a set of neighbors that are distinct [42]. Before fitting the mixture of distributions, a probability density function is constructed, by applying kernel density estimation to the normalized histogram of distances, using reflected boundaries near theoretical maximum and minimum values (such as 0 or 1 for correlation metrics). Next, we construct a model consisting of the weighted sum of two probability distribution functions, which is then fit to the approximated pdf, using nonlinear regression (Matlab nlinfit, [43]).

Given a mixture model consisting of a weight w, a model for identified neurons between sessions, f, and a model for unidentified neurons between sessions, g, the mixture model approximates the probability distribution function h, obtained via kernel density estimation from the initial distribution of distances, as $h(x) = wf(x) + (1-w)g(x)$. Given a proposed distance x, the probability that x is sampled from the distribution with pdf f, is given by $\frac{wf(x)}{wf(x)+(1-w)g(x)}$, using Bayes theorem. We have primarily used Gaussian mixture models.

We can also apply soft K-means clustering [44], an adaptation of K-means in which data points are assigned identification probabilities for each cluster, and a "fuzzifier" is introduced to govern the spread of identifications probabilities, adjusting the crispness of the clusters. Similarly to mixture models, this algorithm separates similarities into identified and non-

identified categories, with associated probabilities. This algorithm frequently identified the most neurons, but with a higher false discovery rate. This is the default algorithm for SCOUT.

Generally, little difference is seen between results with soft K-means and mixture models, except for recordings with only two sessions, for which Gaussian mixture models typically exhibit stronger results. GMM distribution fitting is typically slower than K-means and produces a sharper decision boundary. We recommend using K-means except in the case where the recording consists of only two sessions.

2.5.5 Clustering algorithm

After computation of temporal metric similarity for all neuron pairs in each pair of sessions, we assign identification probabilities for each metric using a probabilistic model (i.e., soft K-means, GMM). To combine the metric identification probabilities into a single identification probability, we use a weight vector (a 1 x n vector where n is the number of metrics, such that the sum of entries is 1) which governs the emphasis each metric receives. Multiplying this weight vector by the associated identification probabilities for each vector and summing the result results in a single identification probability for each pair of cells in each pair of sessions. These probabilities are placed in a similarity matrix of size n_cells x n_cells, where n_cells is the total number of neurons extracted across all sessions and entries are the identification probabilities between cell pairs. Cells in the same session are assigned a low similarity (-10,000) to prevent clusters from containing more than one neuron from the same session.

This similarity matrix is decomposed into connected components, and each component is clustered according to the following algorithm. Clusters are initialized by placing the least

similar neurons in separate clusters. Remaining neurons are added to each cluster based on which addition decreases average cluster inter-neuron similarity the least. If no option is available that keeps the average cluster similarity above the user provided *chain_prob* threshold, a new cluster is created. The process continues until all neurons are assigned a cluster. Similarities between neurons in the same session are set to some high magnitude negative number, to prevent assignment to the same cluster.

At each iteration, we calculate the total increase in average similarity between cluster members (over all clusters) gained by assigning each neuron to a new cluster (a switch) (**Figure 2.2 D-E**). We also calculate the total increase in average similarity between cluster members derived by swapping the cluster assignments of each pair of neurons (a swap). The operation that maximizes the similarity increase is chosen, and the algorithm continues until a maximum number of iterations is reached, or no further gains can be made by these operations. Finally, the minimal average cluster similarity (across clusters) is compared with the *chain_prob* threshold. If the cluster similarity falls beneath this value, a new cluster is created and populated with the least similar neuron in the lowest scoring cluster (ranked by average within cluster similarity), and the process repeats until convergence.

To correct the propensity toward the creation of medium sized clusters (and thus against tracking neurons through all sessions), we add a bias term to the switch and swap scores as follows. First, a switch that increases the maximal cluster size of the clusters involved is rewarded with the addition of a constant bias term, while the reverse is penalized by the subtraction of the bias term. A swap is penalized via subtraction of a bias term if the swap causes the cluster with the larger size to decrease its average identification probability, and vice versa if the cluster with the larger size were to increase the average identification probability. Swaps and shifts that decrease the size of a cluster with inter-neuron similarity exceeding the *chain_prob* threshold are ignored. By placing a limit on the number of times any individual neuron can be swapped or shifted between groups, the algorithm converges

in most instances, and usually within a few iterations (< 25).

Discrepancies between clustering results due to initial clustering assignments, as well as the problematic usage of a single weight variable for aggregating identification probabilities motivates a consensus clustering framework. We generate random perturbations of the weight vector by adding a random value produced by from an N(0,0.12) distribution (0.12 is $\frac{3}{4}$ the individual weight if all metrics are used). Weight values below zero are set to zero, and the vector is renormalized. In the implementation, 29 perturbations are generated (resulting in 30 vectors), which are then used to create similarity matrices which are clustered via the previously described algorithm, but using the connected components defined by the initial weight vector.

Next, we construct a consensus matrix for each component of size n_cells x n_cells, where n_cells is the number of cells in the component, and the entry is the percentage of instances in which the associated cell pair were placed in the same cluster. This consensus matrix is then clustered using the same algorithm described above to produce the final cell register.

Chapter 3

Testing SCOUT on simulated and *in vivo* data

We now test both the spatial filter and tracking aspects of SCOUT on both simulated and *in vivo* data. We begin by detailing the experimental data, after which we present the associated results. Finally, we test our cell tracking algorithm on hippocampal data and use place field consistency as an additional verification of cell tracking quality.

3.1 Simulated Data

For all simulated datasets, neuron footprints were simulated as 2−dimensional Gaussian probability distributions, with diagonal covariance matrices. Spatial footprint width was between 20 and 25 pixels. Spikes were simulated from a Bernoulli distribution with probability of spiking per timebin 0.01, and then convolved with a temporal kernel $g(t) = exp(-t/\tau_d) - exp(-t/\tau_r)$, with fall time $\tau_d = 6$ timebins, and rise time $\tau_r = 1$ timebins. Local background

spatial footprints were simulated as 2-D Gaussians, but with larger covariance entries than for the neuron spatial footprint. Blood vessel spatial footprints were simulated using a cubic function, which was convolved with a 2-D Gaussian (Gaussian width: 3 pixels). A random walk model was used to simulate temporal fluctuations of local background and blood vessels. 23 background sources were used throughout all simulated experiments, except for the Individual Shift dataset, in which no background sources were present.

Four sets of recordings were simulated for testing purposes (**Figure, 3.1 A-D**). The Gaussian dataset consisted of 11 recordings with 2000-10000 frames each, with a 256 x 256-pixel FOV. Each recording was simulated using 50-200 neurons. The Non-Rigid 1-photon dataset consisted of 39 footprint recordings consisting of four sessions of 2000 frames each. Each simulated spatial footprint was transformed with a different individual non-rigid transformation in each session. This transformation was primarily in place, with little translational effect ($<$ 2 pixels). The Non-Rigid 2-photon dataset consisted of a copy of the Non-Rigid 1p dataset, in which Gaussian noise was replaced with salt and pepper noise, to portray 2-photon conditions more closely. Spatial footprints were converted into characteristic ring shapes via the following transformation. Pixel intensity values were scaled to lie in the [0,1] interval. All pixels with intensity higher than 0.5 were then replaced by the same intensity subtracted from one. The Individual Shift dataset consisted of 29 recordings consisting of two 3000 frame sessions, with a 100 x 100-pixel FOV. Each recording was simulated using 50-100 neurons. The individual spatial footprints were shifted independently by between 5 and 7 pixels (˜30-40% neuron width) in the second session.

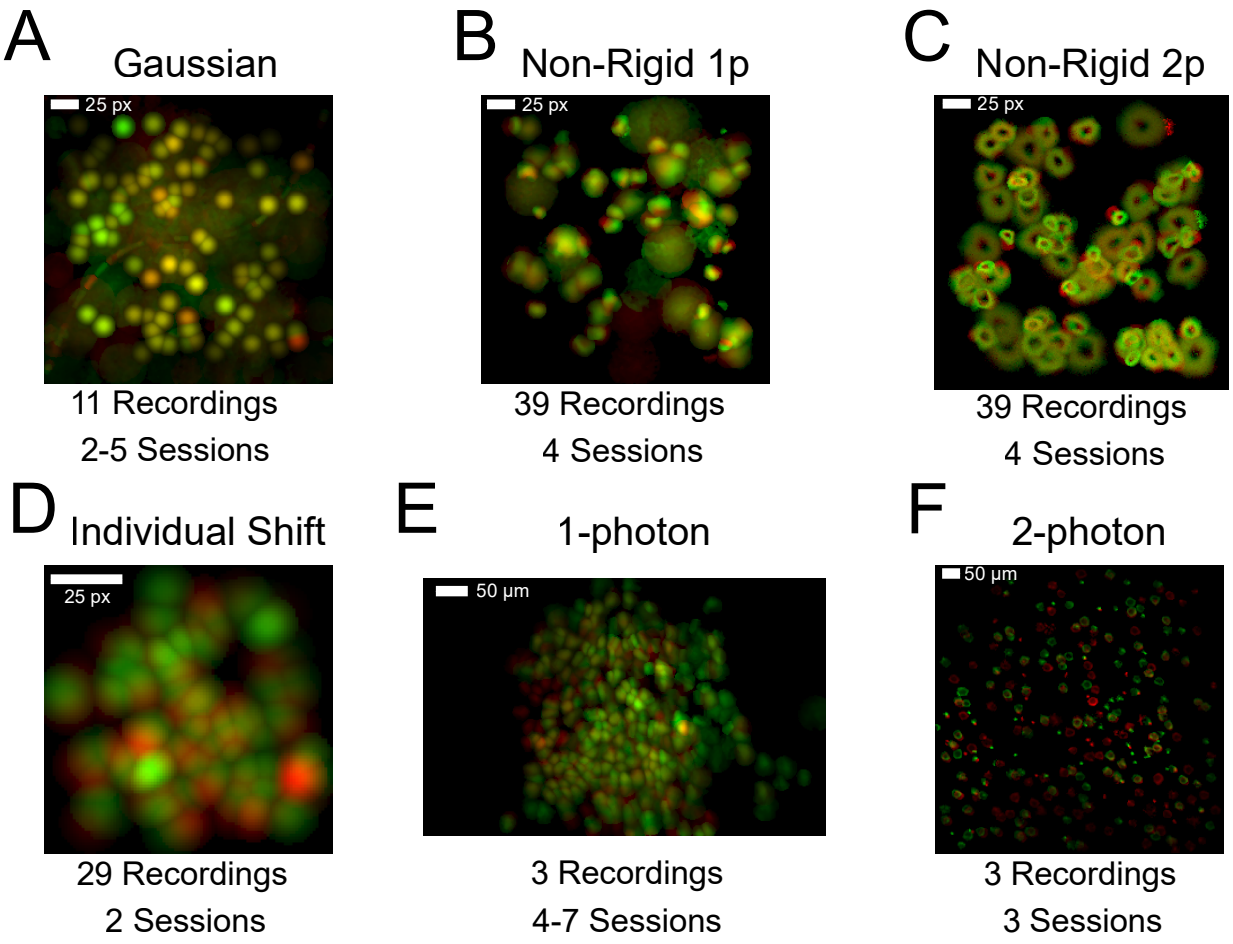

Figure 3.1: Overview of tested data. Neuron extraction from two sessions from each dataset, overlaid to visualize the identification of neurons across each session, as well as the number of recordings and sessions per recording in each dataset. Gaussian (**A**), Non-Rigid 1p (**B**), Non-Rigid 2p (**C**), and Individual Shift (**D**) datasets are simulated, while 1-photon (**E**) and 2-photon (**F**) datasets are in vivo recordings. Simulated datasets exhibit different features and difficulties for cell identification across sessions, such as noise (particularly in the Non-Rigid 1p dataset), cell body transformations (Non-Rigid 1p and Non-rigid 2p) and neuron translations (Individual Shift) represent common issues with in vivo recordings. Similar effects can be viewed in the in vivo recording representations.

3.2 *In vivo* recordings

1-photon recordings: surgical preparation

Mice were housed under a controlled environment with temperature maintained at 21 - 23 °C and humidity at 40% - 70%. Mice had free access to water and diet except water restriction during linear track test. The age was 8-10 months at the time of test, both sexes

were included since we didn't observe difference of behavior and calcium activity between male and female mice. All the experimental protocols were approved by the IACUC of the University of California, Irvine.

Surgery was performed as described previously [28]. Briefly, AAV1-CaMKIIa-GCaMP6f-WPRE-SV40 was purchased from Addgene. Mice were anesthetized with 1.5% - 2% isoflurane and placed on a stereotaxtic instrument (Stoelting). Virus was injected into dorsal CA1 (AP -1.94, L 1.4, DV -1.38 mm, relative to bregma) of the right hemisphere using a glass micropipette. The diameter of the pipette tip was 20 - 30 um. The virus titer was 1 x 10^{13} GC/ml and injection volume was 0.3 µl. Mice were treated with carprofen (3 mg/kg) as analgesia for 3 days after surgery.

A GRIN lens was implanted two weeks after virus injection. Mice were anesthetized with isoflurane, and carprofen and dexamethasone (2 mg/kg) were administered. A 2-mm-diameter cranial window was drilled over recording FOV. Then dura was removed with ultrafine forceps, and cortical tissue above the target area was carefully aspirated using a 29-G blunt needle connected to vacuum, until the vertical striations of corpus callosum appeared. Sterile saline was applied during aspirating. After bleeding was completely ceased, a GRIN lens (1.8 mm diameter, 4.3 mm length, 0.25 PITCH, Edmund Optics) was lowered to contact the corpus callosum (depth -1.55 mm) and secured with superglue and dental cement. The skull and lens were covered with Kwik-Sil silicone elastomer (WPI), and mice were allowed to recover for 2-3 weeks.

Mice were anesthetized again, Kwik-Sil was removed and a miniscope (UCLA) mounted onto a baseplate was placed on the GRIN lens to search the imaging area. After cells being in focus, the baseplate was attached on the skull with dental cement, miniscope was removed and a plastic cap was placed on the baseplate to prevent dust.

1-photon recordings from Visual Cortex, Prefrontal Cortex, and Hippocampus

Separate individual recordings were taken from the visual cortex, prefrontal cortex, and hippocampus (**Figure, 3.1 D**). The visual cortex recording was subject to the following protocol (Grieco et al., 2020). An initial pair of baseline sessions were taken on consecutive days. After baseline collection on the second day, the animal received a ketamine wash treatment followed by a second recording session. Day 3 consisted of a baseline recording session, followed by NRG1 wash and a second recording session. A single recording session was taken on days 4 and 5. The 7 individual sessions consisted of 4,000-9,000 frames. All tested animals were male.

The prefrontal cortex recording was subject to the following protocol [45]. An initial pair of baseline recordings were taken on consecutive days. After baseline collection on the second day, the animal received a ketamine wash treatment followed by a second recording session. Further recordings were taken 2hr, 24hr, 48hr and 72hr after ketamine treatment. The 7 individual sessions consisted of 4,000-9,000 frames.

The hippocampus recording was subject to the following protocol [28]. Two baseline control recordings taken on consecutive days, were followed on the following day by treatment with CNO (clozapine-N-oxide) and a third session. A post control session was taken after 4 days. Sessions consisted of 7,000-9,000 frames.

Recordings were obtained from and used with permission of the Xu lab (https://sites.uci.edu/xulab/). Recordings previously published in [28, 45, 46].

1-photon hippocampal recordings from contextual discrimination experiment

We randomly selected three sessions from three separate mice undergoing a long term contextual fear discrimination experiment for spatial filter testing. Mice were surgically treated as in the previous protocol. Experimental conditions did not appear to contribute unusual variations to spatial filter quality.

1-photon hippocampus recordings for place field stability

The hippocampal region of the brain exhibits so called "place cells," neurons which are critical for the formation of spatial memory and the encoding of spatial position. Individually, these cells encode spatial information in a very simple way. Each place cell has a corresponding "place field," an environmental region (typically on the order of centimeters) in which the neuron preferentially fires, analogous to the receptive fields in the visual cortex. The information encoded in these place fields is then aggregated into a spatial map than enables encoding of location in memory. Critically, these place cells are highly consistent across time, maintaining stability over months or longer. This means these cells can be used as an objective measure for cell tracking quality.

A total of 3 mice were water restricted until their bodyweight reached 85% - 90% of the initial weight, then they were trained to run back and forth on a 1-meter-long linear track to obtain 10-20 µl of water reward on either end of the track. After 5 days of training, miniscope was tethered and mice were trained for another 5 days. The testing consisted of two trials at 30 min apart each day and was repeated for 3 days. On the first day, the linear track was placed in the initial orientation of training. On the second day, the track remained the initial orientation in the first trial and rotated for 90 degrees in the second trial. On the third day, the orientation was 90-degree rotated in the first trial and back to the initial

orientation in the second trial. Mice were tested for 30 laps in each trial, usually finishing in 10-15 min. Calcium activity of CA1 neurons was recorded by miniscope, and mouse behavior was recorded by a Logitech webcam simultaneously. The linear track was cleaned with 70% ethanol before each recording. Place field consistency analysis was computed on recordings taken only with the initial orientation, resulting in four sessions.

Information Score Computation and Place Cell Identification

The method for identifying place cells is based on a previous study (Sun et al 2019). For each neuron, peak activity locations of neuron activity (neuron.S) were identified, and associated peak intensities were computed. As CNMF-E can produce a significant number of outlying peak intensity values, low outliers were removed by setting a threshold of (0.5)*median(peak intensity), and peak intensities exceeding a threshold equal to three median absolute deviations from the median peak intensity (the MATLAB default) were set to this upper bound. Finally, the output was smoothed using a Gaussian kernel (width 0.5).

Next, we identified time points where the mouse was within 10% of the distance from either end of the linear track, or when mouse speed was lower than an estimated movement threshold, and the corresponding neural signal was removed from the analysis. Finally, we divided the interior region (i.e. the middle 80% of the track) into 20 bins (horizontal axis only) and computed the ratio of the neural signal measured in each bin to the amount of time the mouse spends in each bin to produce the place field.

For each neuron, the mutual information between position and neural activity (information score) was computed as $\sum p_i \lambda_i log2(\lambda_i)$, where p_i denotes the probability of the animal being in each bin, and λ_i denotes the ratio of the probability of firing while in the bin to the mean probability of firing. The sum is taken across all bins.

Information percentiles were computed for each neuron by recomputing the information score after randomly shifting the position by at least 3 seconds a total of 500 times. The information percentile was then assigned as the percentile of the non-translated information score in the distribution of translated information scores.

2-photon recordings

2-photon recordings were obtained from the Allen Brain Observatory AWS archive ([47],(**Figure, 3.1 F**). Recordings consisted of three sessions. Recordings were temporally downsampled by a factor of 2, and the first 5,000 frames of each session were used for testing.

Experiments consisted of showing head fixed mice separate visual protocols on three separate days. Protocols consisted of drifting gratings, static gratings, sparse noise, natural scenes and natural movies. See Allen brain observatory documentation for additional information (https://help.brain-map.org/display/observatory/Documentation).

Allen Brain Atlas access information. VISl: Experiment ID 564425775, Genotype: Emx1-IRES-Cre/wt; Camk2a-tTA/wt; Ai93(TITL-GCaMP6f)/wt. VISrl: Experiment ID 660510591, Genotype: Cux2-CreERT2/wt; Camk2a-tTA/wt; Ai93(TITL-GCaMPf)/wt VISp: Experiment ID 642651896, Genotype: Rorb-IRES2-Cre/wt; Camk2a-tTA/wt; Ai93(TITL-GCaMP6f)/wt.

3.3 SCOUT spatial filter

3.3.1 *SCOUT spatial filter on simulated data*

We tested the individual recording extraction portion of SCOUT on the Gaussian and Non-Rigid 1p datasets (the Non-Rigid 2p dataset exhibited high extraction quality with both methods, and the individual shift dataset has spatial profiles inconsistent with typical assumptions, due to significant spatial movement between sessions). We examined two extraction conditions for each dataset, one with restricted initializations, in which the threshold for neuron initialization was set sufficiently high to exclude most spurious initialization points (which can exist either due to random fluctuations in background noise, or incomplete subtraction of discovered neurons during the initialization procedure), and the other in which thresholds for neuron initialization were low. Neuron initialization is primarily based on local correlation and signal intensity (**Figure 2.1 A-C**). Extractions with unrestricted initializations had many false discoveries, which allows us to demonstrate the robustness of SCOUT. We term these extractions as restricted and unrestricted through the remainder of this section.

We performed 6 extractions on the Gaussian and Non-Rigid 1p dataset, for each initialization condition (restricted, unrestricted), 5 extractions in which we varied JS divergence thresholds for the spatial constraints (JS constraint values [0.03, 0.06, 0.09, 0.12, 0.15]), and a CNMF-E extraction. We found little significant increase in the number of detected neurons for thresholds exceeding 0.09, and a sharp drop off in detected neurons for thresholds lower than 0.06, so we report statistics from the extractions with these two parameters (see **Figure 3.2 A-D**).

Larger thresholds may be required on *in vivo* data, particularly in cell tracking applications, as false positives have a smaller effect on the result. Results are reported in the form mean

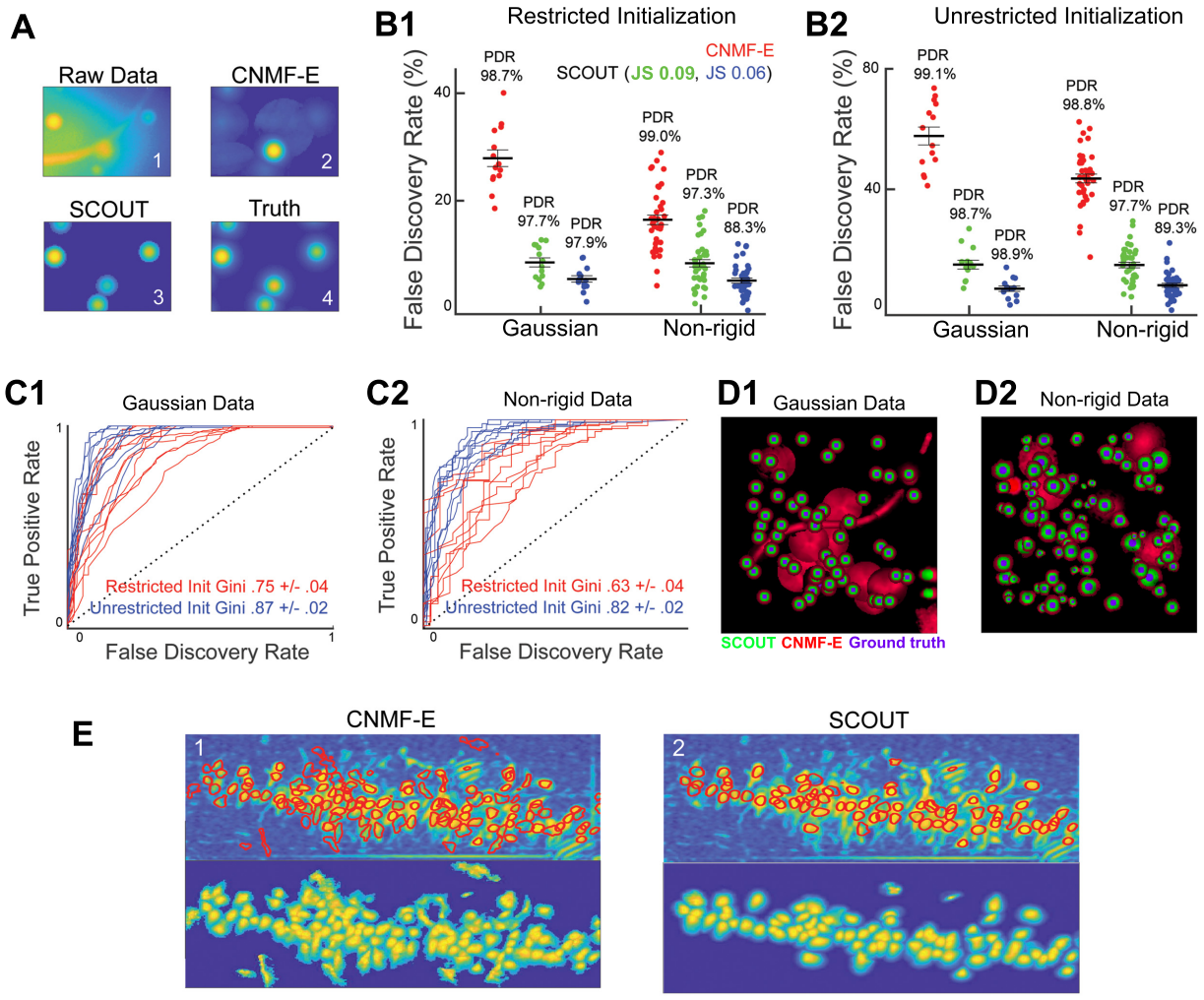

Figure 3.2: A cell-shape based spatial constraint improves extraction performance by controlling false discovery detection and improving neural extraction quality. **A**: Spatial footprint extraction quality shows improvement using SCOUT over CNMF-E on a simulated video recording. (1) maximum projection of a 70 pixel x 100 pixel section of a recording, (2) spatial footprint extraction results of CNMF-E applied to recording, showing multiple false discoveries (3) spatial footprint extraction results of SCOUT applied to recording, retaining all ground truth neurons with no false discoveries, (4) ground truth spatial footprints. **B**: False discovery rates across the Gaussian and Non-rigid datasets, displaying CNMF-E and SCOUT extraction results. (1) The results for extractions with restricted initialization, (2) results for unrestricted initialization. PDR (percent detected rates) are shown above. **C**: ROC curves at various spatial thresholds and average GINI indices (defined as 2 x AUC -1) as a qualitative measurement of classifier efficiency. (1) The results on a set of 10 random recordings from the Gaussian dataset, (2) a set of 10 recordings from the Non-rigid dataset. **D**: Extracted spatial footprints from sample recordings in the Gaussian (1) and Non-rigid (2) datasets demonstrate the difference in false discoveries using SCOUT over CNMF-E. Neuron footprints are normalized to have maximal unit intensity for comparison purposes. Neurons are colored by which methods detected each extracted neuron, with SCOUT: green, CNMF-E: red, Ground Truth: Blue. Detected events include true neurons, blood vessels, and background sources. **E**: Extracted footprints from recordings of CA1 layer of the mouse hippocampus, conducted using CNMF-E (1), and SCOUT (2). (top): The correlation image of the recording, with circled neurons corresponding to those detected by the extraction algorithm. (bottom): The extracted spatial footprints detected by the algorithm, normalized to have the same maximum pixel intensity. Spatial footprints extracted via SCOUT were smoothed during the final spatial template application.

±standard error. Statistical tests were two sided, and Welch's paired t-test and one-way ANOVA was used where stated.

Pearson correlation was calculated between extracted fluorescence traces and ground truth neurons. An extracted neuron was labeled as a false discovery if the maximum correlation value between its fluorescence trace and any ground truth neuron was smaller than 0.8. A ground truth neuron was counted as detected if the maximum correlation between its trace and that of an extracted neuron was at least 0.8. Statistical results were similar for higher thresholds.

For each dataset and initialization condition, we computed the false discovery rate (FDR), defined as the percentage of false discoveries out of the detected neurons, and percent detected rate (PDR), defined as the percentage of ground truth neurons detected in a simulated recording for the extractions given by CNMF-E, SCOUT (JS: 0.06), and SCOUT (JS: 0.09). Statistics were calculated based on PDR and FDR calculated on each recording in each dataset. Results are reported in the form mean ±standard error. Statistical tests were two sided, and Welch's paired t-test and one-way ANOVA were used where stated.

Applying one-way ANOVA to the results from each dataset and initialization condition separately, we identified significant difference of average FDR using CNMF-E, and SCOUT with constraints 0.06 and 0.09 ($p < 7.4 \text{ x } 10^{-18}$, taken over all datasets and initialization conditions). Pairwise comparisons between CNMF-E and SCOUT showed SCOUT detected fewer false discoveries on average at both constraint levels across all datasets and conditions (t-test $p < 1.3 \text{ x } 10^{-7}$). Average FDR exceeded 40% on the unrestricted extractions, and 16% on restricted extractions, across datasets. SCOUT reduced the number of false discoveries by at least half, and up to 85%, while retaining the percent detected rate within 1-2 percentage points of CNMF-E. Total reported average PDR generally exceeded 97%, depending on the dataset, initialization, and extraction method. As expected, more false negatives were assigned by SCOUT in the Non-rigid dataset, requiring a higher JS threshold (0.09) to retain

at least 97% of true neurons (see **Figure 3.2 B**).

To further investigate the efficacy of SCOUT as a classifier, we considered 10 recordings randomly taken from each dataset and condition, and constructed receiver operating characteristic curves, by applying the spatial constraint after neuron initialization (Fig. 2C). The resulting average GINI coefficients, defined as 2 x AUC - 1 (where AUC represents area under the ROC curve), were 0.75 $\pm$ 0.04 and 0.63 $\pm$ 0.04 for Gaussian and Non-rigid data sets, respectively, with the restricted extraction. Average GINI coefficients with the unrestricted extraction were higher, with the respective coefficients being 0.87 $\pm$ 0.02 and 0.82 $\pm$ 0.02 for Gaussian and Non-rigid data sets. These quantitative metrics demonstrate that SCOUT is a robust classifier across different datasets (**Figure 3.2 C**).

3.3.2 SCOUT spatial filter on *in vivo* data

After verifying the effectiveness of SCOUT on simulated data, we continue to examine the effects of introducing spatial constraints to neuron extraction in experimental *in vivo* recordings from hippocampal CA1, obtained from three separate mice from the contextual discrimination experiment. Each recording consisted of approximately 7000 frames each and was extracted using both CNMF-E and SCOUT. While ground truth data was not available for these recordings, we developed a set of three criteria for classifying neurons as true discoveries. First, the spatial footprint was examined visually. Neurons with abnormal footprints were removed from consideration. Next, the fluorescence trace corresponding to each neuron was examined for irregularities, such as traces with non-zero baselines, or traces that exhibited localized activity that may be attributable to recording noise, or poor extraction quality. Finally, the remaining neurons were plotted on the correlation image (which shows local correlation between neighboring pixels, **Figure 3.2 E** (1,2) top subpanels), and neurons that appear to encompass spatially distinct regions of the correlation image were removed as false

discoveries. Across all three recordings, an average of 24.3 ± 2.3% of neurons discovered by CNMF-E, were classified as false discoveries, while 9.0 ±2.1% of neurons discovered by SCOUT were classified as false discoveries (**Figure 3.2 E**).

We also tested SCOUT on three additional recordings from the visual cortex, the hippocampal CA1, and the prefrontal cortex, consisting 4000-8000 frames each. CNMF-E detected an average of 15% more neurons than SCOUT after false discovery removal. The number of detected true discoveries was, on average, larger with SCOUT than with CNMF-E. In these additional datasets, we found an average of 24.4 ± 7.9% of neurons detected by CNMF-E were classified as false discoveries, while 8.1 ± 0.7% of neurons detected by SCOUT were classified as false discoveries.

3.4 Efficacy of temporal metrics for cell tracking

We first show that the temporal metrics SNR, decay rate and temporal correlation are consistent across and within sessions and provide additional discriminatory features useful for identifying neurons across sessions (**Figure 3.3 A-C**). To demonstrate within session consistency, we split the first session of *in vivo* recording sessions in half longitudinally and compute SNR and decay rate metrics on the resulting sessions. For each neuron in the first half, we compute the absolute difference of the decay rate and SNR for the same neuron in the second half, and the nearest non-identified neuron in the second half as comparison (**Figure 3.3 D-E**).

Ratios of the median difference between the same and nearest neighbor neurons are significantly lower than 1.0 (p = 2.9 x 10^{-33} ,1.5 x 10^{-4}, 1.2 x 10^{-10}, VC, PFC, Hipp., Wilcoxon rank sum test), as are median SNR difference ratios (p = 2.3 x 10^{-24}, 6.9 x10^{-4}, 4.8 x 10^{-3}). Both SNR and fluorescence decay exhibit slightly higher ratios on 2-photon data, though

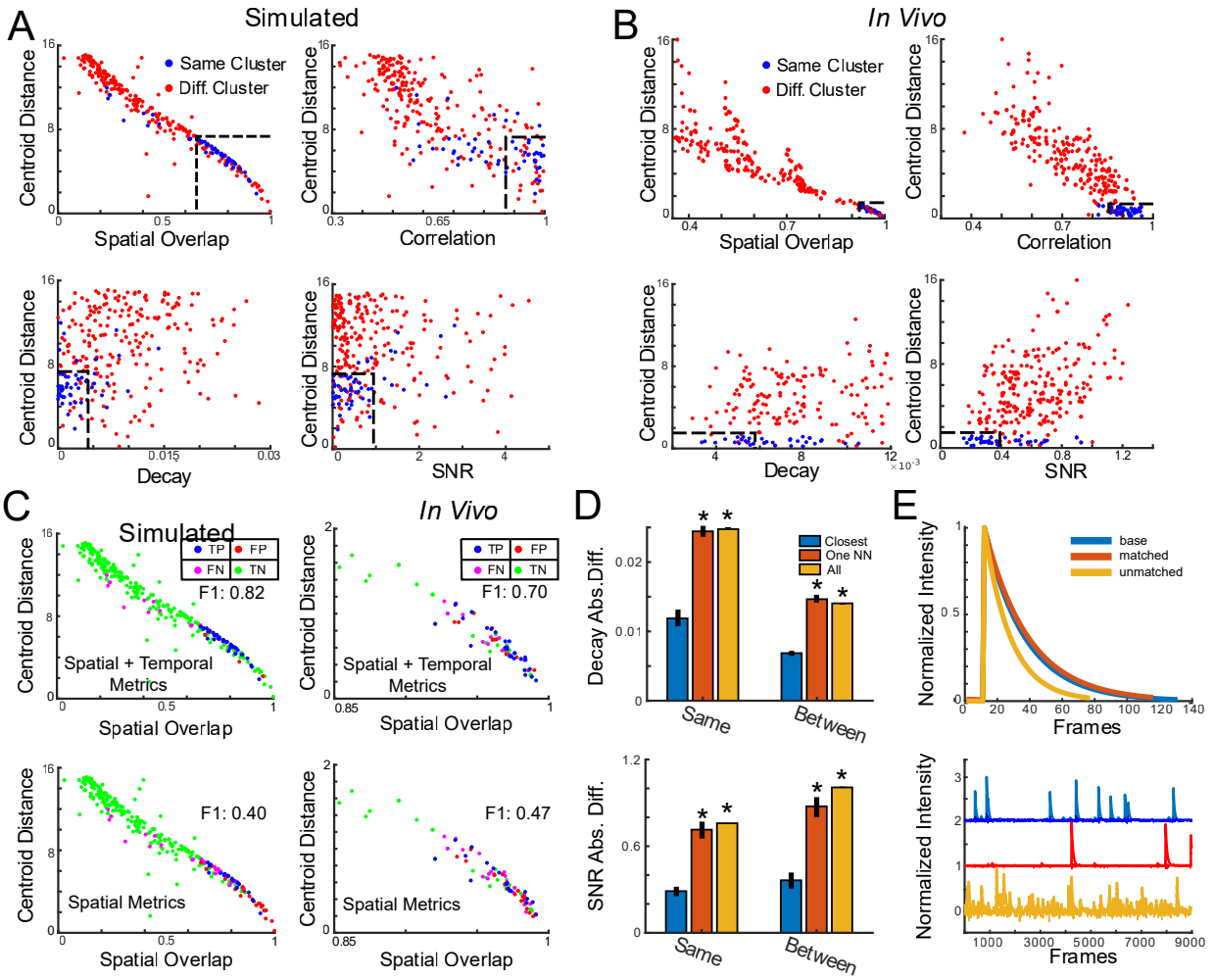

Figure 3.3: Inclusion of temporal metrics is a key step when there is significant shift between neurons across sessions. **A**: Scatter plots compare spatial (overlap) and temporal (SNR, decay, correlation) metrics (x-axis) to the centroid distance metric (y-axis) for all neuron pairs (identified: blue, non-identified: red, labeled by ground truth) from two sessions of a recording taken from the Individual Shift dataset. Approximate decision boundaries for each metric are indicated by the black dashed line. **B**: Scatter plots compare inter-cluster metric similarity on a 1-photon in vivo recording consisting of 7 sessions. Correct identifications based on human annotated ground truth cell register. Incorrect identifications are simulated by randomly exchanging a neuron in a ground truth cluster with a near neighbor (as measured by centroid distance). Plotted points indicate average similarity (by metric) for both correct (blue) and incorrect (red, at least one error) clustering results. Approximate decision boundaries for each metric indicated by the black dashed line. **C**: Similarity metrics (as shown in **B**, **C**) are aggregated with resulting cell similarities used for clustering. False positives (FP), true positives (TP), false negatives (FN), and true negatives (TN) are labeled in red, blue, purple, and green respectively. Results using all metrics (top) and exclusively spatial metrics (bottom), presented for the simulated (left) and in vivo (right) recordings. **D**: Bar charts compare SNR and Decay metrics between identified neurons, nearest neighbors, and average similarity across all neurons, within and between two sessions of an in vivo 1p recording. (Top) SNR absolute differences (y-axis) after splitting the first recording into two sessions (same), and between sessions (between) for identified neurons (closest), nearest neighbors (One NN), and all neuron pairs (All). (Bottom) Absolute signal decay rate differences within and between sessions. The reduction in value between sessions is due to the use of post extraction computation decay for the within session data. Error bars indicate standard error across neurons. **E**: (Top) Neural traces associated to a single spike from 3 neurons taken from two sessions of an in vivo 1-photon recording: (blue) a baseline neuron from the first session, (red) an identified neuron from the second session, (yellow) a non-identified neuron from the second session. (Bottom) Neural traces from the baseline, identified, and non-identified neurons along with the noise level after normalization to unit peak intensity.

this effect is primarily due to a single recording (decay: p = 0.02, 0.04, 0.04, SNR: p = 1.4 x 10^{-7}, 4.7 x 10^{-2}, 7.4 x10^{-10}, VISl, VISrl, VISp). Ratios ranged from 0.5-0.65 (i.e. 50-65% of the difference of non-identified nearest neighbors), indicating much lower differences between SNR and decay rate of the same neuron compared with its neighbors.

Next, we compute similarity of temporal metrics (including correlation metric) between neurons in the first session (unsplit) and the second session of *in vivo* recordings, to demonstrate that temporal metrics discriminate between identified and non-identified neurons between sessions. This context adds complications, as some neurons may not correspond to identified pairs in the other session, and the ground truth is unknown. We assume that the most similar (based on the current temporal metric) neurons with overlap exceeding 0.9 are identified, which are compared with the nearest neighbor below this threshold. Neurons in the first session with no other spatial footprints within 4 pixels in the second session are excluded from the analysis.

On 1-photon data, median absolute difference in decay rates is significantly lower (p = 4.1 x 10^{-9}, 8.8 x 10^{-5}, 2.4 x 10^{-5}, VC, PFC, Hipp., Wilcoxon rank sum test), as are differences in SNR (p = 6.3 x 10^{-8}, 2.5 x 10^{-4}, 3.1 x 10^{-6}), with similar results on 2-photon recordings (p = 0.006, 0.006, 6.3 x 10^{-9} , VISl, VISrl, VISp). One result did not show significant SNR differentiation (p = 0.98, 0.03, 0.01). Average correlation metric values on both 1-photon data (p = 2.2 x10^{-22}, 2.2 x 10^{-8}, 2.8 x 10^{-9}) and 2-photon data (p = 7.2 x 10^{-5}, 1.1 x 10^{-3}, 6.8 x 10^{-12}) show higher median temporal correlation for identified vs non-identified neurons.

Ratios of median temporal metric difference are higher between sessions (i.e. more difference between sessions than within sessions), with differences between 60-75% those of nearest neighbors for SNR and decay, and with correlation ˜1.5x higher between identified neurons and nearest neighbors. This implies that the median pair of identified neurons have significantly more similar temporal profiles than nearest neighbors which motivates the inclusion of temporal metrics in SCOUT.

3.5 Testing SCOUT on simulated multi-session recordings

On simulated recordings we first consider only neurons identified through all sessions as these are the most easily interpreted in downstream analysis. Cell tracking quality is defined using the F1 metric $\frac{2PDR(1-FDR)}{(1-FDR+PDR)}$, which takes values between 0 and 1 (1 being the highest quality). Here PDR (percent discovery rate), is defined as the percentage of available neurons tracked by a method, and FDR (false discovery rate), is defined as the percentage of tracked neurons containing at least one false identification. Next, we consider tracking quality for all sets of identified neurons using the Jaccard similarity metric (computed as $|A \cap B| \; / \vee A \cup B \vee ?$, where A and B represent sets of identified neurons). This analysis includes all clusters of identified neurons. For testing purposes, we use neuron footprint centroid distance, spatial overlap, and Jensen-Shannon divergence (spatial metrics), and SNR, fluorescence decay rate, and correlation (temporal metrics).

We test SCOUT (with and without temporal metrics), cellReg and CaImAn multiple times on each dataset with varying parameters (*max_dist*, *min_prob*, *chain_prob* for SCOUT, see **Figure A.1 and A.2** for cellReg and CaImAn parameters). Here we present results from parameters maximizing the F1 score across each tested method. Statistical results are computed using ANOVA and post hoc Bonferroni correction (Bonferroni 1936) for multiple comparisons. Quantitative results are presented as mean +/- standard error, where each datapoint corresponds to a single recording in the dataset.

We identify significant F1 score differences on the Gaussian, Non-Rigid 1p and Individual Shift datasets (Gaussian: p = 3.06 x 10^{-12}, F = 40.77; Non-Rigid 1p: p = 1.87 x 10^{-16}, F = 32.92; Individual Shift: p = 3.35 x 10^{-19}, F = 45.14; ANOVA). Pairwise comparisons show a statistically higher average F1 score between SCOUT and CaImAn on the Gaussian

dataset (**Figure 3.4 A, Figure A.1 A-C**), a higher average F1 score between SCOUT and both cellReg and CaImAn on the Non-Rigid 1p dataset (**Figure 3.4 B, Figure A.1 D-F**), no significant differences on the Non-Rigid 2p dataset (**Figure 3.4 C, Figure A.2 A-C**), and a higher F1 score for SCOUT than both cellReg and CaImAn on the Individual Shift dataset (**Figure 3.4 D, Figure A.2 D-F**), (CaImAn comparisons $p < 1.16$ x 10^{-11}, cellReg comparisons $p < 2.6$ x 10^{-17}). Additionally, SCOUT with temporal metrics outperforms SCOUT without temporal metrics on both the Non-Rigid 1p dataset, and the Individual Shift dataset ($p < 3.5$ x10^{-9}).

In summary, SCOUT exhibits high quality cell tracking performance when compared with methods such as cellReg and CaImAn, particularly on the Non-Rigid 1p and Individual Shift datasets (**Figure 3.5 , Figure A.1 and A.2**). Comparisons of SCOUT with and without temporal metrics show that inclusion of temporal metrics in the analysis results in higher average F1 scores.

We next compute Jaccard similarity on the same data. This method identifies significant differences in mean tracking quality on all datasets except for the Non-Rigid 2p dataset (Gaussian: $p = 2.1$ x 10^{-15}, F = 127.8; Non-Rigid 1p: $p = 1.64$ x10^{-15}, F = 46.6; Individual Shift: $p = 7.8$ x 10^{-13}, F = 39.6; ANOVA, **Figure 3.4 E**). Post hoc Bonferroni tests show SCOUT exhibits higher Jaccard similarity than CaImAn on the Gaussian dataset ($p = 2.2$ x 10^{-14}), higher Jaccard similarity than both CaImAn and cellReg on the Non-Rigid 1p dataset ($p < 1.4$ x 10^{-11}), and higher Jaccard similarity than both cellReg and CaImAn on the Individual Shift dataset ($p < 4.8$ x 10^{-9}). These results correspond with F1 score results and show that SCOUT exhibits higher cell tracking quality across all proposed clusters, not just neurons tracked across all sessions (**Figure 3.5**).

To identify possible biases toward large cluster sizes with SCOUT, we compute the Jensen-Shannon divergence between projected cluster distributions and ground truth cluster distributions for each method on the Non-Rigid 1p dataset (the dataset exhibiting the largest

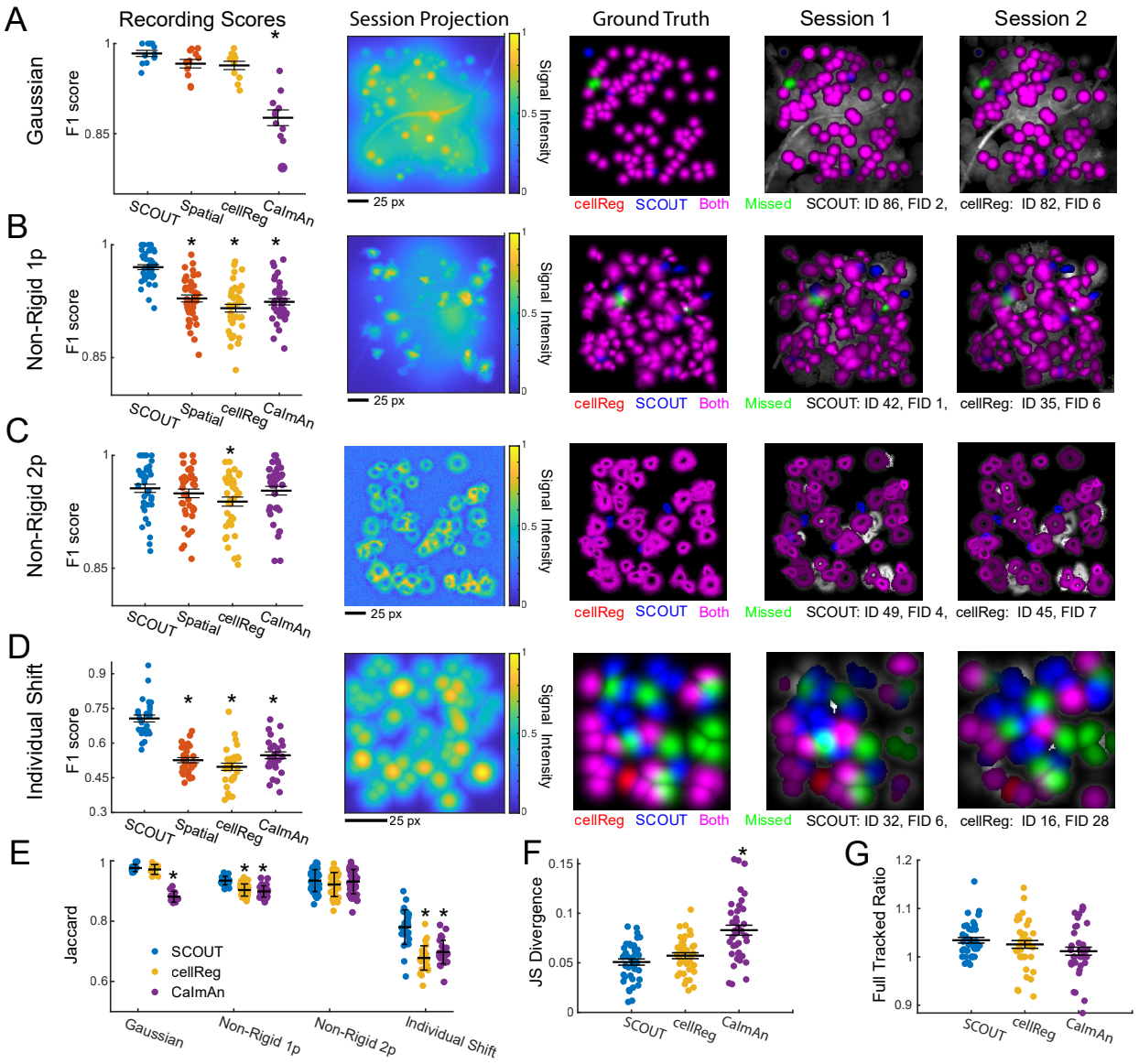

Figure 3.4: Inclusion of temporal similarity metrics improves cell tracking across sessions on simulated datasets. (Recording Scores) Maximal F1 scores (y-axis) using all tested cell tracking algorithms. Statistical differences compared with SCOUT (ANOVA, Bonferroni correction) marked with an asterisk. Methods (x-axis) are SCOUT, spatial (SCOUT using only spatial metrics), cellReg, and CaImAn. Bars indicate mean +/- standard error. Points correspond to the highest F1 score associated to each recording in the dataset. (Session Projection) Max projection of sample individual session from each dataset, across methods. (Ground Truth) Ground truth neurons available for tracking through all sessions. Colors indicate neurons tracked by each method. The number of correctly tracked (ID) and incorrectly tracked with at least one error (FID) cell register entries labeled below for each method. (Session 1-2) Tracked and missed neurons superimposed on the max projection of extracted neurons from sessions 1-2 of a sample recording. **A-D**: F1 scores, session projections, ground truth, and identified neurons by method for the Gaussian (**A**), Non-Rigid 1p (**B**), Non-Rigid 2p (**C**), and Individual Shift (**D**) datasets. **E**: Maximal Jaccard similarity scores (y-axis) from each recording and session in the simulated datasets (x-axis). Bars indicate mean +/- standard error. Asterisks indicate statistical differences. **F**: JS divergence (y-axis) of identified cluster sizes for each method (x-axis) with the ground truth for each recording in the Non-Rigid 1p dataset using parameters producing highest F1 scores. **G**: (y-axis) the ratio of neurons tracked through all sessions by each method to the ground truth number of neurons available for tracking using parameters producing highest F1 scores.

A

	Gaussian	Non-Rigid 1p	Non-Rigid 2p	Ind. Shift
SCOUT	*0.986 +/- 0.005*	***0.970 +/- 0.003***	0.956 +/- 0.005	***0.707 +/- 0.013***
spatial	*0.967 +/- 0.007*	0.929 +/- 0.005	0.949 +/- 0.005	0.526 +/- 0.010
cellReg	*0.964 +/- 0.007*	0.916 +/- 0.005	0.938 +/- 0.006	0.500 +/- 0.016
CaImAn	0.876 +/- .0131	0.924 +/- 0.004	0.953 +/- 0.006	0.547 +/- 0.015

B

	Gaussian	Non-Rigid 1p	Non-Rigid 2p	Ind. Shift
SCOUT	*0.97 +/- 0.004*	***0.93 +/- 0.002***	0.93 +/-0.006	***0.78 +/- 0.011***
cellReg	*0.97 +/- 0.005*	0.90 +/- 0.003	0.92 +/- 0.006	0.68 +/- 0.008
CaImAn	0.88 +/- 0.006	0.90 +/- 0.003	0.93 +/- 0.007	0.68 +/- 0.007

Figure 3.5: F1 and Jaccard statistics for simulated data. (upper) Average F1 score for each dataset and method. Bold entries indicate statistically significant improvement over cellReg. Italicized entries indicate statistically significant improvement over CaImAn. Spatial indicates SCOUT without temporal similarity metrics. (lower) Average Jaccard similarity accessing the ability of each method in identifying neurons for each dataset. Bold entries indicate statistically significant improvement over cellReg. Italicized entries indicate statistically significant improvement over CaImAn.

difference with more than two sessions per recording), using the parameters giving the best average F1 score for each method (**Figure 3.4 F**). ANOVA shows significant differences between average JS divergence across methods (p = 4.5 x 10^{-9}, F = 22.9), with post hoc comparisons exhibiting lower average JS divergence for SCOUT compared with CaImAn (p < 2.5 x 10^{-9}; JS: SCOUT: 0.051 +/- 0.003, cellReg: 0.057 +/- 0.003, CaImAn: 0.083 +/- 0.005). Computing the ratio of neurons tracked through all sessions for each method to the ground truth, which demonstrates comparable results for all methods when considering the statistic $|(1 - ratio)|$(**Figure 3.4 G**).

We next compute Jaccard similarity, JS distribution similarity and percentage of tracked neurons after removing 30% of neurons from each session (and the associated ground truth

cell register), to determine whether significantly reducing the available neurons per session affected cell tracking quality. This analysis shows SCOUT exhibits significantly higher similarity with the ground truth distribution compared with cellReg and CaImAn (ANOVA: F = 22.8, $p = 4.9 \text{ x}10^{-9}$, pairwise comparisons $p < 0.03$), while exhibiting lower over-detection rates of neurons through all sessions when compared with cellReg (ANOVA: F = 5.9, $p = 3.6 \text{ x}10^{-3}$, pairwise comparisons $p = 0.020$, statistic *abs(1-ratio)*) (**Figure A.3 A-C**). This implies that inferred tracking registers produced by SCOUT exhibit close similarity to ground truth distribution on similar scales with other methods, while also being robust to neuron deletion.

Next, we repeatedly perform cell tracking with SCOUT using the spatial metrics and a single temporal metric on the Non-Rigid 1p and Individual Shift datasets, as these datasets exhibit significant variation upon inclusion of temporal metrics. On the Non-Rigid 1p dataset, inclusion of each individual additional temporal metric results in higher average F1 scores, with similar results on the Individual Shift dataset, except for the SNR metric, which is expected as no background noise is simulated in this dataset (**Figure A.1 D-F, A.2 D-F**). Combining temporal metrics increases the average F1 score in most instances.

Finally, we run speed tests (24 core, 128 GB pc, 2.2 Ghz processor) by duplicating simulated data sessions to produce recordings with up to 30 sessions. We compare SCOUT, cellReg and CaImAn on 50 simulated recordings. Both cellReg and SCOUT exhibit quadratic time increase in the number of sessions (**Figure A.3 D**), compared with linear time increase for CaImAn. However, the quadratic coefficient for SCOUT is lower than for cellReg, and SCOUT significantly outperforms cellReg in terms of runtime, while maintaining comparable results with CaImAn for recordings of length up to 20 sessions.

Using both F1 and Jaccard metrics, SCOUT scores higher than alternative methods on the Non-Rigid 1p and Individual Shift datasets. Jensen-Shannon divergence shows that the distribution of identified neuron cluster sizes is significantly closer to the ground truth using

inferred cell registers from SCOUT, while the percentage of neurons tracked through all sessions is largest with SCOUT. Inclusion of temporal metrics increases average F1 score, and shows SCOUT typically runs faster than cellReg, with comparable speed to CaImAn for recordings of length up to 20 sessions.

3.6 SCOUT successfully tracks cells on in vivo multi-session recordings

We evaluate SCOUT on *in vivo* 1-photon recordings (**Figure 3.1 E**) taken from the visual cortex, prefrontal cortex, and hippocampus of mice consisting of 4-7 sessions (4,000 – 9,000 frames each) from each region, compared with annotated cell registers.

On the visual cortex dataset, optimal parameters (among those tested) give F1 scores of 0.736 and 0.590, tested on SCOUT with and without temporal metrics, respectively. This resolves to PDR of 78.0% and 61.0% and FDR of 30.3% and 42.9% (**Figure 3.6 A, Figure A.4 A**). On the prefrontal cortex dataset, optimal parameters give F1 scores of 0.701 and 0.468, with PDR 71.1% and 47.4% and FDR 30.8% and 53.9% (**Figure 3.6 B, Figure A.4 B**). On the hippocampus dataset, optimal parameters give F1 scores of 0.481 and 0.367 with PDR of 58.5% and 35.4% and FDR of 59.2% and 62.0% respectively (**Figure 3.6 C, Figure A.4 C**). Average F1 scores and PDR with SCOUT (0.639 +/- 0.080, 69.2% respectively) exceed those of cellReg (0.431 +/- 0.107, 48.9%) and CaImAn (0.368 +/- 0.123, 34,0%), and individually exceed both methods on all 1-photon datasets.

The three 2-photon recordings (**Figure 3.1 F**) consist of three sessions, one taken from the VISl (**Figure 3.7 A, Figure A.5 A**), the VISrl (**Figure 3.7 B, Figure A.5 B**) and the VISp (**Figure 3.7 C, Figure A.5 C**). Optimal parameters give average F1 scores and

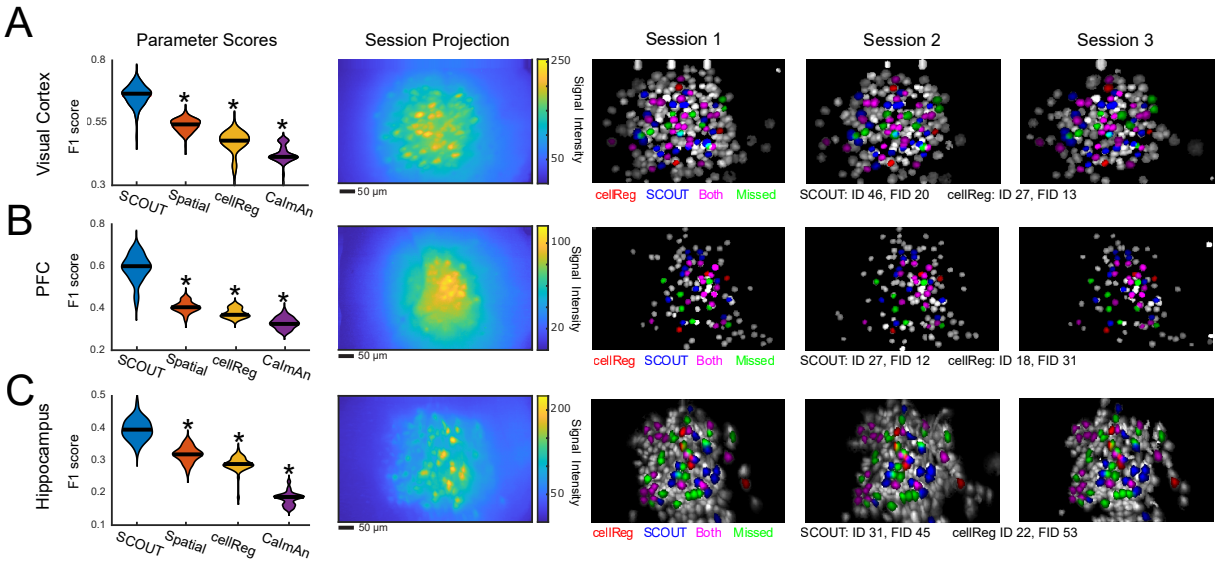

Figure 3.6: Inclusion of temporal metrics boosts cell tracking performance on 1-photon in vivo data. **A-C**: (Parameter Scores) F1 scores (y-axis) computed based on human annotation for in vivo 1-photon datasets obtained from the visual cortex (**A**, 7 sessions), prefrontal cortex (**B**, 7 sessions), and hippocampus (**C**, 4 sessions). Violin plots with median values constructed using F1 scores across parameters after outlier removal and computed using kernel density estimation. Asterisks indicate statistical differences with SCOUT (ANOVA, Bonferroni). (Session Projection) Maximum projection of the first session of each recording from all datasets. (Session 1-3) Identified neurons from cellReg and SCOUT overlaid on max projection of the human annotated neurons tracked through all sessions.

PDR for SCOUT (0.875 +/- 0.025, 91.1% with temporal metrics, 0.862 +/- 0.024, 89.5% without temporal metrics), cellReg (0.816 +/- 0.038, 85.2%), and CaImAn (0.803 +/- 0.04, 78.0%).

Together, SCOUT exhibits F1 scores $\approx 50\%$ higher than cellReg and CaImAn on 1-photon data, while median place field consistency for neurons identified exclusively by SCOUT is comparable to consistency of neurons identified by both methods on the three additional hippocampus datasets, and significantly lower than the consistency between random pairs of neurons. While the difference is smaller, SCOUT also produces top ranked F1 scores on the 2-photon dataset.

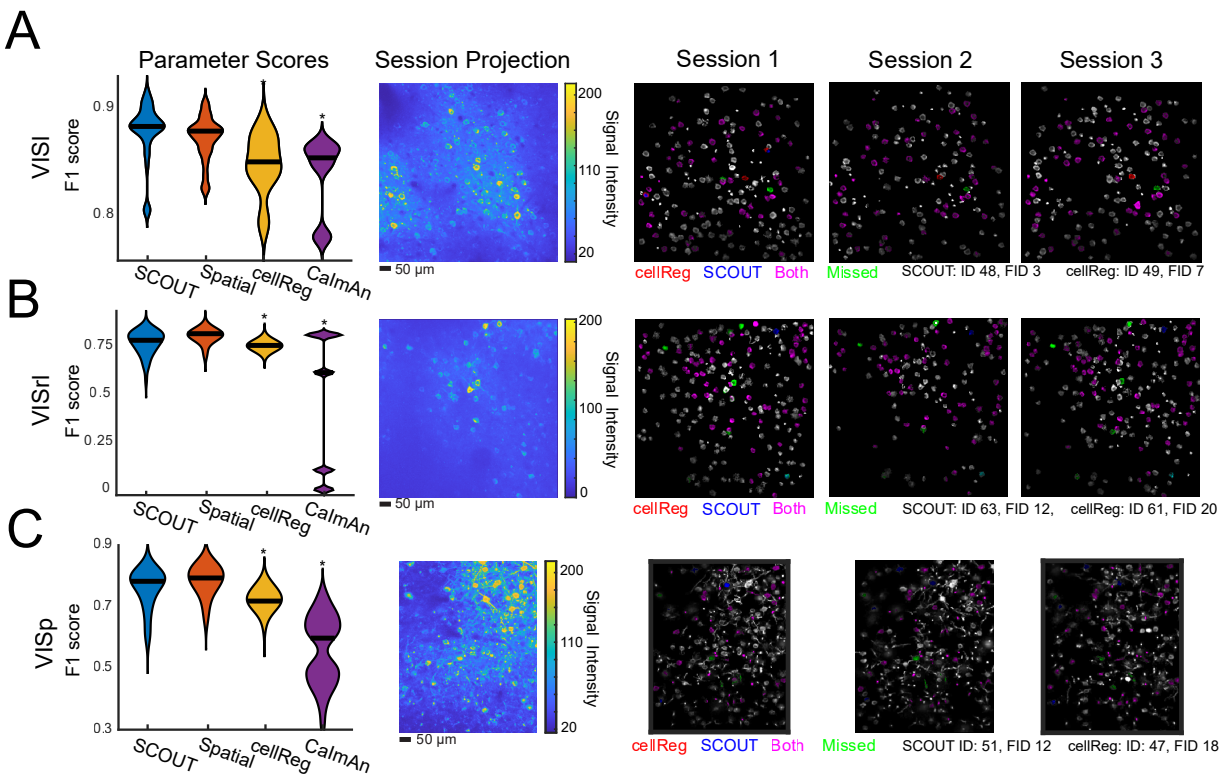

Figure 3.7: SCOUT accurately tracks neurons across 2-photon in vivo recordings. **A-C**: (Parameter Scores) F1 scores (y-axis) computed based on human annotation for in vivo 2-photon datasets obtained from the visual cortex (**A**; VISl, **B**; VISrl **C**; VISp) consisting of three sessions each. Violin plots with medians constructed using F1 scores across parameters after outlier removal and computed using kernel density estimation. Asterisks indicate statistical differences with SCOUT (ANOVA, Bonferroni). (Session Projection) Max projection of the first session of each recording. (Session 1-3) Identified neurons from cellReg and SCOUT overlaid on max projection of the human annotated neurons tracked through all sessions for sessions 1-3 of each recording.

3.7 Testing performance via place cell stability analysis

We test SCOUT and cellReg on three additional 1-photon recordings of the hippocampus to verify cell tracking results via place field stability. These recordings consist of four sessions with 10,795 frames each (at 30 hz), taken as mice run on a 1-meter linear track. Sessions are extracted via CNMF-E, and both SCOUT and cellReg are used to compute cell registers for each recording set, using the best average parameters on the 1-photon data from the previous analysis. For each neuron in each session, information scores and place fields are computed, and information percentiles are computed via random shuffling of the position

vector.

To reduce noise, we remove all register entries consisting of neurons identified in only two sessions and analyze results for all identified cell pairs (e.g. a cell tracked through four sessions produces 6 identified cell pairs). We place identified cell pairs in three categories: cell pairs identified by both methods, cell pairs identified by SCOUT, and cell pairs identified by cellReg. This results in an average of 558 +/- 208 pairs identified by both methods, 327 +/- 182 pairs identified exclusively by cellReg, and 122 +/- 48 pairs identified exclusively by SCOUT. The high variance is due primarily to fewer extracted neurons in the third recording.

For each cell pair, we compute statistics for average JS divergence between place fields, percentage of identified cells with JS divergence below assignment threshold (i.e. consistent place fields), and the percentage of identified neurons in which either place cells are matched with place cells, or non-place cells are matched with non-place cells. We analyze results at a variety of information percentiles (range [0.95,0.99]) and information score thresholds (range [0.5,1.5]). For each threshold, we restrict analysis to only neuron pairs in which at least one member exhibits information percentile or information score exceeding the specified thresholds. Prior to analysis, we verify for each recording the average JS divergence rate for identified cells by both SCOUT and cellReg is significantly lower than the average between random cell pairs ($p < 2.6$ x 10^{-9}, two-sample t-test) implying a significant number of identified neuron pairs exhibit place field stability across sessions (**Figure 3.8 A**).

Computing statistics for average JS divergence between place fields, percentiles exceeding 0.95 and information score thresholds in range $[0.8, 1.5]$ result in statistically lower JS divergence between SCOUT exclusively identified place fields compared with those identified exclusively by cellReg (min $p = 0.0079$, max $p = 0.0488$, Linear Mixed Effects model with fixed cell tracking method grouped by recording, **Figure 3.8 B**). This range of values matches a previous study [48] that suggests a percentile threshold of 0.95-0.99 of place cells, while experimentation showed that neurons with information score threshold exceeding 1.0

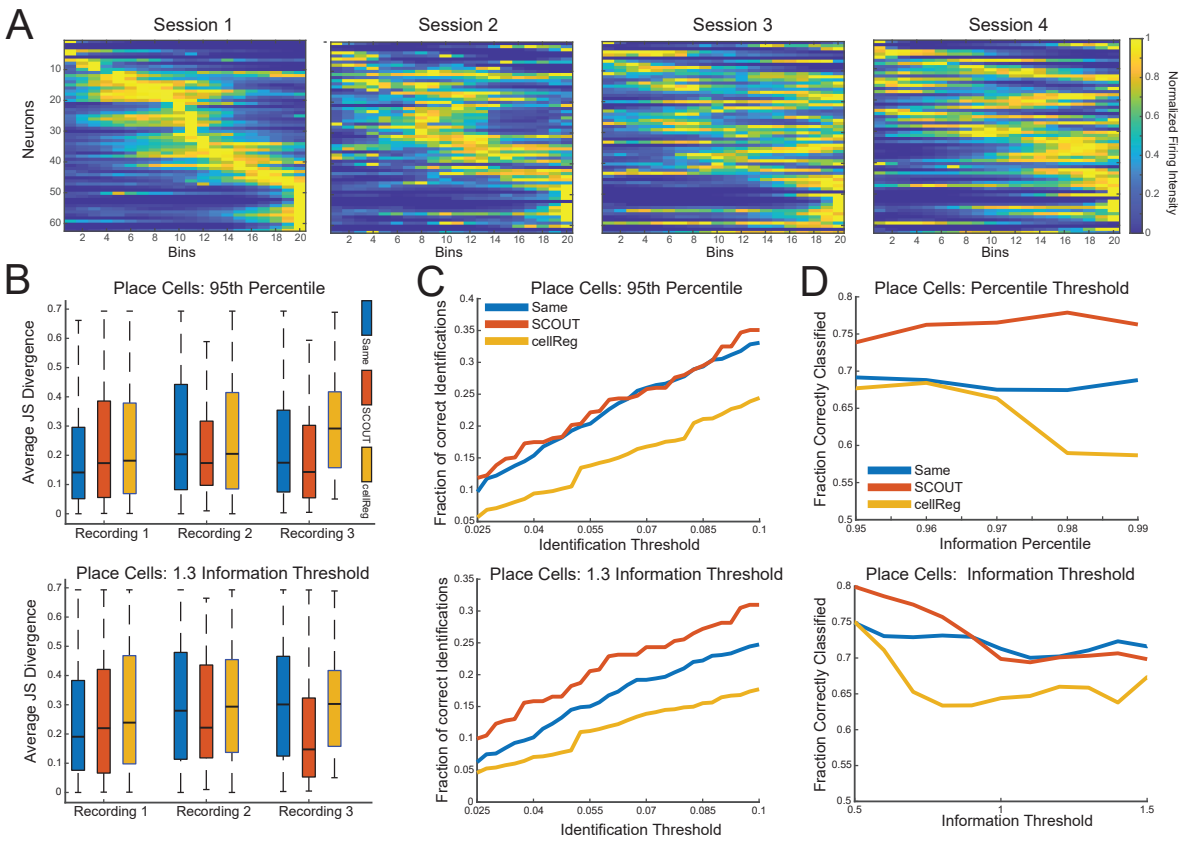

Figure 3.8: Testing cell tracking results using place field stability. **A**: Sample place fields ordered from left to right (indexed via the first session) after normalization by peak intensity. **B**: Box plot of JS divergence (y-axis) for each recording computed for identified neuron pairs by each method in which either one neuron exceeded the 95th percentile (top), or a hard threshold of 1.3 (bottom). **C**: The weighted average (across recordings) of the fraction of identified cell pairs (y-axis) which exhibit pairwise JS divergence lower than a specified threshold (x-axis), computed using cell pairs in which one neuron exceeded the 95th percentile (top), or a hard threshold of 1.3 (bottom). **D**: The weighted average (across recordings) of the fraction of identified cell pairs (y-axis) which consisted of either both place cells or both non-place cells for information percentile thresholds (top) and information score thresholds (bottom).

showed strong spatially localized firing activity.

Setting a threshold of 0.95 for percentile and 1.3 for info score threshold, we compute the percentage of identified neurons with JS divergence below an acceptance threshold (range [0.025,.1]) for neuron pairs containing place cells, a range that typically implies strong place field overlap. For percentile and information score thresholds, SCOUT exhibits a larger fraction of identified neurons below the acceptance threshold across the entire range when compared with cellReg (Average Difference, Percentile: 0.086 +/- 0.065, Threshold: 0.098

+/- 0.06, mean +/- std. error, **Figure 3.8 C**). Similarly, the fraction of SCOUT identified pairs is more likely than those identified by cellReg to identify place cells with place cells (and non-place cells with non-place cells) across sessions (Average Difference across datasets and thresholds, Percentile: 0.150 +/- 0.106, percentile threshold range = [0.95,0.99], Threshold: 0.068 +/- 0.070, information score threshold range=[0.5,1.5], mean +/- std. error, **Figure 3.8 D**), though the threshold result is biased by the third recording.

In summary, neuron pairs identified exclusively by SCOUT exhibit higher average place field similarity, while also exhibiting a larger fraction of pairs with highly similar place fields. Additionally, SCOUT identified pairs are more likely to lie within the same categorization of either place cells or non-place cells. While at the specified thresholds, SCOUT identified fewer neurons than cellReg, the average JS divergence for SCOUT pairs identified by SCOUT is lower than cell pairs identified by both methods (in two out of three recordings), while the opposite is true for cellReg (**Figure 3.8 B**). This indicates the significant possibility of a higher false identification rate for cellReg as is seen in the previous 1-photon datasets.

3.8 Effects of parameter selection on cell tracking results

Computing mean normalized standard deviation of F1 scores on 1-photon *in vivo* datasets (after outlier removal) gives (0.067 +/- 0.0072, 0.066 +/- 0.0087, 0.090 +/- .0144; mean +/- std. error), for SCOUT, cellReg, and CaImAn respectively. Computing mean normalized standard deviation on the 2-photon datasets gives (0.041 +/- 0.0098, 0.032 +/- 0.012, 0.12 +/- 0.11) for SCOUT cellReg, and CaImAn respectively. This suggests comparable parameter stability between SCOUT and cellReg, with somewhat higher variability using CaImAn. Computing the percentage of SCOUT parameters producing higher F1 scores than the maximum produced by CaImAn and cellReg on each dataset gives an average of 99.1 +/- 0.9%

on the 1-photon dataset, and an average of 48.7 +/- 20% On the 2-photon datasets.

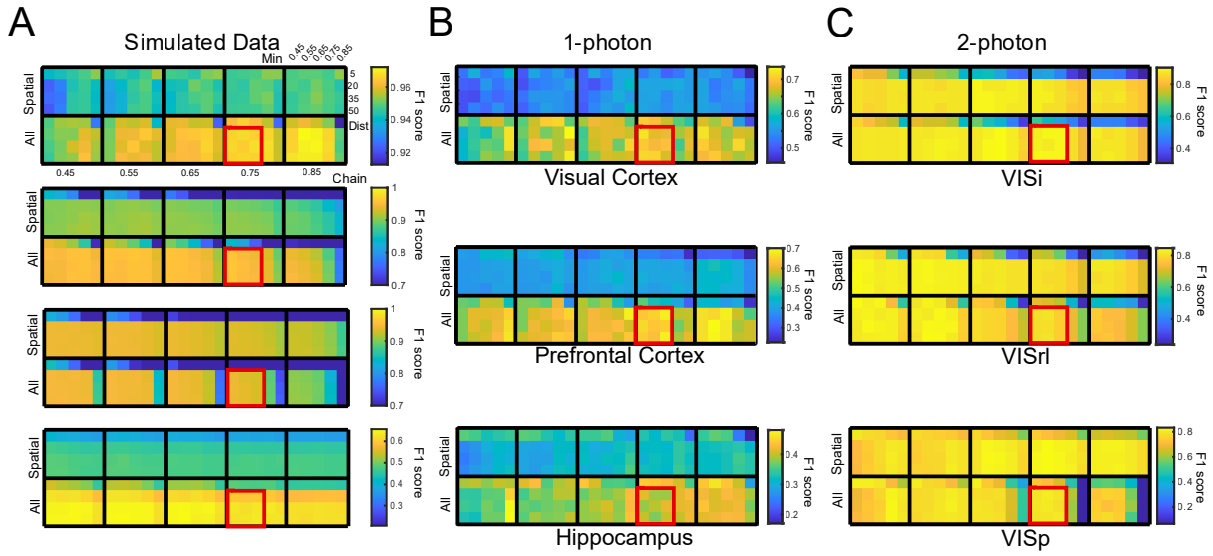

Figure 3.9: SCOUT F1 scores by parameter for each dataset. Heatmaps denote F1 scores across parameters. Each parameter in the All rows obtained using SCOUT with all metrics, while the Spatial rows use only spatial metrics. Each box contains vertical parameter changes corresponding to the max_dist parameter, and horizontal parameter changes, corresponding to the min_prob parameter. The horizontal dimension across boxes corresponds to the chain_prob parameter. **A**: SCOUT F1 score averages for the (top to bottom) Gaussian, Non-Rigid 1p, Non-Rigid 2p, and Individual Shift datasets. **B**: SCOUT F1 scores for 1-photon recordings labeled as (top to bottom) visual cortex, prefrontal cortex, and hippocampus. **C**: SCOUT F1 score for 2-photon recordings from the visual cortex labeled as (top to bottom) VISl, VISrl, VISp. Parameter ranges discussed in text labeled in red.

We compute the ratio of F1 score to maximum F1 score for each method and dataset across all parameters. Higher values indicate results closer to the optimum. Averaging across datasets, we find that SCOUT (median 0.921) exhibited significantly higher ratios than cellReg (median 0.851) and CaImAn (median 0.852) ($p < 1.2 \text{ x } 10^{-5}$, Wilcoxon rank sum test), implying SCOUT produces highly consistent results across parameters when averaging across datasets. Only 11% of cellReg parameters, and 0% of CaImAn parameters produce F1 ratios exceeding the median SCOUT value. We also identify a parameter range (labeled on **Figure 3.9 A-C**) that consistently produces strong results (average 0.950 +/- .0014 F1 ratio to optimal). These parameters emphasize a low threshold for individual identification of neurons across sessions (*min_prob,* range 0.55-0.75), combined with a high threshold for the acceptance of identified clusters (*chain_prob,* 0.75).

In summary, SCOUT exhibits comparable or higher parameter stability compared with other methods, while consistently returning top F1 scores. On the 1-photon recordings, virtually every tested parameter choice produces higher F1 scores than competing methods, while nearly 50% of 2-photon results also outperformed the topline CaImAn and cellReg F1 scores. Finally, averaging results across all datasets, the median F1 score (across parameter choices) exceeds 90% of the maximum, and we identify a parameter range on which average F1 scores consistently exceed 95% the maximum.

3.9 Conclusions

SCOUT introduces a spatial filter that significantly decreases the false discovery rate of neurons in 1-photon recordings over CNMF-E. This enables automated extraction of high quality neurons even in low SNR environments, which is critical when performing experiments that can generate hundreds of recordings.

The cell tracking module of SCOUT exhibits robust performance on all the tested datasets, generally exceeding the performance of commonly used methods such as cellReg and CaImAn in simulated situations involving significant spatial/morphology shifts or high noise levels, and *in vivo* recordings in general. SCOUT retains strong performance even in the presence of confounding variables such as non-rigid spatial shifts and poor signal extraction quality by incorporating temporal metrics, a novel clustering algorithm, and consensus clustering.

While SCOUT was initially motivated for use with 1-photon recordings, we also introduce a possible extension of the spatial filter for two photon data, and demonstrate robust cell tracking performance on 2-photon recordings, though the inclusion of temporal metrics here does not significantly improve results on the tested data. The significant differences between 1-photon and 2-photon results are due to the stronger signal quality in 2-photon data, which

may reduce the discriminatory power of some temporal metrics such as SNR in this context.

SCOUT exhibits strong performance even on recording sessions impacted by experimental conditions. Application of ketamine, NRG1, or CNO as is present in the 1-photon datasets produces significant impact on the signal intensity of individual neural signals [28, 46], but inclusion of temporal metrics still significantly improves overall cell tracking results results.

Testing SCOUT on 1-photon hippocampal recordings, we have analyzed cell tracking output using place field stability as measured by three separate metrics for which SCOUT produces strong performance. Neuron pairs identified exclusively by SCOUT exhibit lower JS divergence between place fields, a higher average percentage of SCOUT identified neuron pairs exhibited consistent place fields, and a higher average percentage of SCOUT identified neuron pairs are of the same type (place cell to place cell or non-place cell to non-place cell) when compared with cellReg.

SCOUT quickly identifies neurons across multiple sessions, with cell tracking taking less than 10 minutes for cell tracking across up to 30 sessions (depending on the number of neurons). Increasing the number of sessions or neurons per session can significantly increase the runtime, which can be addressed using a combination concatenation-cell tracking methodology, or by thresholding the neuron footprints to reduce spatial overlap, thereby decreasing the component size when clustering.

In conclusion, SCOUT shows strong extraction and cell tracking performance on both simulated and *in vivo* datasets. We have shown that inclusion of a spatial filter significantly lowers false discovery rates, while inclusion of temporal metrics when identifying cells across sessions significantly increases the quality of cell tracking. We have also shown that the cell tracking module of SCOUT exhibits strong parameter consistency across a relatively large parameter range across all datasets. We foresee that the new concepts and techniques used in SCOUT will improve capabilities for long term cell tracking related experiments,

particularly in complex situations where SCOUT retains strong performance compared with alternative methods.

Chapter 4

Chapter 4: SCOUT applications

We now present several applications which utilize SCOUT. This chapter is divided into two sections. First, we consider analysis of several long term contextual discrimination recording sets for which SCOUT was able to track neurons, but cellReg was not. Then, we develop a machine learning algorithm used to quantify the effects of experimental variation on neuron firing.

4.1 Contextual Discrimination Experimental Overview

We consider a contextual fear discrimination experiment[1] [49] in which mice are trained to discriminate between a context (visually distinguishable region) in which shock is applied, and a control context. This experiment proceeds through the standard stages of habituation (no shock in either context), learning (no shock in control context), extinction (no shock in either context), and relearning (no shock in control context).

[1]Project envisioned by Dr. Xiangmin Xu and Dr. John Guzowski. Primary data production by Dr. Steven Grieco

As the hippocampal CA1 is integral for the creation of integration of short and long term memories, 1-photon recordings were taken from the hippocampi of freely moving mice throughout the course of the experiment. A previous study in rats have indicated that distinct orthogonal neuron ensembles activate in the separate contexts after learning, and that this is integral to memory processing, as inducing inflammation in the CA1 reduces orthogonalization and decreases discrimination between contexts [50].

Mice were trained to differentiate between two similar, but visually distinct, square open field environments; miniscope imaging of hippocampal CA1 excitatory neurons was simultaneously conducted in behaving mice during the tasks (**Figure 4.1 A-B**). Mice were habituated for a 12-14 day period (with recordings taken in the final 3-5 days), in which they were allowed to freely explore each environmental context daily. At the end of this habituation phase, context discrimination training started by introducing a mild foot-shock stimulus in the stimulus context but not in the control context, which continued for 12-14 days. Mice learned to freeze as a contextual discrimination response in anticipation of the stimulus (**Figure 4.1 C**). Subsequently, a 12-14 day extinction phase in which no shock was applied led to reduction in discrimination and freezing behavior. We then reinstated the stimulus to study neural response during reacquisition of the discrimination behavioral response, for 12-14 days.

Each day, individual mice were initially introduced to a random context (either control context or stimulus context), followed by 20 minutes in a neutral context, after which the mice were introduced to the remaining context. Recordings were taken in both contexts. Mice spent 3 minutes in the control context, and 5 minutes in the stimulus context. The recordings from the stimulus context were split into a 3-minute baseline recording, and a 2-minute stimulus recording, in which a 1s 0.25-0.5 footshock was applied, 30 seconds into the stimulus recording. During the habituation and extinction phases, no stimulus was delivered. Freezing and neural responses in the stimulus context were analyzed from the

3-minute baseline recordings. Cages were thoroughly cleaned between sessions involving different mice. Mouse freezing behavior, as evidenced by a lack of movement except that necessary for respiration, was manually measured offline from video recordings of the session[2].

4.2 Longitudinal analysis of contextual discrimination associated ensemble neuronal dynamics

Using SCOUT, we obtain improved population cell tracking (when compared with cellReg) of behavior-associated hippocampal neural ensemble dynamics at single-cell resolution for longitudinal analysis. Neurons were tracked through recordings from control and stimulus (both pre- and post-shock) recordings . In total, we tracked an average of 135 $\pm$ 21 neurons extracted per mouse across 5 mice, throughout the contextual discrimination experiment that lasted approximately two months, with between 36 and 44 recording sessions. Comparatively, cellReg detected an average of 77 $\pm$ 10 neurons, not including two mice for which cellReg returned no tracked neurons.

Context-dependent neural activity at the global level (calculated as the average area-under-curve (AUC) value across all neurons for each recording session) was exhibited in three of the mice (**Figure 4.1 D-E**, (**Figure 4.2 F**). Two mice exhibited higher daily average neural activity in the stimulus context (not including footshock), compared to the control context, and one mouse exhibited increased daily average neural activity in the control context. For these mice, activity was higher in their preferred context 75.6 $\pm$ 4.3% of the time, a significant deviation from the average ($p = .02$, Welch's t-test). For the remaining experimental mice, average neural activity did not show significant preference for either context.

We identified novel neural ensembles that exhibited a sustained (over 3-5 days) increase

[2]work by Rachel Crary

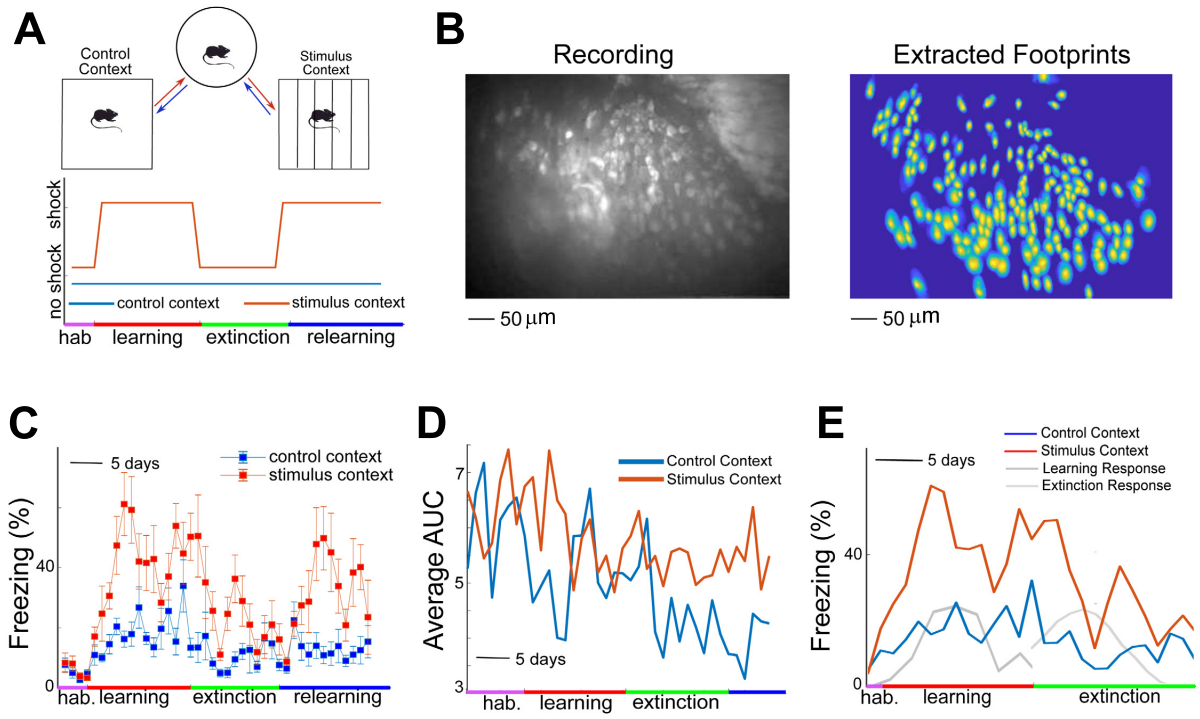

Figure 4.1: Longitudinal analysis of hippocampal ensemble activities during contextual discrimination experiments **A**: (top) Visualization of the stimulus and control contexts, as well as the experimental process. Mice are placed in either the control or stimulus context for either 3 or 5 minutes (depending on context and experimental phase), then transitioned to a neutral context for twenty minutes, before being placed in the opposite context for another 3 or 5 minutes. (bottom) Visual representation of contextual discrimination procedure. After habituation (hab.), the mouse receives a single mild footshock in the stimulus context during two distinct periods (learning and relearning) separated by an extinction phase. In the stimulus context, the brief footshock is administered 3 minutes after being placed in the context. In the control context, no shock is administered. In panels c and d, freezing levels are measured for the 3-minute period after introduction to the context, and before the administration of footshock. **B**: Max projection image (left) and spatial footprints of extracted neurons (right) by SCOUT with data collected from a long-term contextual discrimination experiment for an individual mouse. For this mouse, 168 neurons were tracked across 38 sessions. **C**: During initial phases of training with footshock, mice exhibit behavioral generalization by increasing freezing in both contexts. After several days of training, mice exhibit contextual discrimination, which is evidenced by higher proportions of time freezing in the stimulus context as compared to the control context. During extinction, freezing decreases in both contexts, but shows a greater reduction in the stimulus context. Similar qualitative results occur during relearning (reinstatement) when footshock is applied again specifically in the stimulus context. Data shown is the mean time mice spent freezing in the first 3 minutes after placement in the respective context averaged for six mice (error bars indicate standard error of the mean). **D**: Each data point represents the area-under-curve (AUC) of the extracted neural calcium signals, averaged over all extracted neurons, for both contexts, for the specified day. Calculated AUC from the stimulus context does not include the time points after application of stimulus. Average neural activity indicates significant neural discrimination between contexts throughout the experiment. This effect is noted in three of the five mice. **E**: Freezing rates for a single mouse in stimulus (red) and control (blue) contexts, are compared with the mean cell activity across active neurons in the learning and extinction phases of the experiment (grey and black, respectively). Note the relative increase of neural activity during the peak of the acquisition and extinction phases, where changes in behavioral responses are most prominent.

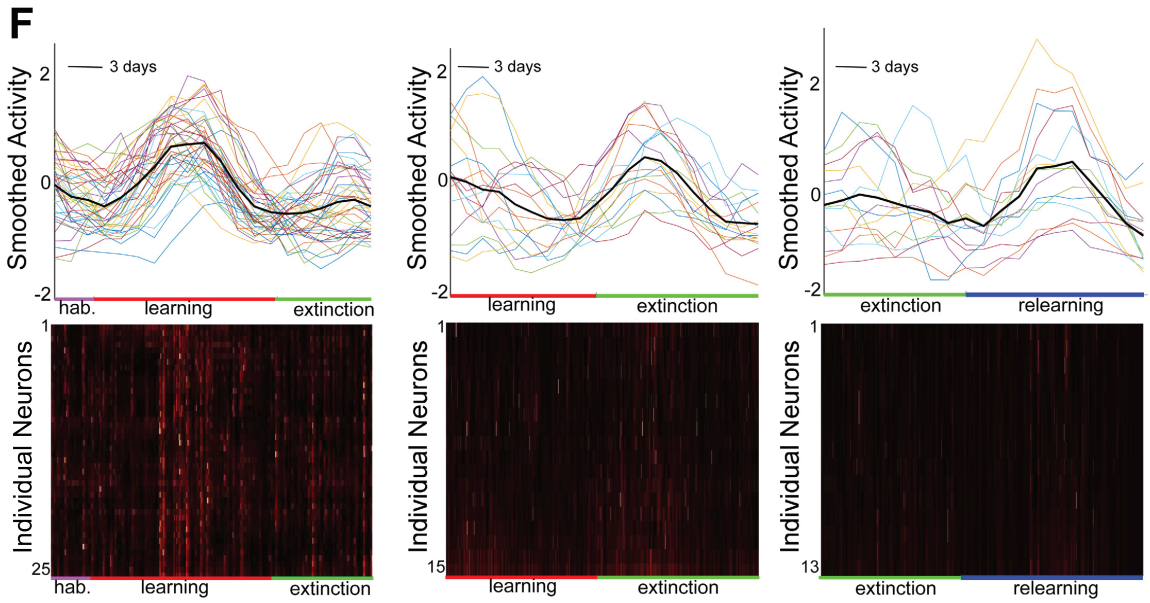

Figure 4.2: **F**: (top row) Each plotted point represents the smoothed daily area-under-curve (AUC) of the calcium signal trace for a single neuron. Here we plot daily AUC for a subset of neurons that exhibit increased activity during different stages of the contextual discrimination experiment. This panel demonstrates context-dependent neural activity changes observed across 3 of the mice (raw (unsmoothed) daily AUC is plotted in Supplementary Fig. 3B for reference). The data shows a clearly distinguishable increase in distributed activity with learning, extinction, and relearning. (bottom row) Raster plots of individual calcium signals for the corresponding neuron subsets show visually distinguishable increases in activity at the corresponding times above. Each row of the heatmap indicates the signal intensity for a single neuron, throughout the portion of the experiment indicated at the bottom. Signals were normalized to have the same maximum intensity for visualization purposes.

in activity followed by return to a baseline level. Across the three mice exhibiting global context preference, 32 ± 5.5% of all neurons exhibited stimulus-context dependent response changes in at least one context through the course of the experiment. At the learning stage, 16.7 ± 5.4% of the cells exhibited increased activity in either the stimulus context (prior to footshock) or control context within 1-5 days of imposition of shock, descending to baseline levels before the next phase of the experiment (**Figure 4.2 F**, left, top and bottom panels), with 13.2 ± 1.6% and 8.9 ± 2.3% exhibiting increased activity in the extinction and relearning phases (**Figure 4.2 F**, middle and right), starting 1-5 days after experimental phase change, respectively.

During learning, 7.7 ± 4.0% of neurons exhibit increased activity (in at least one context)

exhibit increased activity in both contexts, while only 4.8 ± 2.6 of neurons exhibiting increased activity during extinction exhibit increased activity in both contexts. A proportion of active neurons during extinction (<10%) exhibit increased activity across contexts or across learning phases.

4.3 Evidence for Localization of Cell Response after Introduction or Removal of Stimulus

Activity for each neuron (in each session) is computed via AUC (area under curve), which is the sum of the fluorescence trace (C) normalized by the length of the session. An initial K-means clustering applied to daily neural activity implied the existence of a sustained activity increase occurring within 1-5 days of the initiation of the learning, extinction, and relearning phases in a significant subset of neurons, in both contexts, for nearly all mice. In order to test this theory, and demonstrate that directly after introduction and removal of stimulus were the only three probable stages in the learning process where a significant, sustained increase in activity occurs among a large subset of extracted neurons, we created a comparison template of length 11 days, with behavior similar to the detected behavior (namely increase in activity over four days, followed by 3 days holding steady, followed by decrease in activity). The number of neurons exhibiting similar activity patterns (based on a correlation of 0.65 with the template) starting at each day of the trial was counted, and significantly more neurons followed this activity pattern within two days of the imposition or removal of stimulus, than the average at other time periods in the experiment, indicating that the responses were stimulus dependent. Modifying the template in terms of activity rate increase and sustained activity intensity did not significantly alter the results, though decreasing the template length resulted in the detection of additional neurons with characteristic activity starting at each day, diluting the uniqueness of the effect.

4.4 Neural network models for place field decoding

We now review the fundamental artificial neural network model, some variations, and describe the motivation and application of neural networks for place field decoding.

4.4.1 Biological motivation

Artificial neural networks were (as the name implies) originally motivated by the unfolding understanding of biological neuron communication [51]. Initially, the fundamental operating principle was "neurons that wire together, fire together," [52] that is correlated activation of pre- and post-synaptic neurons strengthens the connection, making activation of the pre-synaptic neuron more likely to trigger an action potential in the post-synaptic neuron.

This motivates the construction of the artificial neural network, a graphical mathematical model composed of nodes (referred to elsewhere as neurons, we retain nodes temporarily to distinguish between artifical and biological neurons) and weights. The nodes are organized in layers, and each node performs a non-linear transformation of a unique weighted aggregate of information passed from the previous layer. A fundamental difference between artificial neural network and the biological analog is the sequential nature of the ANN. That is, information is passed layer by layer through the ANN. Even adding recurrent connections merely results in a discretization of a biological network. Attempts to circumvent this issue such as the spiking neural network [53] have not proven popular due to issues with training and interpretation.

The final biological motivation for the ANN is the concept of the activation function. Recall that neurons exhibit a non-linear behavior in activation. When a neuron receives sufficient excitatory input to exceed a threshold (unique to the neuron), the neuron rapidly depolarizes producing an action potential that propagates down the axon facilitating further communi-

cation. However, in this instance, nature betrays us. While the necessity for a non-linear activation function is obvious (otherwise neural networks are just linear functions), the original non-linearities designed to mimic action potential (perceptron, sigmoid) have proven inferior to alternatives such as the ReLU (Rectified Linear Unit) [54], a function that returns zero if the input is negative, and the identity if the input is positive. Of course, much research has gone into developing new and improved non-linearities (leaky-ReLU, ELU [55] among others), but the ReLU is still commonly used today.

From this background we may now define the artificial neural network. This model consists of sequential layers of nodes, and associated weights. Each node is given as a linear combination of values (plus a bias term) outputted from the previous layer, followed by application of the activation function. Explicitly, for layer j let $\mathbf{x}$ denote the output of the previous layer, σ_j denote an activation function, $\mathbf{W}_j$ an $n \times m$ matrix, where n is the number of nodes in the previous layer, and m the number of nodes in the current layer, and $\mathbf{b}_j$ a constant vector of length m. The output of this layer is given by $\sigma(\mathbf{W}_j\mathbf{x} + \mathbf{b}_j)$, where σ acts elementwise on the output vector.

The final core concept of the ANN is the loss function. This is a scalar valued function computes the difference between the true output $\mathbf{y}$ and the ground truth output $\hat{\mathbf{y}}$. Defining this loss function as $L(\mathbf{y}, \hat{\mathbf{y}})$, L should take a minimum value (typically zero) when $\mathbf{y} = \hat{\mathbf{y}}$. Generally, the loss function should be differentiable (almost everywhere), which enables us to compute derivatives $\frac{\partial L}{\partial \theta}$ for all parameters θ in the network (primarily weights and biases). This enables parameter training via gradient descent [56] or variants thereon (momentum, ADAM [57]).

One addition to the model is regularization. Neural networks have a high model capacity, and can suffer from overfitting the training data (e.g. the model becomes sensitive to small variations in training that are specific to the training set, preventing generalization). This can be mitigated via regularization. Common regularization techniques include weight pe-

nalization (penalizes high magnitude weights [58]), dropout (replaces a subset of inputs at each layer by zeros, which reduces dependence on individual inputs [59]), and batch normalization (a method which z-scores the inputs to each layer followed by a learned affine transformation before application of the activation function [60]).

4.4.2 Why neural networks?

Why use neural networks? First, neural networks have high capacity. Second, neural networks have become increasingly cheap and easy to train. Third, neural networks are extraordinarily modular.

Neural networks with just a single hidden layer are capable of approximating any Borel measurable function from one finite dimensional space to another, as long as there are sufficient nodes available[3] [61]. Thus, neural networks form an easily constructible and trainable class of universal approximators.

Both forward and backward operations (required for training) are extraordinarily cheap, consisting entirely of matrix multiplication, addition, and nearly trivial activation functions. The advent of GPUs, designed for graphical processing, provided an ideal hardware for the parallel computations required for neural network training. Thus, training models with millions of parameters became feasible. The success of deep learning (neural networks with large numbers of hidden layers) in image classification and other fields made the technique both profitable and popular.

Finally, neural networks are extremely modular. In many instances, it is possible to directly encode additional information or desired features into the model either through its architecture or through the loss function. Convolutional neural networks (inspired by analysis

[3] In practice, single layer neural networks tend to overfit, and the number of required neurons may be exponentially larger than required for deep networks.

of animal visual circuits [62]) are essentially an architectural modification of the traditional ANN incorporating shared weights for individual image regions. Small modifications of the loss function and architecture can turn a neural network into a generative model [63]. This modularity is critical to quickly developing models which incorporate biological or practical features.

4.4.3 Neural network model for place cell decoding

Recall that place cells fire preferentially when an animal is in a certain region (the place field). Therefore, it is possible to decode an animals location from the neural signals of place cells. Previous methods have utilized Bayesian probabilistic models [64], but here we develop a neural network model for position decoding (a similar method was developed concurrently [65]).

Given data in the form $n_{neurons} \times n_{timepoints}$ consisting of spike trains for each neuron (in practice, the deconvolved spikes $s_i(t)$ in the CNMF-E model), we first split the data into train and test sets. As our data is provided as a time series, the test set must consist of a continuous segment of time points. Generating a random index, we select a window consisting $10 - 20\%$ of the total length of the spike trains immediately following the index as the test set. The train set consists of the remaining data, excluding a 5 second window around the test set to ensure train and test sets are independent.

Next, we infer spiking probabilities from the given spike trains, as the neural network model requires continuous data for training. We tested multiple methods for inferring probabilities, including Gaussian filters, moving windows, and exponential approximations . The training set is then normalized (e.g. z-scored) and the parameters saved for application to the test set.

We next construct modified train/test sets by selecting a window around each time point and utilizing all timepoints within that window as input to the neural network (**Figure 4.3 A**). In effect, this creates input matrices of size $(2win\_length + 1)n_{neurons} \times train - \alpha$, where win_length is the window length, *train* are the lengths of the train indices (replace train with test for testing) respectively, and α indicates the amount of data for which a time window is not available. Finally, we train the model to identify position based on the given spike trains, using the mean absolute error loss function ($L(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{N}\sum|y_j - \hat{y}_j|$).

4.4.4 Model caveats and extensions

In practice, place cells are not exclusively or perpetually activated by location. Therefore, position decoding is muddled when animals exhibit complicated behaviors. Non-moving animals, engaging in grooming or feeding activities, do not always generate characteristic place cell behavior, which can interfere in model training and interpretation. We follow the standard practice of removing neural signal data when the mouse is receiving water rewards (e.g. at the ends of the track), and subset the remaining data based on thresholding velocity [25]. Additionally, as our experiment was run on a linear track, only one coordinate was predicted as output of the model.

The model itself can be modified for various ends. Among other aspects, we have considered decoding position using a one sided window so as to infer position based only on previously seen data. We have considered variants of the model such as convolutional neural networks, a model which provides additional structure by utilizing shared weights (in this case for the first layer). This equates to utilizing a shared aggregation/activation acting on the individual windows of data for each neuron, followed by passing the data to a standard ANN. We also implemented a recurrent neural network (RNN) which is more suitable for time series learning as it incorporates a recurrent state computed based on the previous data in the time series,

which is combined with the current input to decode position.

We also implement methods for sensitivity analysis. In general, place cells do not uniquely encode position and have overlapping place fields. Therefore, only a small subset of neurons are required for position decoding. We considered gradient based sensitivity, which utilizes the average gradient $\frac{\partial y}{\partial x_i}$ magnitude to determine the importance of each input value to the overall result. Due to the complexity of the neural network, such analyses can prove inaccurate, so we also implemented a leave one out strategy, in which the model is consecutively trained with a subset of neurons missing to determine the importance of each input. Going one step further, after training at each stage, we remove the least important neuron (e.g. the neuron whose removal caused the smallest reduction in accuracy) until the accuracy was reduced by a certain factor, thus identifying a minimum subset of neurons to predict position throughout the track (an average of 40-50 neurons was required to maintain an R^2 coefficient of determination [66] at 93% of the maximum.)

4.5 Applications of the model to neural circuit modeling

4.5.1 Trisynaptic circuit

In addition to the place cell structure, the hippocampus also exhibits a clearly defined neural circuit. In other words, the various structures of the hippocampus can be viewed as nodes in a directed graph, in which edges indicate one-way communication. The primary neural circuit in the hippocampus is called the trisynaptic pathway [67], which begins with the entorhinal cortex signaling to the dentate gyrus, which signals to the CA3 hippocampal region, then to the CA1 region, at which point signal is passed through the subiculum, and back to the cortex (**Figure 4.3 B**). While the trisynaptic pathway is typically viewed as

a unidirectional circuit, this is explicitly a high level view, and non-canonical connections (such as the CA3 recurrent connections [68]) significantly influence the functionality of the circuit.

Because the base level structure for spatial memory encoding is so simple, and because the hippocampus displays a well-defined neural circuit, place cells have been a preferred cell type for experimentation involving spatial learning and memory.

Recent studies have identified a significant non-canonical back projection of signal from the subiculum to the CA1 [69]. The functional role of this communication pathway is largely unexplored. We first apply monosynaptic rabies tracing to verify the existence of the subicula back projection. The direct path from the visual cortex to CA1-projecting SUB neurons and their output to the CA1 and perirhinal cortex strongly suggest that one of the roles for this circuitry may be to provide critical information necessary for conjunctive object–place representations in the hippocampus and the perirhinal cortex. To test this, we used DREADD (designer receptors exclusively activated by designer drugs [70])-mediated inactivation of CA1-projecting excitatory SUB neurons to determine whether they are necessary for object-location learning behavior. This enabled us to selectively inhibit back projecting neurons in the subiculum via admission of Clozapine N-oxide (CNO). At this point, we performed 1-photon calcium imaging of the CA1 neurons, pre, mid, and post CNO treatment, while freely moving mice sought water rewards at the end of a 1-meter linear track. Our primary question is whether deactivation of the subicular neurons caused a fundamental rearrangement in the way the track location was encoded in the brain, and whether this effect (if present) was permanent.

4.5.2 Neural network identifies functional variations due to altered neural circuits

We consider and experiment consisting of three segments, a control (Ctrl) segment (2 sessions), a CNO segment (1 session) in which the subiculum back projection is deactivated via DREADDs, and a Post-control (Pctrl) segment (2 sessions) recorded 7-10 days after CNO administration. A single Control and Post-control session are selected for most computations. Neurons are recorded from the hippocampal CA1 in each individual session, followed by tracking through all sessions, for the purpose of identifying individual signaling variations consequent to CNO adminstration.

Preliminary analysis[4] identifies the place cells, place fields, and the correlation of place fields between control and CNO/Post-control sessions, based on the place cells identified in the control sessions (**Figure 4.3 C**). Both Ctrl vs CNO and Ctrl vs Pctrl show similarly low place field variation (with expected higher variation when position is randomly shuffled) (**Figure 4.3 D**), indicating that little to no remapping of place fields occurs after deactivation of backprojecting neurons.

We compute the information content[5] of each neuron, and compute z-scores via randomization of positiona, and select a threshold for place cell identification (1.96). We consider information score differences between Ctrl and CNO sessions, and Ctrl and Pctrl sessions. This divides the neurons into 4 groups, Bit Decrease, Bit Increase, Un-recovered and Unassigned (**Figure 4.3 E**), based on the pairwise comparisons between sessions (Bit Increase neurons exhibit higher information content in the CNO session compared with Ctrl and Pctrl, Bit Decrease shows the opposite effect, Un-recovered do not exhibit consistent Ctrl and Pctrl behavior, Un-assigned did not show significant changes between sessions).

[4]The following two paragraphs as well as **Figure 4.1 C-G** produced primarily by co-author Dr. Suoqin Jin

[5]computed in bits/sec, this is the same metric used previously to define place cells

Approximately 50% of neurons show decreased information content in the CNO session (e.g. Bit decrease) compared with $\sim 23\%$ Bit Increase, and 27% unrecovered (**Figure 4.3 F**). This indicates the majority of neurons are less sensitive to location after deactivation of the backprojection. This *rate remapping* is evidenced in the peak calcium signal intensities between sessions among Bit Decrease and Bit Increase neurons, which show significantly lower peak intensities of the Bit Decrease neurons in the CNO session compared with the Ctrl/Pctrl (**Figure 4.3 G**).

To confirm results and identify commonality between timepoints with low position encoding signal, we train the neural network model (standard ANN, trained separately using all neurons and defined place cells) on the control session, and test how consistently position can be decoded in the CNO and Pctrl sessions. Results show low prediction error in the Ctrl (training set), with significantly higher error on the CNO, followed by position decoding recovery in the Pctrl session, which was evidenced in the $n = 5$ mice considered (**Figure 4.3 H**). When restricting to place cells (and to a lesser extent all neurons), this characteristic recovery of decoding accuracy is present in the R^2 coefficient of determination. Typical variance in position decoding occurred in the middle of the linear track. Fewer neurons encode position in the middle of the track, and those that do show less consistency between sessions (**Figure 4.4 I-J**).

Examining the specific time points for which decoding gives poor results in the CNO session (by setting a threshold of 10 centimeters), we find that these time points exhibited lower overall signal (when aggregated across neurons), compared with time points of high accuracy (t-test, $p < .05$). This indicates that rate remapping of the required neurons for spatial decoding directly effects spatial position recognition for these specific time points (as opposed to the aggregate computed in Bit Decrease/Increase neurons).

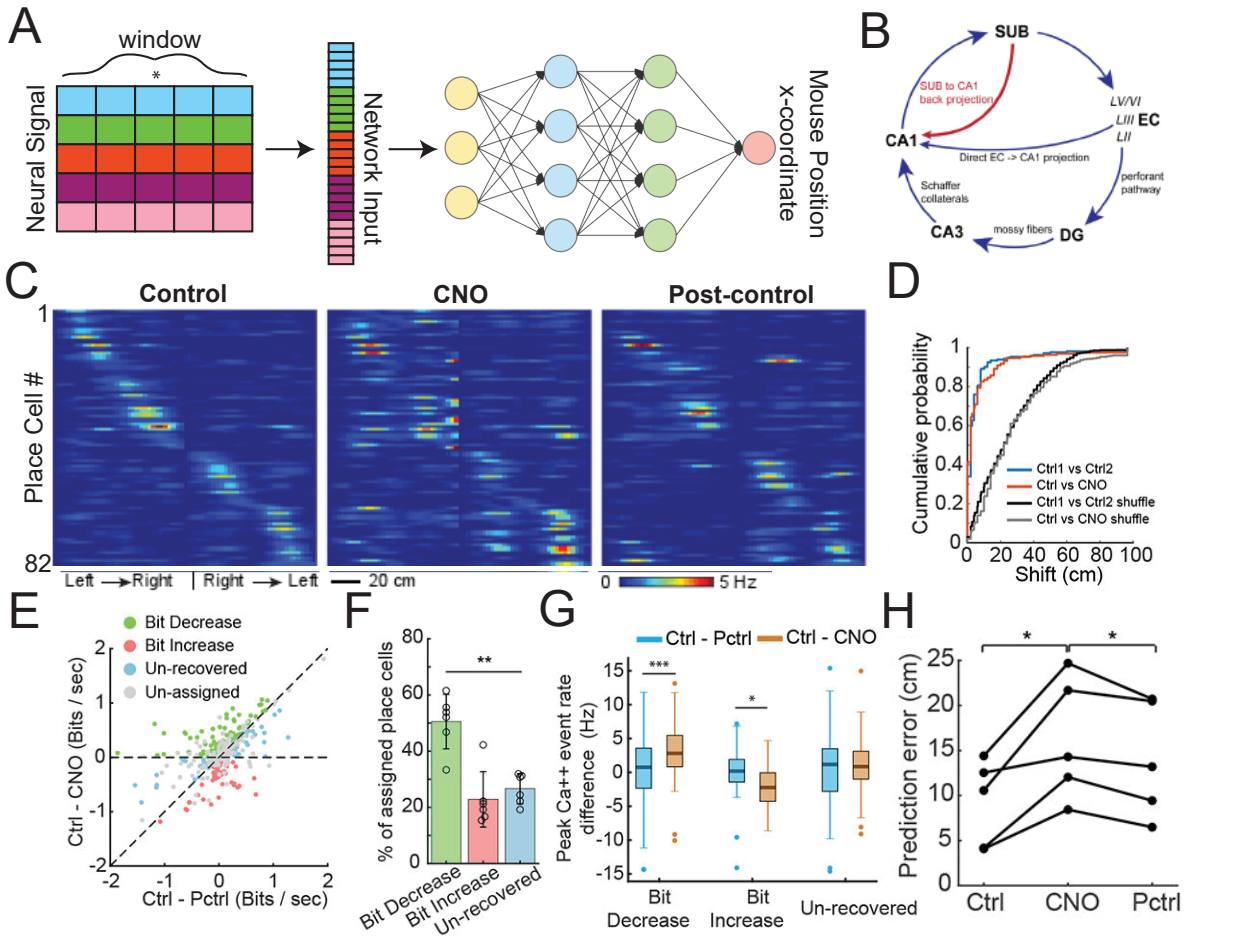

Figure 4.3: A machine learning model for position decoding from place cell signal confirms the causes of memory loss in after circuit deactivation. **A**: For each timepoint (asterisk) firing activity is taken from a window around the point, and used to predict position. **B**: A diagram of the trisynaptic pathway (with extensions to the Subiculum and Entorhinal cortex), with major pathways in black, and the backprojection in red. **C**: Place fields of identified place cells, ordered by location in the control session. Sessions are split based on the direction of travel. **D**: The distributions of correlation-peak shift magnitudes for the place cells in the saline experiment (blue line) and the CNO experiment (red line. There is no significant difference between the distribution of Ctrl vs CNO and the Ctrl1 vs Ctrl2 ($p = 0.38$, two-sample KS, two-tailed), or their shuffled variant (black and grey). **E**: Recorded CA1 place cells can be classified into 3 non-overlapping groups termed bit-decrease, bit-increase, and un-recovered, based on the statistical significance of differences in information scores between CNO and Ctrl, and between CNO and Pctrl. Statistical testing employed a jackknife resampling method for each place cell with appropriate corrections for error terms. **F**: Of the 201 place cells that show significant differences (assigned place cells) from 6 mice, 50% show decreased information scores in CNO sessions compared to the control and post-control sessions (bit-decrease group, green bar). A smaller subset ($\sim 23\%$) show increased information scores in CNO compared to the control and post-control (bit-increase group, red bar). The remaining ones are the un-recovered group which accounts for $\sim 27\%$ of place cells. Comparing the mean percentages of each type seen in each mouse, a significant difference in the % of place cells among these three groups is observed ($p = 0.002$, repeated measures ANOVA, $n = 6$ mice). Data are presented as mean ± SE in the bar plot. **G**: Comparisons of peak calcium event rates between Ctrl - Pctrl and Ctrl - CNO in bit decrease (two-tailed, paired t-test, p $= 3 \times 10^{-7}$, $n = 97$ cells), bit increase (two-tailed, paired t-test, $p = 0.027$, $n = 48$ cells) and un-recovered groups (two-tailed, paired t-test, $p = 0.20$, $n = 56$ cells), respectively. **H**: Quantification of the prediction errors between predicted trajectories and actual trajectories for decoding accuracy using the trained model based on the first control session. Each line represents the prediction errors of Ctrl, CNO and Pctrl sessions from one mouse. Significantly higher prediction errors are observed in CNO sessions compared to those in Ctrl ($p = 0.016$, two-tailed, paired t-test) and Pctrl ($p = 0.015$, two-tailed, paired t-test) sessions. $n = 5$ mice.

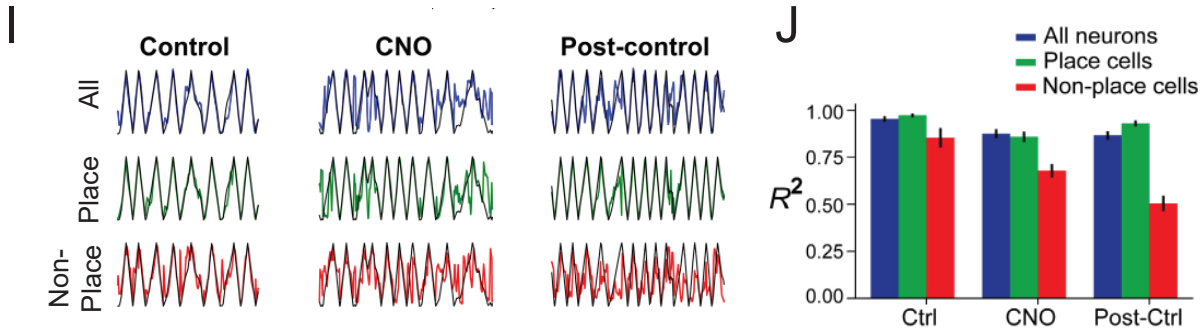

Figure 4.4: **I**: Decoded position from a sample session using all, place, and non-place cells (colored) compared with the true x-coordinate (black line). **J**: R^2 coefficients for regression using prior cell subsets for each session. Decoding using place cells shows a statistically significant decoding accuracy recovery in the Pctrl session (t-test, $p < .05$). Error bars taken as mean $\pm$ standard error across mice.

Chapter 5

Neuron function and the transcriptome:

5.1 Transcriptomics overview

The transcriptome is the set of all RNA transcripts, both coding and non-coding, in an individual cell or cell population, though this is sometimes restricted to mRNA[1] [71]. Unlike the genome, the transcriptome changes dynamically on short time scales, allowing the cell to manufacture proteins required for specific tasks or to adapt to changing conditions.

The proteome (the range of proteins present in a cell) is constructed from the transcriptome as mRNA is translated into proteins. The composition of the transcriptome over the cell's lifespan determines the functionality and activity of the cell. In neurons, synthesized proteins determine all aspects of neuron communication. AMPA and NMDA receptors dyanmically influences synaptic strength [72], with other proteins playing critical roles in neurotransmitter reuptake, among other critical processes.

The transcriptome and the proteome are only moderately correlated ($r = 0.482$, [73]), meaning that analysis of the transcriptome only accounts for the portion of protein products

[1] Ribonucleic acid which carries the instructions for protein synthesis

present in the cell. However, the transcriptome is much easier to sample and sequence, with next-gen sequencing enabling profiling of the full transcriptome across tens or hundreds of thousands of cells at low cost [71].

Therefore, we turn our interest to the effect of the transcriptome on neuron firing activity.

5.2 Profiling of activity and transcriptome

Patch-seq [17] involves sequential electrophysiological experimentation, transcriptome sequencing, and morphology reconstruction of the same neuron. This allows direct analysis of the correlation between transcriptome and firing pattern.

Electrophysiological experimentation on neurons utilizes patch clamps [74] to stimulate voltage differences across the cell membrane, allowing analysis of spiking behavior (**Figure 5.1 A-B**). A variety of statistics can then be computed from the elicited firing trains, including resting membrane potential, threshold potential (minimum voltage to induce firing), action potential width and amplitude, after polarization and after hyperpolarization (AHP).

Morphology is harder to quantify numerically, as the result is essentially a physical reconstruction of the tested neuron. This notwithstanding, features such as axon length and soma width show significant variation between celltypes, and can be used when quantification is required.

5.3 The role and function of interneurons

GABAergic[2] inhibitory interneurons comprise $\sim 10-20\%$ of all neurons in the brain, and contribute to the regulation of synaptic transmission, network oscillations and neural plasticity [75–77]. The diversity of interneurons is crucial to brain function, as their dysfunction results in psychiatric disorders such as schizophrenia [78]. Interneuron types are distinguished by neurochemical marker expression, morphology, intrinsic electrophysiological properties, and functional roles for behavior. Parsing how these characteristics impact functions according to interneuron type has been difficult due to lack of molecular tools [79, 80]. While different interneuron types were first identified morphologically by Cajal over 100 years ago, genetic labeling and manipulation using transgenic mice now allows for further identification of types and further analysis of their functions [81–85]. Protein and peptidergic neurochemical markers including parvalbumin (PV), calretinin (CR), somatostatin (SOM), vasointenstinal peptide (VIP), neuropeptide Y (NPY), cholecystokinin (CCK), enable identification of interneurons based on their transcriptomic profile.

There are important species differences in the chemical peptidergic signatures of inhibitory neuron subtypes. In contrast to the rat, mouse interneurons co-express SST and CR, and this dual peptidergic expression gives rise to unique morphological and electrophysiological properties [86]. Co-expression of different peptidergic neurochemical markers appears to extend the functional repertoire of interneurons.

Cholecystokinin (CCK) was initially characterized as a gastrointestinal peptide and was subsequently determined to be one of the most abundant neuropeptides in the central nervous system (CNS) [87]. It is synthesized as a preprohormone of 115 amino acids that can be enzymatically cleaved into multiple isoforms. The most abundant isoform in the CNS is the sulfated octapeptide CCK-8S which is expressed at high levels in the hippocampus,

[2]The remaining portions of this chapter are co-authored with Dr. Steven Grieco

amygdala, septum and hypothalamus [88]. CCK is present at high levels in interneurons, but is not exclusively expressed in inhibitory interneurons as it is also expressed in excitatory pyramidal cells [89, 90]. Further evidence suggests that CCK-expressing interneurons (CCK+) are heterogenous, as expression of CCK is relatively promiscuous [91]. The functional consequence of this is not yet understood.

In order to determine how CCK expression extends the functional properties of interneurons, we began by studying CCK co-expression in calcium binding protein parvalbumin (PV)-expressing (PV+) interneurons. PV+ interneurons exhibit fast firing rates, distinct intrinsic electrical and synaptic properties, and control the precise timing of network oscillations. They are strongly implicated in schizophrenia and a host of other psychiatric and neurological diseases [92]. Recent studies show that PV+ cells express CCK at the mRNA level, but this has not been validated further with any depth [93–97].

We show that CCK expression in interneurons is correlated with electrophysiological differences in activity among PV+ neurons, and that CCK expression varies continuously across PV+ interneurons (e.g. on a transcriptional gradient), and that neurons at both ends of the spectrum exhibit statistically different electrophyisological characteristics.

5.4 Single-cell data and analysis pipeline

We first analyze raw sn/sc-RNA-seq from publicly available datasets. A total of five datasets were analyzed, consisting of two scRNA-seq (10x v3 and SMART-Seq) sample sets covering the isocortex and hippocampal formation, and three snRNA-seq (1 10x v2, 2 10x v3) samples taken from the primary motor cortex of the adult mouse [93]. Individual datasets were analyzed via the same pipeline on the Seurat platform [98]. This pipeline consists of log-normalization, filtering of cells with outlying numbers of features, identification of variable

genes, scaling, and PCA dimensionality reduction [99]. Interneurons were identified based on GAD1 and GAD2 expression, as well as provided taxonomy from stated publications. This resulted in 21,991 cells from the Allen SMART-seq dataset (labeled 2019Allen), 96,323 cells from the Allen 10x dataset (labeled 2020Allen), 10,944 cells from the BICCN 10x v2 dataset (labeled 2021AIBS_v2), 6,817 from the first BICCN 10x v3 dataset (labeled 2021AIBS_v3), and 18,093 cells from the second BICCN 10x v3 dataset (labeled 2021BROAD_v3). Of these, a total of 28,951 were labeled as PV interneurons, and 7,053 were labeled as SNCG. Datasets were integrated via Harmony [100]. Results corroborated previous experiments showing interneuron cell types were shared across brain regions, motivating combined analysis across regions [93]. Dropout gene reads were imputed via MAGIC [101]. Correlation analysis was performed via the WGCNA [102]. Gene ontology analysis was performed via the topGO R package [103].

The transcriptomic results of Patch-seq samples consisting of interneurons from the visual cortex was analyzed via the same pipeline seen above [94]. Data consisted of 4,270 cells (after quality control) of which 724 were labeled as PV interneurons, and 250 as SNCG interneurons. Analysis of the electrophysiology of the associated cells began with identification of the rheobase threshold, and identification of cell latency and average firing rate at 30-40 mV above the rheobase threshold (depending on data availability). Statistical results were computed using the Wilcoxon rank sum test. PV+/CCK+ and PV+/CCK- were identified as the PV interneurons showing CCK expression in the top (bottom) 25% of PV cells after magic imputation.

5.5 Effects of transcriptional variation on function

5.5.1 Linkage between electrophysiological and transcriptional characteristics of PV/CCK interneurons

We characterized the electrophysiological properties of PV-/CCK+ (e.g. SNCG), PV+/CCK+ and PV+/CCK- cells from the Patch-seq data made publically available from the Allen Institute [94]. This data set consists of 4,270 cortical GABAergic interneurons from adult mouse visual cortex. From this data set we first confirmed interneuron identity based on GAD1 and GAD2 expression, as well as taxonomy. Identified cell types of interest are PV+, CCK+, and SNCG+. From our analysis of 4,270 cortical GABAergic interneurons, we identified 724 neurons as PV+ interneurons, and 250 cells were identified as SNCG+ (**Figure 5.1 A**). Based on dimensionality reduction data, CCK+ cells show clear overlap in the UMAP region for PV+ interneurons and most CCK+ cells are found in the SNCG+ UMAP region (**Figure 5.1 A**). We then characterized PV+/CCK+ and PV+/CCK- interneurons by identifying the PV interneurons showing CCK expression in the top or bottom 25% of PV+ cells after Markov Affinity-based Graph Imputation of Cells (MAGIC). Visual inspection of single spike features elicited by depolarizing current pulses reveals differences between these operationally defined PV+/CCK+ and PV+/CCK- interneurons, particularly with regards to AHP amplitudes (**Figure 5.1 B**). Similar to results from previous recordings, the mean firing rates and amplitudes do not significantly differ between PV+/CCK+ (n = 178) and PV+/CCK- (n= 169) interneurons. We then quantitatively analyzed peak AHP values in this data-mined set comparing PV+/CCK+ and PV+/CCK- interneurons and find that the magnitude of the AHP in PV+/CCK- interneurons is significantly larger than that of PV+/CCK+ interneurons as their peak values are significantly more negative than those of PV+/CCK+ cells (**Figure 5.1 C-D**). Together these results show that there are electrophysiological features that distinguish between PV+ interneurons that strongly express

CCK versus those that do not.

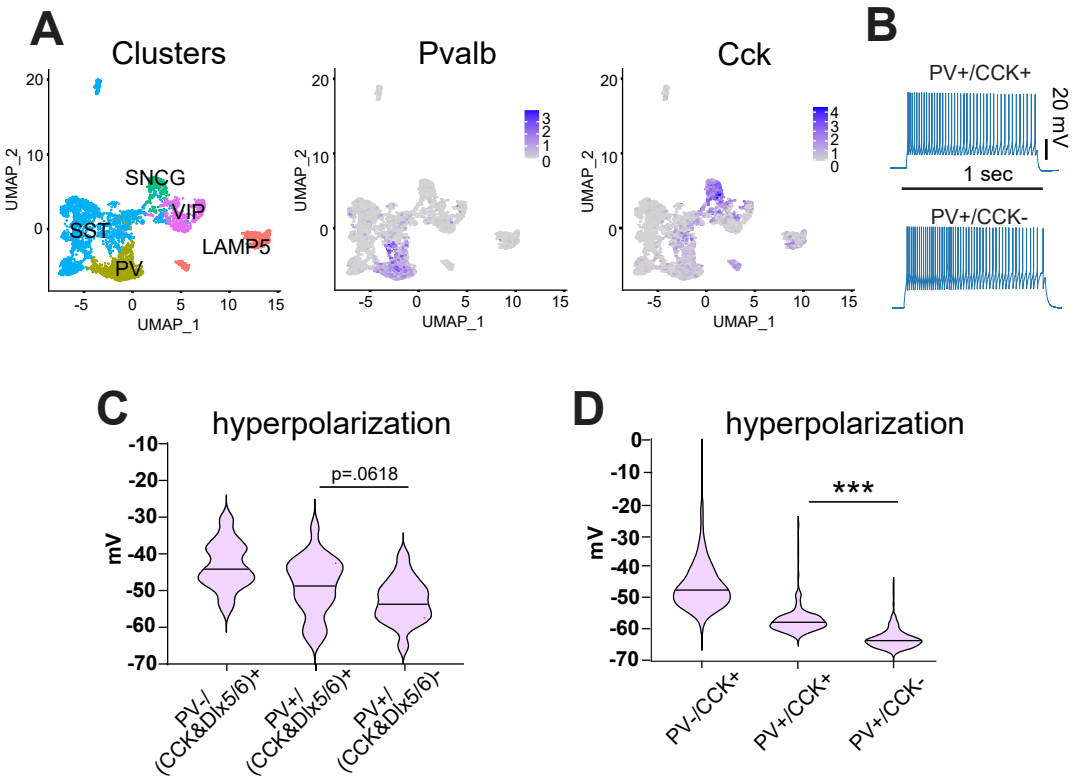

Figure 5.1: Comparison of electrophysiologies for PV+/CCK+ and PV+/CCK- GABAergic inhibitory interneurons in mouse. (**A**-**B**,**D**) Patch-Seq data from adult mouse visual cortical interneurons made publically available from the Allen Institute. (**C**) Cell types were identified via expression of interneuron gene markers (Sst, Pvalb, Vip, Lamp5, and Sncg) (**A**, Left). (Pvalb) Pvalb gene expression, imputed using MAGIC, shows strong localization of Pvalb expression to the PV subtype (**A**, middle). (Cck) Cck gene expression, imputed using MAGIC, is exhibited in both SNCG and PV interneurons (**C**, right). For electrophysiology data are further analyzed through the following pipeline. Cck expression is imputed using magic, and PV interneurons were identified as PV+/CCK+ if Cck expression is above the 75th percentile, and PV+/CCK- if Cck expression is below the 25th percentile. (**B**) Current clamp electrophysiological recording from acute brain slices containing visual cortex. Representative voltage traces for PV+/CCK+, or PV+/CCK- interneurons in response to current clamp mode-induced depolarizations. The vertical scale bar is 20mV. The horizontal scale bar is 1 second. (**C**) Summary violin plot of the electrophysiological property "after hyperpolarization potential" (AHP) from PV-/(CCK& Dlx5/6)+ (n=15), PV+/(CCK& Dlx5/6)+ (n=24), and PV+/(CCK& Dlx5/6)- (n=19) interneurons, profiled in the Xu lab (production and analysis by Dr. Steven Grieco and Dr. Yanjun Sun). The violins show data with the median line in the middle. The top and bottom of the plots are the minima and maxima (Kruskal-wallis test: overall p=0.0007. Mann Whitney test: PV+/(CCK& Dlx5/6)+ versus PV+/(CCK& Dlx5/6)-, p=0.0637). (**D**) Summary violin plot of AHPs from PV-/CCK+ (n=198), PV+/CCK+ (n=178), and PV+/CCK- (n=169) interneurons from the Allen Institute data set. The violins show data with the median line in the middle. The top and bottom of the plots are the minima and maxima. PV+/CCK- interneurons have significantly larger AHP amplitudes compared to PV+/CCK+ interneurons (One-way ANOVA: overall p< 6.72 e-96. Bonferroni multiple comparisons: PV+/CCK+ versus PV+/CCK-, p=1.19 e-6).

5.5.2 Transcriptomic characterization of PV+/CCK+ inhibitory interneurons

We characterized the transcriptomic properties of PV+/CCK+ and PV+/CCK- interneurons by mining published data from the Allen Institute for Brain Science and the recent article bundle (published in *Nature*) from NIH's Brain Research through Advancing Innovative Neurotechnologies (BRAIN) Initiative Cell Census Network (BICCN) [9, 93]. A total of five datasets were analyzed, consisting of two sc-RNA-Seq (10x v3 and SMART-Seq) sample sets covering the isocortex and hippocampal formation, and three sn-RNA-Seq (one 10x v2, two 10x v3) samples taken from the primary motor cortex of mouse (**Figure 5.2 A**). From these data sets, interneurons were identified based on GAD1 and GAD2 expression, as well as the provided taxonomy from the stated publications. Identified cell types are SST+, PV+, VIP+, SNCG+, and LAMP5+. From this analysis a total of 28,951 neurons were identified as PV+ interneurons, and 7,053 cells were identified as SNCG+ (**Figure 5.2 A**). CCK+ cells show clear overlap in the UMAP region for PV+ interneurons, and again, most CCK+ cells are found in the SNCG+ UMAP region (**Figure 5.2 A**).

We then characterized PV+/CCK+ and PV+/CCK- interneurons by identifying the PV interneurons showing CCK expression in the top or bottom 25% of PV+ cells determined by MAGIC. There is a preponderance of agreement across data sets. The most-well corroborated genes are found to be systematically up- or down-regulated in PV+/CCK+ interneurons as compared to PV+/CCK- interneurons in all 5 data sets (**Figure 5.2 B**). Notably the Cox6a2 gene is robustly up-regulated in PV+/CCK+ interneurons. In PV+ interneurons, there is a positive and highly significant correlation between increased CCK expression and increased Cox6a2 expression. This correlation with Cox6a2 expression is not present in SST+, VIP+, SNCG+ or LAMP5+ cells (**Figure 5.2 C**). We then performed gene co-expression analysis and identified several highly co-expressed modules with PV+ interneurons. A major module includes cytochrome oxidase genes, that are also highly

co-expressed with CCK (**Figure 5.2 D**). We then analyzed differentially expressed genes in the COX/CCK module for their associated ontologies; overexpressed ontologies include "Axon/Dendrite", "Regulation of Synaptic Transmission", "Mitochondrial ATP synthesis", "Electron Transfer", and "Mitochondrial respiration" (**Figure 5.2 E**).

To better understand how the transcriptomes of PV+/CCK+ and PV+/CCK- relate to known attributes of either PV+, CCK+, or SNCG+ interneurons, we analyzed the gene expression levels of individual genes associated with PV+, CCK+ and/or SNCG+ interneurons. Synuclein Gamma (SNCG) is expressed in CCK+ interneurons in which CCK expression does not overlap heavily with other peptidergic markers based on transcriptomic analysis [91]. Our transcriptomic analysis confirms that PV expression is high in PV+ cells, CCK expression is high in CCK+ cells, and SNCG expression is high in SNCG+ cells. Further, SNCG is not expressed strongly in PV+/CCK+ or PV+/CCK- cells. CCK is expressed strongly in SNCG+ cells as well as in PV+/CCK+ cells. Expression of Synuclein Alpha (SNCA) also is strongly confined to SNCG+ cells, but PV+/CCK+ cells appear to show more expression than PV+/CCK- cells. Cannabinoid Receptor 1 (CNR1) expression, which is often associated with CCK+ cells, mirrors SNCA expression; CNR1 is strongly expressed in SNCG+ cells [104–106], but PV+/CCK+ cells seem to show more expression than PV+/CCK- cells. The alpha-1A subunit of the P/Q-type voltage-gated calcium channel (CACNA1A) is associated with PV+ interneurons, and the alpha-1B subunit of the N-type voltage-gated calcium channel (CACN1B) is more associated with CCK+ interneurons. Both PV+/CCK+ and PV+/CCK- cells express CACNA1A strongly, whereas SNCG+ cells do not. SNCG+ cells express CACNA1B strongly, and both PV+/CCK+ and PV+/CCK- express CACNA1B to a much lesser extent. Similarly, the gene expression patterns of the presynaptic calcium sensor synaptotagmin 2 (SYT2), is associated with PV+ interneurons [107]. Both PV+/CCK+ and PV+/CCK- cells express SYT2 strongly, while SNCG+ cells do not. Both PV+/CCK+ and PV+/CCK- cells express TAC1, a marker of PV+ basket cells, while SNCG+ cells do not [93, 94, 97, 108]. PV+/CCK+ express COX6a2 very strongly, whereas PV+/CCK-

cells do not. Overall, these transcriptomic results show that PV+/CCK+ and PV+/CCK- interneurons have distinct molecular phenotypes (**Figure 5.3 F**).

Since we find in PV+ interneurons that there is a positive and highly significant correlation between increased CCK expression and increased COX6a2 expression, we focused on the COX6a2 protein. Upon searching the literature, we found that Cox6a2, which is an ADP-sensitive isoform of Cox6a [109, 110], was described previously as being highly restricted to PV+ interneurons [111]. It was postulated that the high energy demands of fast-spiking PV+ interneurons requires distinctive machinery in the mitochondrial oxidative phosphorylation pathway, thus necessitating Cox6a2 expression as part of complex IV (which also integrates cytochrome c oxidase) [112]. Previous studies determined that PV+ interneurons account for ∼60% of all Cox6a2+ cells in the brain [111]. Guided by our transcriptomic findings and these published results, we tested the hypothesis that the CCK+ subset of PV+ interneurons also strongly express COX6a2. Using COX6a2 immunostaining in the CA1 region of brain sections from our PV-TdTomato: CCK& Dlx5/6-GFP mouse, we find that most PV+/CCK+ cells are COX6a2+ (n= 168 cells; ∼89% positive), but most PV+/CCK- cells are not COX6a2+ (n= 109 cells; ∼14% positive) (**Figure 5.2 G**). This immunostaining result agrees well with our transcriptomic findings that CCK expression by PV cells correlates strongly with increased Cox6a2 expression (**Figure 5.2 C**). We also find that PV+ interneurons account for ∼70% of all Cox6a2+ cells. Together these results show that PV+/CCK+ interneurons express COX6a2, whereas PV+/CCK- cells do not, and that PV+/CCK+ interneurons have unique molecular profiles that can contribute their functional importance.

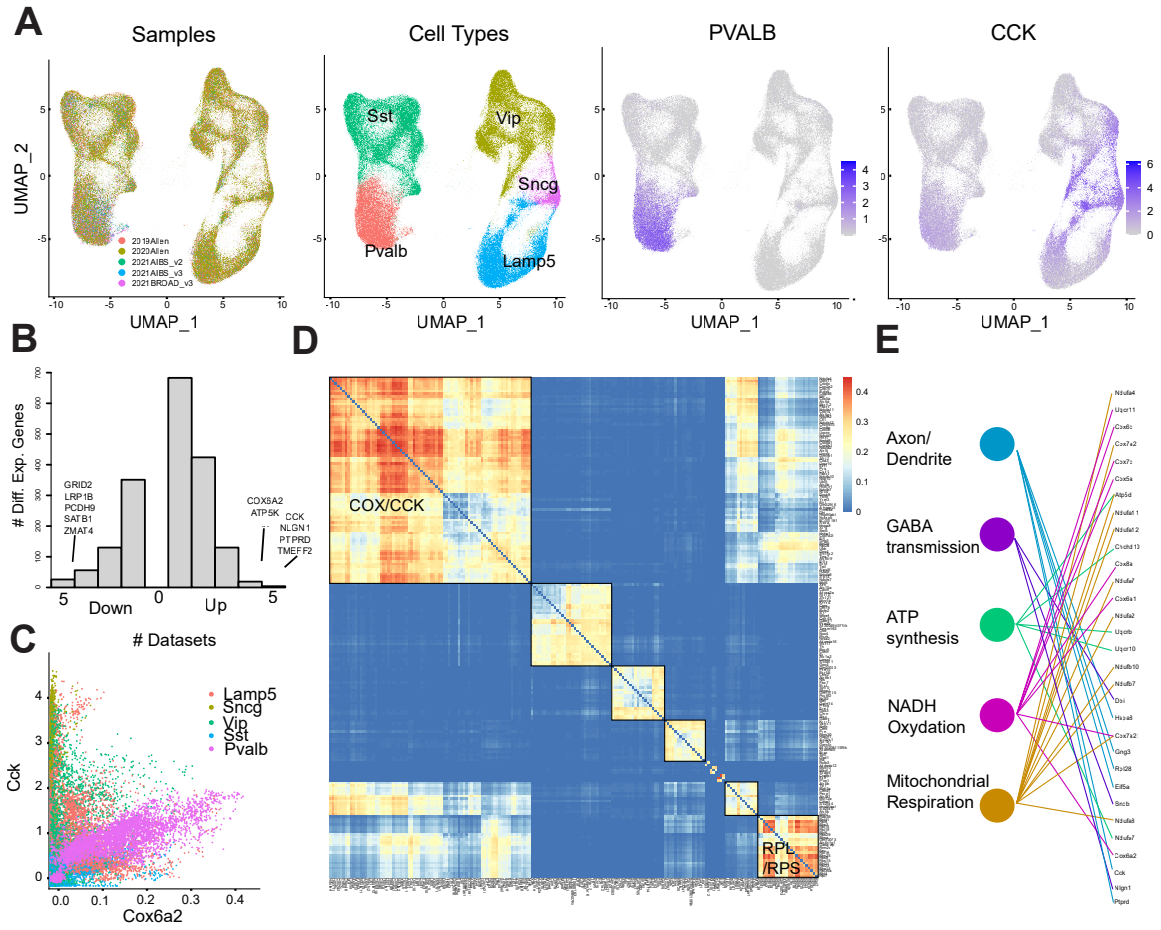

Figure 5.2: Comparisons of transcriptomes of PV+/CCK+ and PV+/CCK- GABAergic inhibitory interneurons in mouse. (**A**) (Samples) Individual sn-RNA-Seq datasets were processed as follows. Cells exhibiting an extremely high or low number of features were eliminated. Count matrices were log-normalized and scaled. Highly variable genes were identified and used as features for PCA dimensionality reduction. Next, anchors were identified and datasets were integrated via the Seurat framework. The data was projected to two dimensions via UMAP for visualization. (Cell Types) The integrated datasets were clustered, and cell types were identified via expression of interneuron gene markers (Sst, Pvalb, Vip, Lamp5, and Sncg). These are characteristically divided into the MGE (SST, PV) and CGE (VIP, LAMP5, and SNCG) lineages. (Pvalb) Pvalb gene expression, imputed using MAGIC, shows strong localization of Pvalb expression to the PV subtype. (Cck) Cck gene expression, imputed using MAGIC, is exhibited in both SNCG and PV interneurons. (**B**) Individual datasets are further analyzed through the following pipeline. Cck expression is imputed using MAGIC, and PV interneurons were identified as PV+/CCK+ if Cck expression is above the 75th percentile, and PV+/CCK- if Cck expression is below the 25th percentile. Differential expression testing between the groups identified >50 genes differentially expressed in at least four datasets. Many of these genes were associated to differences in interneuron communication (Cck, Nlgn1, Ptprd, Cox6a2, Grid2, Pcdh9). (**C**) Cck and Cox6a2 expression (post imputation) are highly correlated in the PV cell type (0.73), but not in any other interneuron subtype. (**D**) Gene co-expression analysis via WGCNA identified several highly coexpressed modules. Genes exhibiting coexpression less than 0.15 were removed from the visualization. The rest clustered into 6 groupings, including a COX grouping, that was also highly co-expressed with CCK. (**E**) Differentially expressed genes and genes in the COX/CCK module were analyzed for their associated ontologies. Overexpressed ontologies included Axon/Dendrite, Regulation of Synaptic Transmission, Mitochondrial ATP synthesis, Electron Transfer, and Mitochondrial respiration. These ontologies were associated to mostly non-overlapping gene groupings, possibly indicating largely independent functional subclusters within the COX/CCK module.

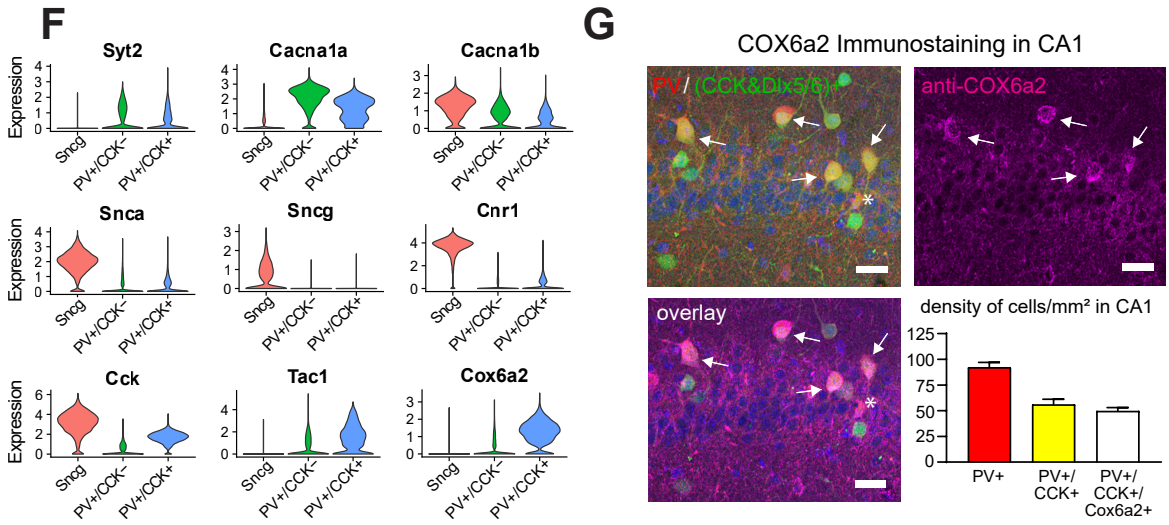

Figure 5.3: (**F**) Gene expression of typical CCK+ or PV+ interneuron function exhibited significant differences between PV+ and PV- (SNCG+) groupings, as would be expected from their disparate lineages. Additionally, Cck, Tac1 and Cox6a2 exhibit large differences between PV+/CCK+ and PV+/CCK- groupings. Data are presented as violin plots. (**G**) Immunostaining of the COX6a2 protein (magenta) in CA1 from PV-tdTomato; CCK& Dlx5/6-GFP (yellow) mouse brain sections (top left, right). COX6a2 staining was robust in PV+/(CCK& Dlx5/6-GFP) (white) cells (bottom left). COX6a2 was co-localized with ∼89% of PV+/(CCK& Dlx5/6)+ (white, arrows). Cell with an asterisk is an example of a PV+ cell that is COX6a2 negative. Quantification of cell densities in CA1 for COX6a2 immunopositive PV+/(CCK& Dlx5/6)+ cells (bottom right)(N=3 mice). Bars indicate mean ± standard error.

5.5.3 Discussion

In this section we motivate the interest in transcriptional gradients by showing that: 1) the transcriptome and the electrophysiology of neurons are strongly connected. 2) subtle differences in the transcriptome within the same cell type manifest in distinct electrophysiological characteristics. While these characteristics are not as pronounced as those between cell types (e.g. PV+ vs PV-/CCK+, which have completely separate cell lineages), they are measurable, and are likely critical to varying neuron functionality. 3) Transcriptome gradients are composed of multiple correlated genes with varying functionality.

This motivates the creation of a method for agnostic detection of transcriptional gradients, which can then be tested for subtle functional/morphological variations.

Chapter 6

Machine learning models for transcriptional gradient detection

6.1 Variational Autoencoders

Autoencoders are a specific type of neural network model, consisting of a paired encoder and decoder [113]. The encoder transforms the input data into a (typically) lower dimensional space, while the decoder transforms the encoded data back to the original dimensionality. In theory, the encoder and decoder are treated as general functions, but in practice, these are typically artificial neural networks. The encoder and decoder are taken (or trained) such that the loss between input and output (reconstruction loss) is minimized, hence they are a form of unsupervised network[1].

Autoencoders perform two unique functions. First, they encode high dimensional data into a low dimensional latent space. These latent dimensional variables represent information contained in the high dimensional input, but avoid the curse of dimensionality. As the decoder

[1] they require no labels

only has input from the latent variables, training the autoencoder forces the latent dimensional space to represent the essential differences between input points. Thus in scRNA-seq analysis, the latent dimension representation can be used for clustering [114]. Decoders attempt to represent the input, however passing the data through the low dimensional space results in a bottleneck which removes extraneous information and noise. Thus autoencoders can be used to denoise high dimensional data [115].

We can modify the formulation as follows in order to construct an autoencoder based generative model (a variational autoencoder, VAE [116]) as follows.

Given input $\mathbf{x}$ let $P(\mathbf{x})$ denote the probability distribution of the input space. The goal is to model P using a parameterized distribution p_θ, which will relate the input space and a latent representation. Denote $p_\theta(\mathbf{x}, \mathbf{z})$, a joint distribution of input variables $\mathbf{x}$ and latent variables $\mathbf{z}$. Marginalizing $\mathbf{z}$ gives

$$p_\theta(\mathbf{x}) = \int_{\mathbf{z}} p_\theta(\mathbf{x}, \mathbf{z}) d\mathbf{z} \tag{6.1}$$

$$= \int_{\mathbf{z}} p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z}) d\mathbf{z}. \tag{6.2}$$

The end goal is to maximize $p_\theta(\mathbf{x})$ with respect to parameters θ.

Typically, computation of $p_\theta(\mathbf{x})$ is intractable, thus we cannot generally compute the posterior $p_\theta(\mathbf{z}|\mathbf{x})$. We introduce a new distribution $q_\phi(\mathbf{z}|\mathbf{x})$ to approximate the posterior.

Following the variational inference paradigm, the goal is to identify parameters θ, ϕ (which are now neural network parameters) we consider the quantity

$$D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})).$$

By definition

$$\begin{aligned} D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})) &= \int_{\mathbf{z}} q_\phi(\mathbf{z}|\mathbf{x}) \log(\frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x})}) d\mathbf{z} \\ &= \int_{\mathbf{z}} q_\phi(\mathbf{z}|\mathbf{x}) \log(\frac{q_\phi(\mathbf{z}|\mathbf{x}) p_\theta(\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x}) p_\theta(\mathbf{z})}) d\mathbf{z} \\ &= log(p_\theta(\mathbf{x})) + D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})) - E_{z\sim q_\phi(\mathbf{z}|\mathbf{x})}(log(p_\theta(\mathbf{x}|\mathbf{z}))). \end{aligned}$$

Solving for $log(p_\theta(\mathbf{x}))$, we get

$$\begin{aligned} log(p_\theta(\mathbf{x})) &= D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})) - D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})) + E_{z\sim q_\phi(\mathbf{z}|\mathbf{x})}(log(p_\theta(\mathbf{x}|\mathbf{z}))) \\ &\geq -D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})) + E_{z\sim q_\phi(\mathbf{z}|\mathbf{x})}(log(p_\theta(\mathbf{x}|\mathbf{z}))). \end{aligned}$$

We thus minimize the tractable quantity $D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})) - E_{z\sim q_\phi(\mathbf{z}|\mathbf{x})}(log(p_\theta(\mathbf{x}|\mathbf{z})))$ (the negative of the evidence based lower bound, ELBO).

We now specify formulations for the required distributions. For the Gaussian VAE, we set

$$\begin{aligned} p_\theta(\mathbf{z}) &\sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\ p_\theta(\mathbf{x}|\mathbf{z}) &\sim \mathcal{N}(f(\mathbf{z}), c\mathbf{I}) \\ q_\phi(\mathbf{z}|\mathbf{x}) &\sim \mathcal{N}(g(\mathbf{x}), h(\mathbf{x})), \end{aligned}$$

for some functions f, g, h, parameterizable by variables θ, ϕ.

Substituting into the ELBO, our goal is to find

$$\arg\min_{\phi,\theta} \; D_{KL}(\mathcal{N}(g(\mathbf{x}), h(\mathbf{x}))||\mathcal{N}(\mathbf{0}, \mathbf{I})) + E_{z\sim q_\phi}(\frac{\|x - f(z)\|^2}{2c}). \tag{6.3}$$

Interpreting f, g, h as neural networks, the first term can be interpreted as a form of reg-

ularization, while the second is (up to a constant factor) the reconstruction error, with an intermediate sampling step in the latent space.

Applying backpropagation, the gradient must be passed through the expectation term. But this term appears in the expectation itself. Therefore, we use the reparamaterization trick as follows. Instead of sampling $\mathbf{z}$ directly from $\mathcal{N}(g(\mathbf{x}), h(\mathbf{x}))$, we reparameterize the distribution by sampling $\epsilon \sim \mathcal{N}(0, 1)$, and setting $z = h(\mathbf{x})\epsilon + g(\mathbf{x})$ enables backpropagation of gradients through the sampling.

6.1.1 Gumbel-Softmax VAE

As seen earlier, the standard variational autoencoder utilizes Gaussian probability distributions in the latent space to model $q_\phi(\mathbf{z}|\mathbf{x})$. However, this model is only applicable to continuous data. Many applications are dependent on a discrete latent encodings.

Recall the standard Gumbel distribution

$$f(z) =\propto exp(-e^{-z}), \tag{6.4}$$

from which a sample can be computed via

$$g = -\log(-\log(u)), \tag{6.5}$$

where u is sampled from $U(0, 1)$. The Gumbel distribution satisfies the following theorem. Let X a discrete random variable with $P(X = k) \propto \alpha_k$, and $\{G_k\}$ an independent and identically distributed sequence of standard Gumbel variables. Then $X = arg\max_k(log\alpha_k + G_k)$, that is, the Gumbel distribution provides a recipe for sampling from the categorical distribution.

For the Gumbel VAE ([117], [118]), define $p_\theta(z)$ as distributed from the categorical distribution consisting of K categories, with $1/K$ probability for selection (e.g. each category has the same selection probability). Then define

$$q(\mathbf{z_i}|\mathbf{x}) = Cat(K, \pi_k^i),$$

where $i \leq d$ indexes the latent variable.

Of course, One can sample from the above distribution (assuming known π_i^k) using the Gumbel technique. But the argmax is not differentiable. Therefore, we use the softmax as a continuous approximation, such that sampling takes the form

$$z_k^i = \frac{exp((log(\pi_k^i) + G_k^i))/\tau)}{\sum_{j=1}^k exp((log(\pi_j^i) + G_j^i)/\tau)}, \tag{6.6}$$

a sample from the Gumbel-SoftMax (or Concrete) distribution (density omitted, see [117]). The quantities $\{G_k^i\}_{i \leq d, k \leq K}$ are sampled i.i.d. from the standard Gumbel distribution.

Finally, we define the encoder $g : \mathbf{R}^n \to \mathbf{R}^{K \times d}$, where n is the dimension of the input space, K the number of categories, and d the number of latent variables. This encoder defines the log probabilities $\log(\pi_i^k)$ as functions of the input $\mathbf{x}$, taking the place of the g and h distributions in the previous characterization. The model can then be trained in the same fashion as the standard VAE.

The parameter τ (called the temperature) is of critical importance, as it defines the "sharpness" of the argmax approximation. In the limit as $\tau \to 0$, the softmax converges the argmax, while as $\tau \to \infty$, the softmax converges to the uniform distribution. For low values of τ, the gradients become very steep (approaching infinity in the limit), and have high variance. The typical approach is to start with a relatively large value of τ, and slowly decrease the value as training progresses (and the step size decreases). The parameter τ is typically lower

bounded during training, and during testing the argmax may be used directly, as gradient backpropagation through the model is not required.

6.1.2 Continuous Bernoulli VAE

The Continuous Bernoulli distribution takes support in the $[0, 1]$ interval and is based on a single parameter λ. The probability distribution function takes the form

$$p(x|\lambda) \propto \lambda^x (1-\lambda)^x.$$

The constant of proportionality is easily computed via standard calculus, but we note that it has a singularity at $\lambda = 1/2$, requiring approximation via Taylor series near $1/2$.

When $\lambda = 1/2$, the Continuous Bernoulli distribution simplifies to the uniform distribution on $[0, 1]$. For values greater than $1/2$, samples collected from the distribution have higher probability of being larger than $1/2$, and vice versa for $\lambda < 1/2$, which is useful for encoding and interpreting probabilities. Samples can be computed using the inverse CDF method, enabling the reparameterization trick.

The Continuous Bernoulli VAE [119] model is now constructed by specifying the approximate posterior and prior distributions

$$q_\phi(\mathbf{z}|\mathbf{x}) = CB(\mathbf{z}, \sigma(g(\mathbf{x})))$$
$$p_\theta(mathbf{z}) = CB(\mathbf{z}|\frac{1}{2}) = U(0, 1),$$

where σ denotes the sigmoid function, and g is an encoding neural network.

6.2 Neural network for probabilistic clustering

6.2.1 Model development

In the context of scRNA-seq analysis, dimensionality reduction and clustering form the most important preprocessing steps [120]. Clustering in this context identifies transcriptomic cell types based on the variation of gene expression between cells. Even though many attempts have been made to incorporate clustering into neural network models [121], the field generally relies on sequential dimensionality reduction based on PCA and UMAP (or tSNE), with clustering via community detection methods at the PCA stage [98]. This processing workflow is fast, reliable and interpretable, aspects which most machine learning models lack.

Most deep clustering algorithms require specification of the number of clusters *a priori* [121], which is virtually always an unknown. Community detection algorithms minimize an objective based on within/between cluster similarity, based on a resolution parameter that controls the number of clusters the data is divided into (this is similar to other density/community based algorithms such as DBSCAN [122] and Louvain [123]).

Here we present a neural network model for combined dimensionality reduction and probabilistic clustering. This model does not require specification of the number of clusters *a priori*, instead defining cluster granularity based on the dimensionality of the latent space.

In the standard Gumbel-softmax VAE, specify the number of categories for each latent variable to be two. This requires the following two functions

$$f_{encode} : \mathbb{R}^n \to \mathbb{R}^{d\times 2}$$
$$f_{decode} : \mathbb{R}^{d\times 2} \to \mathbb{R}^n,$$

where f_{encode} predicts log cluster probabilities, and f_{decode} reconstructs the input after sam-

pling in the latent dimensional space.

As the parameter $\tau \to 0$, this assigns each input one of two categories for the individual latent dimensions. We can therefore view this as a binary encoding of each point. We interpret each possible binary value as a cluster. Therefore, the number of possible clusters grows exponentially with the number of latent variables encoded in the model, which limits the required size of the latent dimensional space.

In practice, not every possible encoding has points assigned, and as is typical in the community detection algorithms, we assign outliers (e.g. clusters with few members) to the nearest cluster (in L_1 space).

We also develop an additional loss term that acts as a resolution parameter, limiting the number of clusters identified. This is defined as

$$L_{res} = \lambda \; mean((exp(\frac{0:d-1)}{\gamma} - 1) \cdot v_k)$$

$$v_k = mean_i\{\pi_k^i\}.$$

This loss function penalizes the distance between cluster probabilities and zero, producing an individual value for each latent variable. This is then scaled by an exponential term that penalizes probability variation as the index increases. The parameters λ, γ govern the weight penalty attached to each latent variable. This loss incentivizes a small $|\pi_k^i|$, with the incentive increasing with the index of the latent variable. In practice, the model is pretrained without this loss, and the latent variables are sorted so that those exhibiting the smallest variation are multiplied by the largest scale factor.

One benefit of this model is that the encoding explicitly defines clusters, but the outputs (for non-zero τ) are log probabilities. That is, this technique produces a soft-clustering. This reflects the biological reality, as in many instances, cells do not belong to distinct

clusters, instead existing on gradient trajectories. While this is most commonly associated with pseudotime analyses, many fully developed cells exist on transcriptional gradients, particularly in the brain.

6.2.2 Clustering single-cell spatial transcriptomics data

We first test the model on a set of simulated clusters (*sklearn.datasets.make_blobs*, data sampled from several 2D Gaussians). We verify that the model accurately classifies the clusters, and that lowering the resolution results in a reduced number of clusters (**Figure 6.1**).

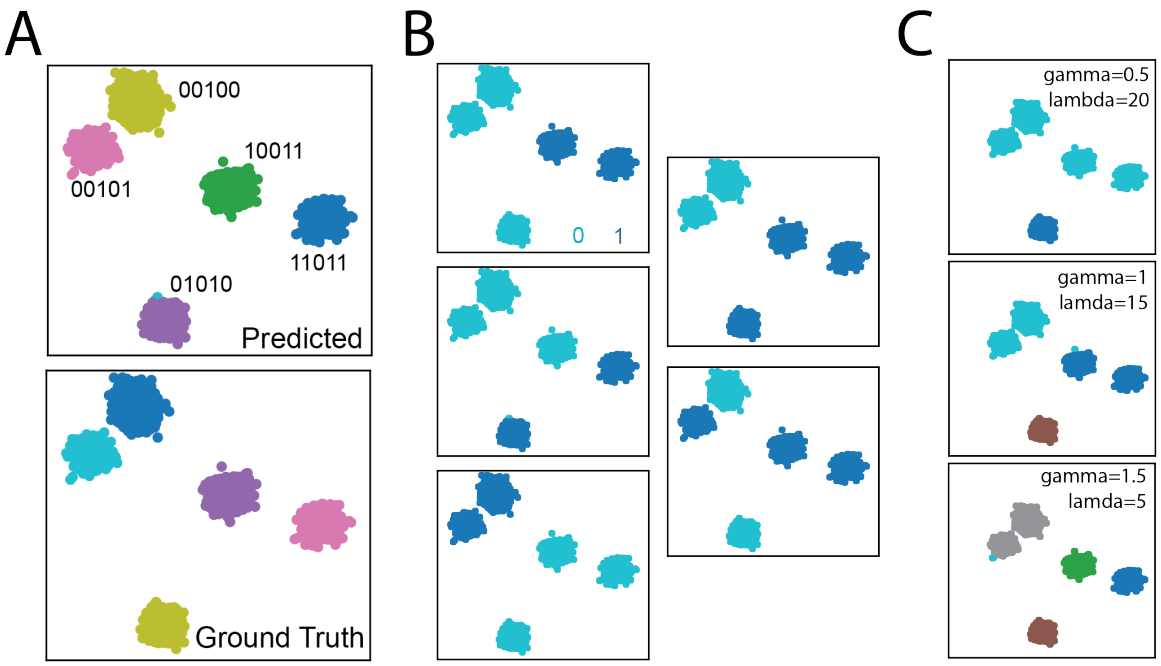

Figure 6.1: Neural network clustering results of simulated clusters. **A**: (top) Classes identified by neural network clustering, with associated binary encodings. (bottom) Ground truth clustering results. **B**: Cluster discrimination in each latent dimension. Points are labeled either 0 or 1 (as only 2 classes are available) and define the clustering result. **C**: Clusters when including the resolution loss (resolution loss added after 200 epochs) at various parameters.

We verify neural network clustering on a spatial transcriptomic (MERFISH) dataset with annotated celltype assignments[2], and compare it to the raw Louvain algorithm clusters via

[2]MERFISH spatial transcriptomics of whole coronal section produced by Dr. Zhiqun Tan, a member of

NMI/ARI (Normalized Mutual Information, Adjusted Rand Index). This dataset consists of 10,000 cells (after subsetting), with gene expression for 300 genes. We have labeled celltypes based on the cluster gene markers in the original PCA/louvain clustering, followed by an additional round of subclustering and cell type labeling. Labeling was based on the Allen scRNA-seq atlas [124]. We initially characterize cells into major cell groupings (e.g. Cortical Excitatory, Microglia, Astrocytes), which we compare with the neural network clustering results. This method detects and differentiates most major cell groups. ARI and NMI between neural network clustering and the denoted celltypes was 0.71 and 0.82 respectively (**Figure 6.2**).

Next, we subcluster the excitatory cortical neurons. This subset of cells is the most difficult to differentiate and label. ARI and NMI between labeled and neural network as 0.48 and 0.65 respectively, showing a significant reduction over the high level clustering (**Figure 6.2 B**). This is expected as cluster consistency decreases with cluster resolution, particularly when relatively few genes are profiled. The neural network model differentiates the L5 PT, L6b, and L5/L6 NP subgroups. It also identifies the hippocampal excitatory neurons, though the differentiation between CA1/CA3 is unclear (this is true both transcriptomically and spatially, as no absolute boundary between these celltypes exists. Both neural network and PCA/louvain clustering showed difficulty differentiating the IT subclusters across the cortical layers (indeed the original clustering utilized visual inspection of spatial location to identify the appropriate layers).

---

the Xu lab, using the Vizgen MERSCOPE

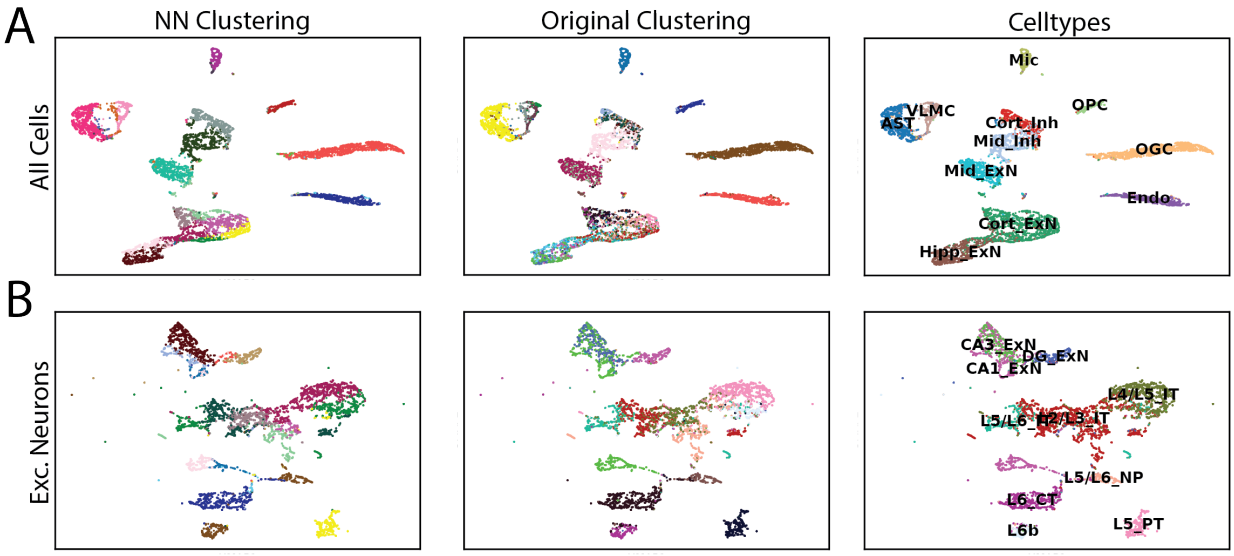

Figure 6.2: Comparisons of neural network clustering (left) PCA/louvain clustering (middle) and annotated cell types (right), using 10,000 randomly selected cells from full set of cells (**A**) and the excitatory cortical neurons (**B**). Computation was performed with 5 latent variables (4 for the excitatory neurons) for clustering, followed by UMAP of the latent projection for visualization.

6.3 scGradient: a neural network model for transcriptional gradient detection

6.3.1 Building on a combination of VAEs

Transcriptional gradients are 1-dimensional manifolds on which gene expression varies continuously [16]. This is similar to the concept of pseudotime [125]. However, pseudotime carries an inherent developmental connotation (e.g. cells develop along pseudotime trajectories), and the associated genes of interest vary in pseudotime. Pseudotime utilizes the idea that cells develop at different rates, which enables inference of developmental processes from a single sample [126]. However, neurons are generally at the end of their developmental trajectories, as neurogenesis in adults occurs in only a few regions [127]. Additionally, neurons can lie on multiple transcriptional gradients simultaneously, while developming cells typically lie on a single inferred pseudotime trajectory. General transcriptional gradients

occur in adult animals. For example, there are significant transcriptional gradients in the cortex [128], which vary spatially according to cortical depth.

We develop scGradient, a machine learning model for the detection of transcriptional gradients. This model is capable of detecting multiple uncorrelated gradients simultaneously, while also the also predicting the location of each cell on the detected gradients, as well as the probability that a cell lies on the given gradient.

Inspired by previous work utilizing VAE models used for pseudotime inference [129], we build scGradient on top of three variational autoencoders (**Figure 6.3 A-C**). First, the categorical VAE described earlier, which describes the cell types present in the data. Second, a Gaussian VAE (e.g. based on Gaussian priors as described originally) which represents a continual latent encoding in a low dimensional space. Third, a Continuous Bernoulli VAE with an N-dimensional latent variable for encoding probabilities (each dimension of the CB latent variable encodes the probabilities that a given cell lies on the specified gradient).

In addition to probability, category, and continuous latent variables, scGradient also predicts the gradient time ($t \in [0,1]^N$) for each cell. This represents a cell's location on each inferred gradient. We utilize a neural ordinary differential equation to relate the continuous latent embedding and gradient time. This is an ODE of the form

$$\frac{d\mathbf{z}}{dt} = f_{ANN}(t^i, \mathbf{z}),$$

where f_{ANN} is constrained to be a neural network, and $\mathbf{z}$ is the latent variable encoding. The weights of the neural ode can be trained using the adjoint method [130], allowing the neural ode to be learned during training. Learning occurs by computing the loss between latent variables and their ODE predictions. The latent $\mathbf{z}_{pred}$ embedding is concatenated with the ODE predictions prior to decoding.

We define a separate f^i_{ANN} for all $1 \leq i \leq N$. This enables detection of multiple gradients simultaneously. There is some question as to whether using a common continuous latent embedding for all ODEs is acceptable, and we have considered models with separate Gaussian VAEs for each Neural ODE, though this significantly increases the number of parameters.

We initially constrained the Neural ODEs to be constant in the continuous latent space (e.g. $\frac{d\mathbf{z}}{dt} = \mathbf{const}$), for some learnable constant **const** to improve computational speed. However, a constant derivative in the latent space does not transform to a constant derivative in transcriptome space. Therefore, we define $\frac{d\mathbf{x}}{dt} = const$, where $\mathbf{x}$ denotes (normalized) gene expression. We then compute $\frac{d\mathbf{z}}{dt}$ via the chain rule. We utilize Euler's method for solving the ODEs [131], and the standard adjoint method for learning Neural ODE weights, which enables inference of the associated constants in expression space.

As no initial point for the ODE is generally present, we sample a point based on the probabilities that each cell is on the given gradient, and predict backwards and forwards in time based on the sampled point and the predicted time values.

Finally, we construct the following combined loss function.

$$
\begin{aligned}
L = {} & \alpha_1 D^{cat}_{KL}(p_{cat}(\mathbf{z}|\mathbf{x}), Cat(K, 1/K)) + \alpha_2 D^{lat}_{KL}(p_{lat}(\mathbf{z}|\mathbf{x}), \mathcal{N}(\mathbf{0}, \mathbf{I})) \\
& + \alpha_3 D^{prob}_{KL}(p_{prob}(\mathbf{z}|\mathbf{x}), \text{Unif}(0,1)) + \alpha_4 ||\mathbf{x} - \hat{\mathbf{x}}_{cat}||^2_2 \\
& + \alpha_5 ||\mathbf{x} - \hat{\mathbf{x}}_{lat}||^2_2 + \alpha_6 (p_{prob}(\mathbf{z}|\mathbf{x}) \odot \sigma(||\mathbf{z}_{lat} - \mathbf{z}_{pred}||^2_2) \\
& + (1 - p_{prob}(\mathbf{z}|\mathbf{x})) \odot (1 - \sigma(||\mathbf{z}_{lat} - \mathbf{z}_{pred}||^2_2))) \\
& + \alpha_7 \text{mean}(\max(\mathbf{0}, 1 - p_{prob}(\mathbf{x})(\mathbf{x}), (\text{dim } 1))) \\
& + \alpha_8 (\max(0, 1 - \text{mean}(\max_k p_{encode}(\mathbf{x}), (\text{dim } 2)))) + \alpha_9 \text{corrcoef}(t) \\
& + \alpha_{10} \text{mean}_i(\text{rep}(p^i_{prob}) \odot \text{pairwise}_{L1}(p_{cat})).
\end{aligned}
$$

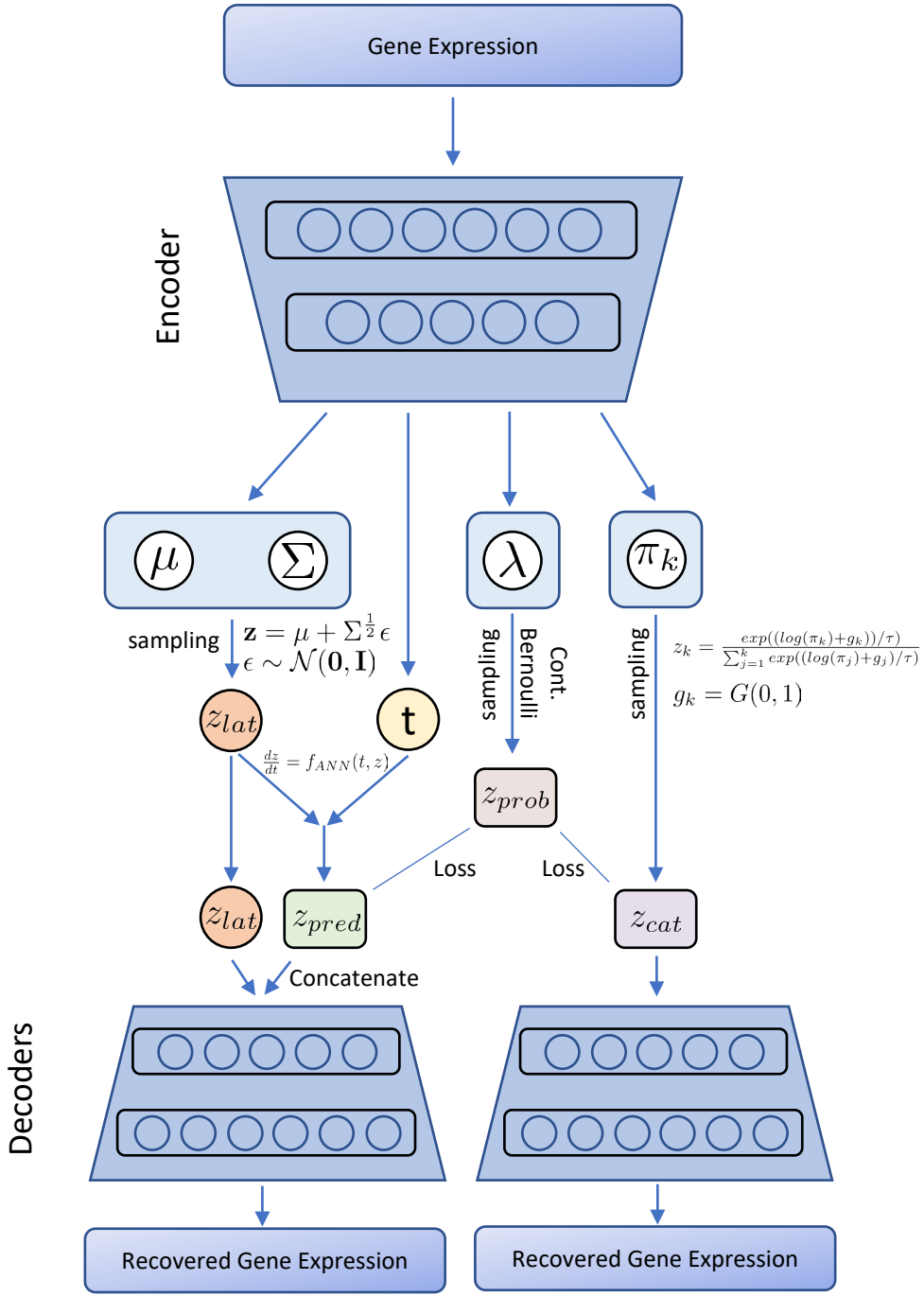

Figure 6.3: scGradient workflow. **A**: scGradient is build on three variational autoencoders. Gene expression is transformed into latent space via an encoder. **B**: Low dimensional outputs are computed and sampled from the Gaussian, Continuous Bernoulli, and Categorical VAEs, as well as the time component for the neural ODE, resulting in samples z_{lat}, z_{prob}, and z_{cat} respectively. Predictions based on the ODE are generated via Euler's method. The predicted probabilities weight error between latent and ODE prediction variables, as well as the latent categorical variables. **C**: Separate decoders are used for the categorical and Gaussian embeddings, to enable computation of the pushforward of the ODE derivatives.

The first three elements of the loss function are the standard KL regularization for each of the three VAEs (the subscripts *cat*, *lat*, and *prob* correspond to the categorical Gumbel-softmax, Gaussian and Continuous Bernoulli predictions). The next two terms correspond to reconstruction error for the Gumbel-softmax and Gaussian VAEs (the probabilities are not reconstructed to predict gene expression). The next term weights the error between the latent variables ($\mathbf{z}$) and the ODE predictions. High probabilities should be matched with low error, and low probabilities with high error. We apply the sigmoid function σ to the ODE error to map to $[0, 1]$. As setting the weights p_{encode} to zero and the error to one minimizes this term, we include two counterbalances $(\text{mean}(\max(\mathbf{0}, 1 - p_{prob}(\mathbf{x})(\mathbf{x}), (\text{dim } 1))), \text{mean}(\max(0, 1 - \max_k p_{encode}(\mathbf{x}), (\text{dim } 2))))$ where the dimension indicates which dimension to take the maximum. These counterbalances ensure at least one cell is assigned to each gradient, and that each cell is assigned to at least one gradient. Penalizing the correlation between time points, prompts discovery of independent transcriptional gradients. The final term penalizes transcriptional gradients across too many cell types. The *rep* function repeats the probability vector (for a given ODE, indicated by the superscript) to create a square matrix, which is multiplied by the pairwise L_1 distance between categorical embeddings. Therefore, high probabilities for cells with low L_1 distance between the categorical embeddings are encouraged.

Determining appropriate weights α_j is largely a matter of trial and error. Typically, reconstruction error gets higher weight than KL error (to ensure model flexibility). The counterbalancing error terms for probability are weighted highly, as is the error term relating probability and ODE prediction accuracy.

Interpreting the results involves two factors. First, the amount of error between the decoded latent variables and gene expression (imputed using MAGIC), and the magnitude of the learned constant ODE term for the associated gene. From this we can determine which genes are exhibiting the largest variation along the gradient trajectory.

6.3.2 Detecting gradients in spatial transcriptomics data

We first apply scGradient to detect transcriptional gradients on the subset of excitatory neurons of the spatial transcriptomics dataset used for clustering[3] Subsetting to 3,000 cells, we pretrain the model for clustering for 500 epochs, and then train the neural ODEs for 300 iterations. A total of 3 dimensions were used for clustering, 5 for the latent representation, and 4 corresponding to the ODEs (e.g. we can detect a maximum of 4 distinct transcriptional gradients).

On this dataset, only one transcriptional gradient was detected. The correlation between (gradient) time points for the separate ODEs was over 0.99 for each pairwise comparison. Neural ODEs were defined as $\frac{dz}{dt} = const$, and the average variance of the learned constants across dimensions was 0.003.

We first compute the pairwise RMSE error between the reconstructions (e.g. reconstructed to PCA space by the neural network and then transformed via low rank PCA approximation back to the original count space) with the raw counts, and with imputed counts (via MAGIC). This results in pairwise RMSEs of 0.657 ± 0.226, 0.144 ± 0.137 and 0.625 ± 0.208, for reconstruction vs raw, reconstruction vs imputed, and raw vs imputed respectively (**Figure 6.4 A**). Thus reconstruction error was significantly closer to the imputed result than the raw data ($p = 2.5 \times 10^{-140}$) implying that scGradient denoises the raw transcripts.

we next compute the pushforward of the derivative (of the first ODE, since they all represent the same gradient), to obtain the approximate variation of counts with respect to gradient time. we identify genes of interest using the following formula

$$abs(mean(deriv)),$$

[3]Results were computed using the initial method where $\frac{d\mathbf{z}}{dt} = const$. Both methods identified a similar subset of neurons on the excitatory dataset including Slc30a3 and Nptxr. Inferred transcriptional gradients were computed via pushforward of $\frac{d\mathbf{z}}{dt}$

where *deriv* is the pushforward of the derivative (for a single gene). Taking the top 11 ranked genes (identified via a threshold of 2 for the average derivative), we compute the average R^2 (coefficient of determination) based on a linear model, between the reconstructed data and the impute gene expression. This yields 0.35 ± 0.17. The score used to identify genes was correlated with R^2 (correlation coefficient 0.57).

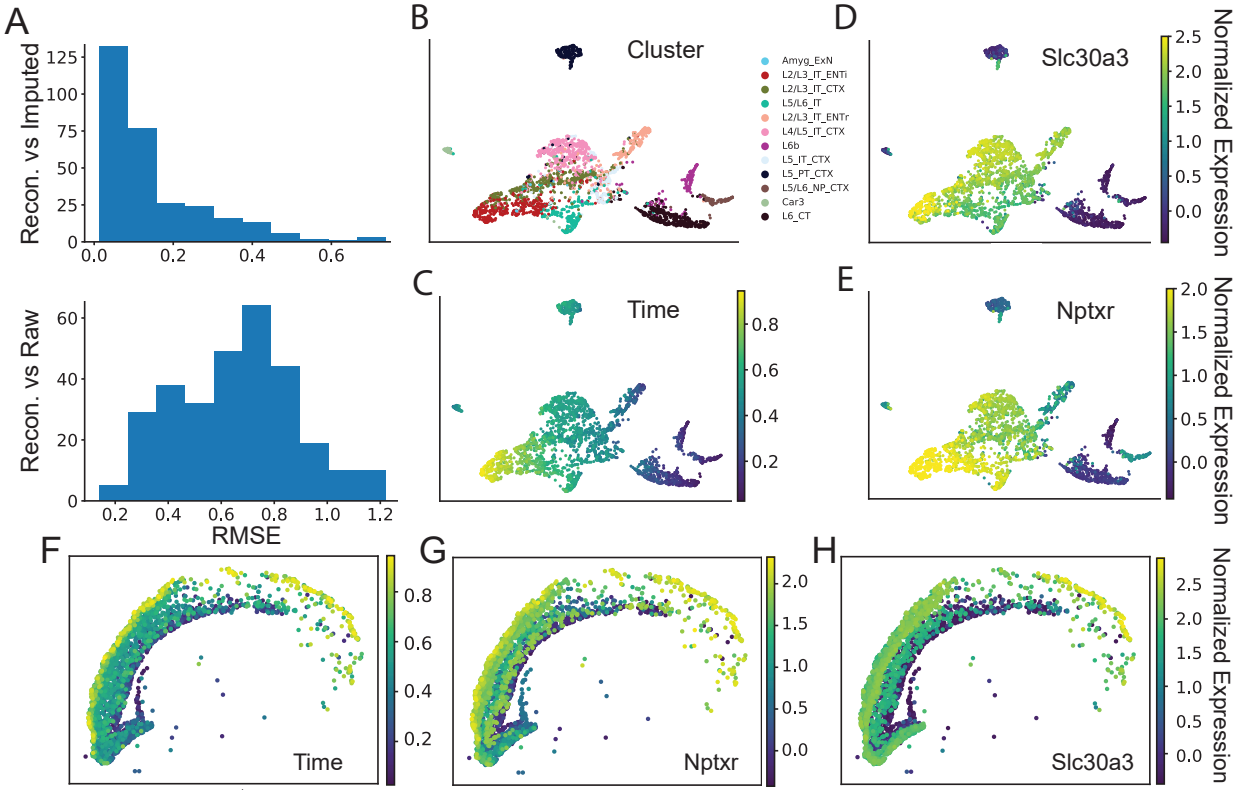

Figure 6.4: Detected transcriptomic gradient in excitatory neurons. **A**: Histogram of RMSE error between reconstructed gene expression and imputed gene expression (top) and reconstructed gene expression and raw expression (bottom). **B-E** represent data on a UMAP plot. **B**: Celltype labels for detected clusters (computed *a priori*). **C**: Computed position on $[0, 1]$ gradient time interval for each cell based on the Neural ODE. **D-E**: Imputed gene expression for Slc30a3 and Nptxr. **F-H**: Spatial representation of gradient time and imputed gene expression for Slc30a3 and Nptxr.

In this case, the transcriptional gradient (**Figure 6.4 B-E**) detected corresponds to a well known gradient corresponding to cortical depth [128], with the largest variation within the IT (Intratelencephalic Neurons, **Figure 6.4 F-H**). Among the top ranked genes, Nptx2 and Nptxr are associated with regulating excitatory synapse formation [132],[133]. Slc30a3

is associated with Zinc capture in the synaptic vesicle, Wfs1 regulates calcium intake [134], required for neural signalling, and Syngap1 regulates post-synaptic currents [135]. Lamp5 relates to synaptic plasticity in cortical excitatory neurons [136]. The remaining detected genes exhibited low R^2 coefficients. These detected genes are integral in neuronal signaling, allowing us to hypothesize that variation along this transcriptional gradient is associated with variations in neuron signaling, particularly across the cortex. We also note that several of the detected genes play significant roles in IT subtypes specifically, which project across the cortex. Nptxr in particular has been linked to mirror movements, a disorder causing contralateral identical movements.

We apply a similar analysis with the PV (Parvalbumin expressing) interneurons in the same spatial dataset. This dataset involved 842 cells. We identified two primary transcriptional gradients across the set, utilizing four ODEs. Reconstruction error compared with MAGIC imputed gene expression was again low (0.065 ± 0.056) compared with raw gene expression (0.321 ± 0.116). We identify a total of 13 genes associated with the first transcriptional gradient, and 14 genes associated with the second, based on the same criteria and thresholds used previously.

Mean R^2 coefficient is 0.271 ± 0.194 for the first gradient and $.107 \pm 0.098$, indicating somewhat lower fidelity between the proposed gradients and imputed gene expression, particularly for the second gradient (the maximum R^2 values were 0.639 and 0.378, for first and second gradient respectively). The top genes exhibiting greatest R^2 coefficient were Pvalb and Gria1 for the first and second gradient respectively. Unfortunately, no genes in the Cox or Rpl/Rps families were profiled in this spatial transcriptomics dataset, preventing a full comparison with the results in Chapter 5. However Pvalb was highly correlated (0.456 after imputation) with Cox6a2, providing evidence that we have uncovered the previously detected gradient. The genes primarily identified with the first transcriptional gradient (Pvalb, Rph3a, **Figure 6.5 A-B**) play roles in calcium binding in interneurons [137], [138]. The genes identified

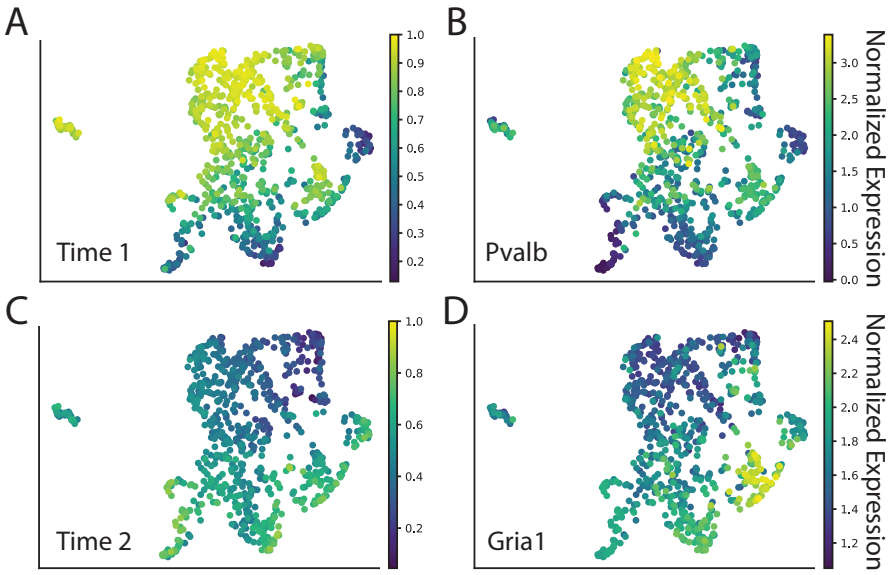

Figure 6.5: Transcriptional gradients for PV interneurons. **A**: Time scale for transcriptional gradient 1. **B**: Imputed Pvalb expression, which exhibits the highest correlation with gradient 1. **C**: Time scale for transcriptional gradient 2. **D**: Imputed Gria1 expression, which exhibits the highest correlation with gradient 1.

with the second transcriptional gradient (Adgra1, Gria1, **Figure 6.5 C-D**) are associated with neuronal signaling in excitatory neurons [139].

6.3.3 Current and future steps

At this point, the fundamental method for detecting transcriptional gradients is developed and validated. However, several issues remain.

The computed gradient time and probabilities exhibit some problematic aspects. Even after penalizing correlation between gradient time components for separate ODEs, there is a strong tendency for each ODE to learn the same gradient. We are currently addressing this issue by including separate latent embeddings for each ODE, however the result may be also be biological (e.g. there may be only one or two transcriptional gradient in the associated cell types). The computed probabilities also have a strong tendency toward assigning all cells to each gradient as cells with zero expression of genes strongly associated to the tran-

scriptional gradient exhibit strong probability scores. This is mathematically acceptable but needs to be adjusted to improve interpretation (e.g. by requiring some latent expression for inclusion within the gradient). Additionally, Neural ODEs require long training times. We are currently applying a downsampling approach to speed up training time.

While some technical issues remain, scGradient has successfully identified a known transcriptional gradient in the excitatory neurons, and several gradients, including the gradient identified in Chapter 5, in the Pvalb inhibitory neuron cell type. We will next apply this method to additional datasets and cell types for validation purposes, followed by exploration of novel gradients identified in additional cell types.

Bibliography

[1] Tsai-Wen Chen, Trevor J Wardill, Yi Sun, Stefan R Pulver, Sabine L Renninger, Amy Baohan, Eric R Schreiter, Rex A Kerr, Michael B Orger, Vivek Jayaraman, et al. Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*, 499(7458):295–300, 2013.

[2] Fuchou Tang, Catalin Barbacioru, Yangzhou Wang, Ellen Nordman, Clarence Lee, Nanlan Xu, Xiaohui Wang, John Bodeau, Brian B Tuch, Asim Siddiqui, et al. mrna-seq whole-transcriptome analysis of a single cell. *Nature methods*, 6(5):377–382, 2009.

[3] GX Zheng, Jessica M Terry, Phillip Belgrader, Paul Ryvkin, Zachary W Bent, Ryan Wilson, Solongo B Ziraldo, Tobias D Wheeler, Geoff P McDermott, Junjie Zhu, et al. Massively parallel digital transcriptional profiling of single cells. *Nature communications*, 8:14049, 2017.

[4] Stephen J Clark, Heather J Lee, Sébastien A Smallwood, Gavin Kelsey, and Wolf Reik. Single-cell epigenomics: powerful new methods for understanding gene regulation and cell identity. *Genome biology*, 17(1):1–10, 2016.

[5] Jeffrey R Moffitt, Junjie Hao, Guiping Wang, Kok Hao Chen, Hazen P Babcock, and Xiaowei Zhuang. High-throughput single-cell gene-expression profiling with multiplexed error-robust fluorescence in situ hybridization. *Proceedings of the National Academy of Sciences*, 113(39):11046–11051, 2016.

[6] Hongshan Guo, Ping Zhu, Xinglong Wu, Xianlong Li, Lu Wen, and Fuchou Tang. Single-cell methylome landscapes of mouse embryonic stem cells and early embryos analyzed using reduced representation bisulfite sequencing. *Genome research*, 23(12):2126–2135, 2013.

[7] Christoph Bock, Matthias Farlik, and Nathan C Sheffield. Multi-omics of single cells: strategies and applications. *Trends in biotechnology*, 34(8):605–608, 2016.

[8] Allen institute for brain science awarded approximately 100 million dollars to lead global collaborations that will build comprehensive atlases of brain cell types, Oct 2017.

[9] E. M. et al. Callaway. A multimodal cell census and atlas of the mammalian primary motor cortex. *Nature*, 598:86–102, 2021.

[10] Danielle S Bassett and Michael S Gazzaniga. Understanding complexity in the human brain. *Trends in cognitive sciences*, 15(5):200–209, 2011.

[11] Liqun Luo. *Principles of neurobiology*. Garland Science, 2020.

[12] Carsen Stringer, Marius Pachitariu, Nicholas Steinmetz, Matteo Carandini, and Kenneth D Harris. High-dimensional geometry of population responses in visual cortex. *Nature*, 571(7765):361–365, 2019.

[13] Daniel Aharoni and Tycho M Hoogland. Circuit investigations with open-source miniaturized microscopes: past, present and future. *Frontiers in cellular neuroscience*, 13:141, 2019.

[14] Haojia Wu, Yuhei Kirita, Erinn L Donnelly, and Benjamin D Humphreys. Advantages of single-nucleus over single-cell rna sequencing of adult kidney: rare cell types and novel cell states revealed in fibrosis. *Journal of the American Society of Nephrology*, 30(1):23–32, 2019.

[15] Trygve E Bakken, Rebecca D Hodge, Jeremy A Miller, Zizhen Yao, Thuc Nghi Nguyen, Brian Aevermann, Eliza Barkan, Darren Bertagnolli, Tamara Casper, Nick Dee, et al. Single-nucleus and single-cell transcriptomes compared in matched cortical cell types. *PloS one*, 13(12):e0209648, 2018.

[16] Bosiljka Tasic, Zizhen Yao, Lucas T Graybuck, Kimberly A Smith, Thuc Nghi Nguyen, Darren Bertagnolli, Jeff Goldy, Emma Garren, Michael N Economo, Sarada Viswanathan, et al. Shared and distinct transcriptomic cell types across neocortical areas. *Nature*, 563(7729):72–78, 2018.

[17] Marcela Lipovsek, Cedric Bardy, Cathryn R. Cadwell, Kristen Hadley, Dmitry Kobak, and Shreejoy J. Tripathy. Patch-seq: Past, present, and future. *Journal of Neuroscience*, 41(5):937–946, 2021.

[18] Federico Scala, Dmitry Kobak, Matteo Bernabucci, Yves Bernaerts, Cathryn René Cadwell, Jesus Ramon Castro, Leonard Hartmanis, Xiaolong Jiang, Sophie Laturnus, Elanine Miranda, et al. Phenotypic variation of transcriptomic cell types in mouse motor cortex. *Nature*, 598(7879):144–150, 2021.

[19] Kevin G Johnston, Steven F Grieco, Hai Zhang, Suoqin Jin, Xiangmin Xu, and Qing Nie. Tracking longitudinal population dynamics of single neuronal calcium signal using scout. *Cell Reports Methods*, 2(5):100207, 2022.

[20] Yanjun Sun, Suoqin Jin, Xiaoxiao Lin, Lujia Chen, Xin Qiao, Li Jiang, Pengcheng Zhou, Kevin G Johnston, Peyman Golshani, Qing Nie, et al. Ca1-projecting subiculum neurons facilitate object–place learning. *Nature neuroscience*, 22(11):1857–1870, 2019.

[21] Flusberg, B. A. and Nimmerjahn, A. and Cocker, E. D. and Mukamel, E. A. and Barretto, R. P. and Ko, T. H. and Burns, L. D. and Jung, J. C. and Schnitzer, M. J. High-speed, miniaturized fluorescence microscopy in freely moving mice. *Nature Methods*, 5(11):935–938, 2008.

[22] Ghosh, K. K. and Burns, L. D. and Cocker, E. D. and Nimmerjahn, A. and Ziv, Y. and El Gamal, A. and Schnitzer, M. J. Miniaturized integration of a fluorescence microscope. *Nature Methods*, 8(10):871–878, 2011.

[23] Y. Ziv and K. K. Ghosh. Miniature microscopes for large-scale imaging of neuronal activity in freely behaving rodents. *Current opinion in neurobiology*, 32:141– 147, 2015.

[24] Cai, D. J., Aharoni, D., Shuman, T., Shobe, J., Biane, J., Song, W., Wei, B., Veshkini, M., La-Vu, M., Lou, J., et al. A shared neural ensemble links distinct contextual memories encoded close in time. *Nature*, 534(7605):115–118, 2016.

[25] Ziv, Y. and Burns, L. D. and Cocker, E. D. and Hamel, E. O. and Ghosh, K. K. and Kitch, L. J. and El Gamal, A., and Schnitzer, M. J. Long-term dynamics of ca1 hippocampal place codes. *Nature Neuroscience*, 16(3):264–266, 2013.

[26] Jimenez, J., Goldberg, A., Ordek, G., Luna, V., Su, K., Pena, S., Zweifel, L., Hen, R., and Kheirbek, M. Subcortical projection-specific control of innate anxiety and learned fear by the ventral hippocampus. Neuroscience meeting planner, Society for Neuroscience, San Diego, 2016.

[27] Rubin, A., Geva, N., Sheintuch, L., and Ziv, Y. Hippocampal ensemble dynamics timestamp events in long-term memory. *Elife*, 4:e12247, 2015.

[28] Sun, Y., Jin, S., Lin, X., Chen, L., Qiao, X., Jiang, L., Zhou, P., Johnston, K., Golshani, P., Nie, Q., Holmes, T., Nitz, D., Xu, X. Ca1-projecting subiculum neurons facilitate object–place learning. *Nature Neuroscience*, 54(10):1546–1726, 2019.

[29] Kitamura, T., Sun, C., Martin, J., Kitch, L. J., Schnitzer, M. J., Tonegawa, S. Entorhinal cortical ocean cells encode specific contexts and drive context specific fear memory. *Neuron*, 87(6):1317–1331, 2015.

[30] Sun, C., Kitamura, T., Yamamoto, J., Martin, J., Pignatelli, M., Kitch, L. J., Schnitzer, M. J., and Tonegawa, S. Distinct speed dependence of entorhinal island and ocean cells, including respective grid cells. *Proceedings of the National Academy of Sciences*, 112(30):9466–9471, 2015.

[31] Barbera, G., Liang, B., Zhang, L., Gerfen, C. R., Culurciello, E., Chen, R., Li, Y., and Lin, D.T. Spatially compact neural clusters in the dorsal striatum encode locomotion relevant information. *Neuron*, 92(1):202–213, 2016.

[32] Klaus, A., Martins, G.J., Paixao, V.B., Zhou, P., Paninski, L. and Costa, R.M. The spatiotemporal organization of the striatum encodes action space. *Neuron*, 95(5):1171–1180, 2017.

[33] Yu, K., Ahrens, S., Zhang, X., Schiff, H., Ramakrishnan, C., Fenno, L., Deisseroth, K., Zhao, F., Luo, M.H., Gong, L. and He, M. The central amygdala controls learning in the lateral amygdala. *Nature Neuroscience*, 20(12):1680–1685, 2017.

[34] Apthorpe, N., Riordan, A., Aguilar, R., Homann, J., Gu, Y., Tank, D., and Seung, H. S. Automatic neuron detection in calcium imaging data using convolutional networks. *Advances in Neural Information Processing Systems*, page 3270–3278, 2016.

[35] Mukamel, E. A., Nimmerjahn, A., and Schnitzer, M. J. Automated analysis of cellular signals from large-scale calcium imaging data. *Neuron*, 63(6):747–760, 2009.

[36] Pnevmatikakis, E. A., Soudry, D., Gao, Y., Machado, T. A., Merel, J., Pfau, D., Reardon, T., Mu, Y., Lacefield, C., Yang, W., et al. Simultaneous denoising, deconvolution, and demixing of calcium imaging data. *Neuron*, 89(2):285–299, 2016.

[37] S.L. Smith and M. Häusser. Parallel processing of visual space by neighboring neurons in mouse visual cortex. *Nature neuroscience*, 13(9):1144–1149, 2010.

[38] Zhou, P., Resendez, S. L., Rodriguez-Romaguera, J., Jimenez, J. C., Neufeld, S. Q., Giovannucci, A., Friedrich, J., Pnevmatikakis, E. A., Stuber, G. D., Hen, R., Kheirbek, M. A., Sabatini, B. L., Kass, R. E., and Paninski, L. Efficient and accurate extraction of in vivo calcium signals from microendoscopic video data. *elife*, 7:e28728, 2018.

[39] E. A. Pnevmatikakis and A. Giovannucci. Normcorre: an online algorithm for piecewise rigid motion correction of calcium imaging data. *Journal of neuroscience methods*, 291:83–94, 2017.

[40] Giovannucci, A., Friedrich, J., Gunn, P., Kalfon, J., Brown, B.L., Koay, S.A., Taxidis, J., Najafi, F., Gauthier, J.L., Zhou, P. and Khakh, B.S. Caiman an open source tool for scalable calcium imaging data analysis. *Elife*, 8:e38173, 2019.

[41] S. Kullback. *Information theory and statistics*. Courier Corporation, 1997.

[42] Sheintuch, L., Rubin, A., Brande-Eilat, N., Geva, N., Sadeh, N., Pinchasof, O., and Ziv, Y. Tracking the same neurons across multiple days in ca2+ imaging data. *Cell Reports*, 21(4):1102–1115, 2017.

[43] The Mathworks, Inc., Natick, Massachusetts. *MATLAB version (R2021a)*, 2021.

[44] J. C. Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, 3(3):32–57, 1973.

[45] Grieco, S. F., Qiao, X., Johnston, K. G., Chen, L., Nelson, R. R., Lai, C., Holmes, T., Xu, X. Neuregulin signaling mediates the acute and sustained antidepressant effects of subanesthetic ketamine. *Translational psychiatry*, 11(1):1–17, 2021.

[46] Grieco, S. F., Qiao, X., Zheng, X., Liu, Y., Chen, L., Zhang, H., Yu, Z., Gavornik, J., Lai, C., Gandhi, S., Holmes, T., Xu, X. Subanesthetic ketamine reactivates adult cortical plasticity to restore vision from amblyopia. *Current Biology*, 30(18):3591–3603, 2020.

[47] Allen Institute for Brain Science. Allen Brain Observatory. `https://registry.opendata.aws/allen-br0ain-observatory/`, 2016.

[48] Grijseels, D. M., Shaw, K., Barry, C., Hall, C. N. Choice of method of place cell classification determines the population of cells identified. *PLoS computational biology*, 17(7):e1008835, 2021.

[49] David J Bucci, Michael P Saddoris, and Rebecca D Burwell. Contextual fear discrimination is impaired by damage to the postrhinal or perirhinal cortex. *Behavioral neuroscience*, 116(3):479, 2002.

[50] Jennifer Czerniawski and John F Guzowski. Acute neuroinflammation impairs context discrimination memory and disrupts pattern separation processes in hippocampus. *Journal of Neuroscience*, 34(37):12470–12480, 2014.

[51] John Bryden. Biologically inspired computing: The neural network. *URL https://www. academia. edu/35380032/BiologicallyI nspiredC omputingT heN euralN etwork*, 2014.

[52] DO Hebb. Organization of behavior. new york: Wiley. *J. Clin. Psychol*, 6(3):335–307, 1949.

[53] Amirhossein Tavanaei, Masoud Ghodrati, Saeed Reza Kheradpisheh, Timothée Masquelier, and Anthony Maida. Deep learning in spiking neural networks. *Neural networks*, 111:47–63, 2019.

[54] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015.

[55] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

[56] Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International conference on machine learning*, pages 1675–1685. PMLR, 2019.

[57] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[58] Guodong Zhang, Chaoqi Wang, Bowen Xu, and Roger Grosse. Three mechanisms of weight decay regularization. *arXiv preprint arXiv:1810.12281*, 2018.

[59] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[60] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[61] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[62] Grace W Lindsay. Convolutional neural networks as a model of the visual system: Past, present, and future. *Journal of cognitive neuroscience*, 33(10):2017–2031, 2021.

[63] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[64] Delia Silva, Ting Feng, and David J Foster. Trajectory events across hippocampal place cells require previous experience. *Nature neuroscience*, 18(12):1772–1779, 2015.

[65] Ardi Tampuu, Tambet Matiisen, H Freyja Ólafsdóttir, Caswell Barry, and Raul Vicente. Efficient neural decoding of self-location with a deep recurrent network. *PLoS computational biology*, 15(2):e1006822, 2019.

[66] Davide Chicco, Matthijs J Warrens, and Giuseppe Jurman. The coefficient of determination r-squared is more informative than smape, mae, mape, mse and rmse in regression analysis evaluation. *PeerJ Computer Science*, 7:e623, 2021.

[67] Per Andersen. Organization of hippocampal neurons and their interconnections. In *The hippocampus*, pages 155–175. Springer, 1975.

[68] Caroline Le Duigou, Jean Simonnet, Maria T Teleñczuk, Desdemona Fricker, and Richard Miles. Recurrent synapses and circuits in the ca3 region of the hippocampus: an associative network. *Frontiers in cellular neuroscience*, 7:262, 2014.

[69] Xiangmin Xu, Yanjun Sun, Todd C Holmes, and Alberto J López. Noncanonical connections between the subiculum and hippocampal ca1. *Journal of Comparative Neurology*, 524(17):3666–3673, 2016.

[70] Bryan L Roth. Dreadds for neuroscientists. *Neuron*, 89(4):683–694, 2016.

[71] Haotian Zhang. The review of transcriptome sequencing: principles, history and advances. In *IOP Conference Series: Earth and Environmental Science*, volume 332, page 042003. IOP Publishing, 2019.

[72] Thomas E Chater and Yukiko Goda. The role of ampa receptors in postsynaptic mechanisms of synaptic plasticity. *Frontiers in cellular neuroscience*, 8:401, 2014.

[73] Sven Schenk, Stephanie C Bannister, Fritz J Sedlazeck, Dorothea Anrather, Bui Quang Minh, Andrea Bileck, Markus Hartl, Arndt von Haeseler, Christopher Gerner, Florian Raible, et al. Combined transcriptome and proteome profiling reveals specific molecular brain signatures for sex, maturation and circalunar clock phase. *Elife*, 8:e41556, 2019.

[74] Amir Segev, Francisco Garcia-Oscos, and Saïd Kourrich. Whole-cell patch-clamp recordings in brain slices. *JoVE (Journal of Visualized Experiments)*, (112):e54024, 2016.

[75] H. et al. Markram. Interneurons of the neocortical inhibitory system. *Nature Review Neuroscience*, 5:793–807, 2004.

[76] M. A. Whittington and R. D. Traub. Interneuron diversity series: inhibitory interneurons and network oscillations in vitro. *Trends in neurosciences*, 26:676–682, 2003.

[77] C. J. McBain and A. Fisahn. Interneurons unbound. *Nature Review Neuroscience*, 2:11–23, 2001.

[78] O. Marin. Interneuron dysfunction in psychiatric disorders. *Nature Review Neuroscience*, 13:107–120, 2012.

[79] Y. Kawaguchi and S. Kondo. Parvalbumin, somatostatin and cholecystokinin as chemical markers for specific gabaergic interneuron types in the rat frontal cortex. *Journal of Neurocytology*, 31:277–287, 2002.

[80] Y. Kubota and Y. Kawaguchi. Two distinct subgroups of cholecystokinin-immunoreactive cortical interneurons. *Brain Research*, 752:175–183, 1997.

[81] Xu, X. and Roby, K. D. and Callaway, E. M. Immunochemical characterization of inhibitory mouse cortical neurons: three chemically distinct classes of inhibitory cells. *Journal of Comparative Neurology*, 518:389–404, 2010.

[82] E. M. Callaway. *Micro-, Meso- and Macro-Connectomics of the Brain*. Springer, 2016.

[83] Haettig, J. and Sun, Y. and Wood, M. A. and Xu, X. Cell-type specific inactivation of hippocampal ca1 disrupts location-dependent object recognition in the mouse. *Learning and Memory*, 20:139–146, 2013.

[84] Y. et al. Sun. Cell-type-specific circuit connectivity of hippocampal ca1 revealed through cre-dependent rabies tracing. *Cell Reports*, 7:269–280, 2014.

[85] T. L. et al. Daigle. A suite of transgenic driver and reporter mouse lines with enhanced brain-cell-type targeting and functionality. *Cell*, 174:465–480, 2018.

[86] Xu, X. and Roby, K. D. and Callaway, E. M. Mouse cortical inhibitory neuron type that coexpresses somatostatin and calretinin. *Journal of Comparative Neurology*, 499:144–160, 2006.

[87] Vanderhaeghen, J. J., Signeau, J. C., Gepts, W. New peptide in the vertebrate cns reacting with antigastrin antibodies. *Nature*, 257:604–605, 1975.

[88] Beinfeld, M. C., Meyer, D. K., Eskay, R. L., Jensen, R. T. and Brownstein, M. J. The distribution of cholecystokinin immunoreactivity in the central nervous system of the rat as determined by radioimmunoassay. *Brain Research*, 212:51–57, 1981.

[89] Gallopin, T., Geoffroy, H., Rossier, J. and Lambolez, B. Cortical sources of crf, nkb, and cck and their effects on pyramidal cells in the neocortex. *Cerebral Cortex*, 16:1440–1452, 2006.

[90] Toledo-Rodriguez, M., Goodman, P., Illic, M., Wu, C. and Markram, H. Neuropeptide and calcium-binding protein gene expression profiles predict neuronal anatomical type in the juvenile rat. *Journal of Physiology*, 567:401–413, 2005.

[91] B. et al. Dudok. Alternating sources of perisomatic inhibition during behavior. *Neuron*, 2021.

[92] D. A., Curley, A. A., Glausier, J. R. , Volk, D. W. Cortical parvalbumin interneurons and cognitive dysfunction in schizophrenia. *Trends in Neurosciences*, 35:57–67, 2012.

[93] Z. et al. Yao. A taxonomy of transcriptomic cell types across the isocortex and hippocampal formation. *Cell*, 184:3222–3241, 2021.

[94] N. W. et al. Gouwens. Integrated morphoelectric and transcriptomic classification of cortical gabaergic cells. *Cell*, 183:935–953, 2020.

[95] R. et al. Nguyen. Cholecystokinin-expressing interneurons of the medial prefrontal cortex mediate working memory retrieval. *Journal of Neuroscience*, 40:2314–2331, 2020.

[96] L. et al. Tricoire. A blueprint for the spatiotemporal origins of mouse hippocampal interneuron diversity. *Journal of Neuroscience*, 31:10948–10970, 2011.

[97] K. D. et al. Harris. Classes and continua of hippocampal ca1 inhibitory neurons revealed by single-cell transcriptomics. *PLoS Biology*, 16:e2006387, 2018.

[98] Tim Stuart, Andrew Butler, Paul Hoffman, Christoph Hafemeister, Efthymia Papalexi, William M Mauck III, Yuhan Hao, Marlon Stoeckius, Peter Smibert, and Rahul Satija. Comprehensive integration of single-cell data. *Cell*, 177(7):1888–1902, 2019.

[99] T. et al. Stuart. Comprehensive integration of single-cell data. *Cell*, 177:1888–1902, 2019.

[100] I. et al. Korsunsky. Fast, sensitive and accurate integration of single-cell data with harmony. *Nature Methods*, 16:1289–1296, 2019.

[101] D. et al. van Dijk. Recovering gene interactions from single-cell data using data diffusion. *Cell*, 174:716–729, 2018.

[102] P. Langfelder and S. Horvath. Wgcna: an r package for weighted correlation network analysis. *BMC Bioinformatics*, 9(559), 2008.

[103] A. Alexa and J. Rahnenfuhrer. topgo: enrichment analysis for gene ontology. `https://bioconductor.org/packages/release/bioc/html/topGO.html`.

[104] Y. et al. Omiya. Vglut3-expressing cck-positive basket cells construct invaginating synapses enriched with endocannabinoid signaling proteins in particular cortical and cortex-like amygdaloid regions of mouse brains. *Journal of Neuroscience*, 35:4215–4228, 2015.

[105] T. et al. Yoshida. Unique inhibitory synapse with particularly rich endocannabinoid signaling machinery on pyramidal neurons in basal amygdaloid nucleus. *Proceedings of the National Academy of Sciences, USA*, 108:3059–3064, 2011.

[106] I. et al. Katona. Distribution of cb1 cannabinoid receptors in the amygdala and their role in the control of gabaergic transmission. *Journal of Neuroscience*, 21:9506–9518, 2001.

[107] J. P. Sommeijer and C. N. Levelt. Synaptotagmin-2 is a reliable marker for parvalbumin positive inhibitory boutons in the mouse visual cortex. *PLoS One*, 7:e35323, 2012.

[108] L., Lukacsovich, D., Luo, W. and Földy. C. transcriptional and morphological profiling of parvalbumin interneuron subpopulations in the mouse hippocampus. *Nature Communications*, 12(108), 2021.

[109] Anthony, G., Reimann, A. and Kadenbach, B. Tissue-specific regulation of bovine heart cytochrome-c oxidase activity by adp via interaction with subunit via. *Proceedings of the National Academy of Sciences, USA*, 90:1652–1656, 1993.

[110] V. Frank and B. Kadenbach. Regulation of the h+/e- stoichiometry of cytochrome c oxidase from bovine heart by intramitochondrial atp/adp ratios. *FEBS Lett*, 382:121–124, 1996.

[111] B. et al. Sanz-Morello. Complex iv subunit isoform cox6a2 protects fast-spiking interneurons from oxidative stress and supports their function. *EMBO*, 39:e105759, 2020.

[112] Kadenbach, B., Hartmann, R., Glanville, R. and Buse, G. Tissue-specific genes code for polypeptide via of bovine liver and heart cytochrome c oxidase. *FEBS Lett*, 138:236–238, 1982.

[113] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[114] Thomas A Geddes, Taiyun Kim, Lihao Nan, James G Burchfield, Jean YH Yang, Dacheng Tao, and Pengyi Yang. Autoencoder-based cluster ensembles for single-cell rna-seq data analysis. *BMC bioinformatics*, 20(19):1–11, 2019.

[115] Gökcen Eraslan, Lukas M Simon, Maria Mircea, Nikola S Mueller, and Fabian J Theis. Single-cell rna-seq denoising using a deep count autoencoder. *Nature communications*, 10(1):1–14, 2019.

[116] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[117] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[118] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

[119] Gabriel Loaiza-Ganem and John P Cunningham. The continuous bernoulli: fixing a pervasive error in variational autoencoders. *Advances in Neural Information Processing Systems*, 32, 2019.

[120] Tallulah S Andrews, Vladimir Yu Kiselev, Davis McCarthy, and Martin Hemberg. Tutorial: guidelines for the computational analysis of single-cell rna sequencing data. *Nature protocols*, 16(1):1–9, 2021.

[121] Bo Yang, Xiao Fu, Nicholas D Sidiropoulos, and Mingyi Hong. Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In *international conference on machine learning*, pages 3861–3870. PMLR, 2017.

[122] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. *kdd*, 96(34):226–231, 1996.

[123] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[124] Zizhen Yao, Cindy TJ van Velthoven, Thuc Nghi Nguyen, Jeff Goldy, Adriana E Sedeno-Cortes, Fahimeh Baftizadeh, Darren Bertagnolli, Tamara Casper, Megan Chiang, Kirsten Crichton, et al. A taxonomy of transcriptomic cell types across the isocortex and hippocampal formation. *Cell*, 184(12):3222–3241, 2021.

[125] Cole Trapnell, Davide Cacchiarelli, Jonna Grimsby, Prapti Pokharel, Shuqiang Li, Michael Morse, Niall J Lennon, Kenneth J Livak, Tarjei S Mikkelsen, and John L Rinn. Pseudo-temporal ordering of individual cells reveals dynamics and regulators of cell fate decisions. *Nature biotechnology*, 32(4):381, 2014.

[126] Alexander Tong, Jessie Huang, Guy Wolf, David Van Dijk, and Smita Krishnaswamy. Trajectorynet: A dynamic optimal transport network for modeling cellular dynamics. In *International conference on machine learning*, pages 9526–9536. PMLR, 2020.

[127] David M Feliciano, Angélique Bordey, and Luca Bonfanti. Noncanonical sites of adult neurogenesis in the mammalian brain. *Cold Spring Harbor perspectives in biology*, 7(10):a018846, 2015.

[128] Ben D Fulcher, John D Murray, Valerio Zerbi, and Xiao-Jing Wang. Multimodal gradients across mouse cortex. *Proceedings of the National Academy of Sciences*, 116(10):4689–4695, 2019.

[129] Qian Li. sctour: a deep learning architecture for robust inference and accurate prediction of cellular dynamics. *bioRxiv*, 2022.

[130] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

[131] Richard L Burden, J Douglas Faires, and Annette M Burden. *Numerical analysis*. Cengage learning, 2015.

[132] Emma L Van Der Ende, Meifang Xiao, Desheng Xu, Jackie M Poos, Jessica L Panman, Lize C Jiskoot, Lieke H Meeter, Elise GP Dopper, Janne M Papma, Carolin Heller, et al. Neuronal pentraxin 2: a synapse-derived csf biomarker in genetic frontotemporal dementia. *Journal of Neurology, Neurosurgery & Psychiatry*, 91(6):612–621, 2020.

[133] Sung-Jin Lee, Mengping Wei, Chen Zhang, Stephan Maxeiner, ChangHui Pak, Salome Calado Botelho, Justin Trotter, Fredrik H Sterky, and Thomas C Südhof. Presynaptic neuronal pentraxin receptor organizes excitatory and inhibitory synapses. *Journal of Neuroscience*, 37(5):1062–1080, 2017.

[134] Abdullah A Osman, Mitsuyoshi Saito, Carol Makepeace, M Alan Permutt, Paul Schlesinger, and Mike Mueckler. Wolframin expression induces novel ion channel activity in endoplasmic reticulum membranes and increases intracellular calcium. *Journal of Biological Chemistry*, 278(52):52755–52762, 2003.

[135] Gavin Rumbaugh, J Paige Adams, Jee H Kim, and Richard L Huganir. Syngap regulates synaptic strength and mitogen-activated protein kinases in cultured neurons. *Proceedings of the National Academy of Sciences*, 103(12):4344–4351, 2006.

[136] Marie-Catherine Tiveron, Corinne Beurrier, Claire Ceni, Naly Andriambao, Alexis Combes, Muriel Koehl, Nicolas Maurice, Evelina Gatti, Dhoher Nora Abrous, Lydia Kerkerian-Le Goff, et al. Lamp5 fine-tunes gabaergic synaptic transmission in defined circuits of the mouse brain. *PloS one*, 11(6):e0157052, 2016.

[137] Olivier Caillard, Herman Moreno, Beat Schwaller, Isabel Llano, Marco R Celio, and Alain Marty. Role of the calcium-binding protein parvalbumin in short-term synaptic plasticity. *Proceedings of the National Academy of Sciences*, 97(24):13372–13377, 2000.

[138] Luca Franchini, Jennifer Stanic, Luisa Ponzoni, Manuela Mellone, Nicolò Carrano, Stefano Musardo, Elisa Zianni, Guendalina Olivero, Elena Marcello, Anna Pittaluga, et al. Linking nmda receptor synaptic retention to synaptic plasticity and cognition. *Iscience*, 19:927–939, 2019.

[139] James O McNamara, James H Eubanks, John Douglas McPherson, John J Wasmuth, Glen A Evans, and Stephen F Heinemann. Chromosomal localization of human glutamate receptor genes. *Journal of Neuroscience*, 12(7):2555–2562, 1992.

Appendix A

SCOUT supplemental figures

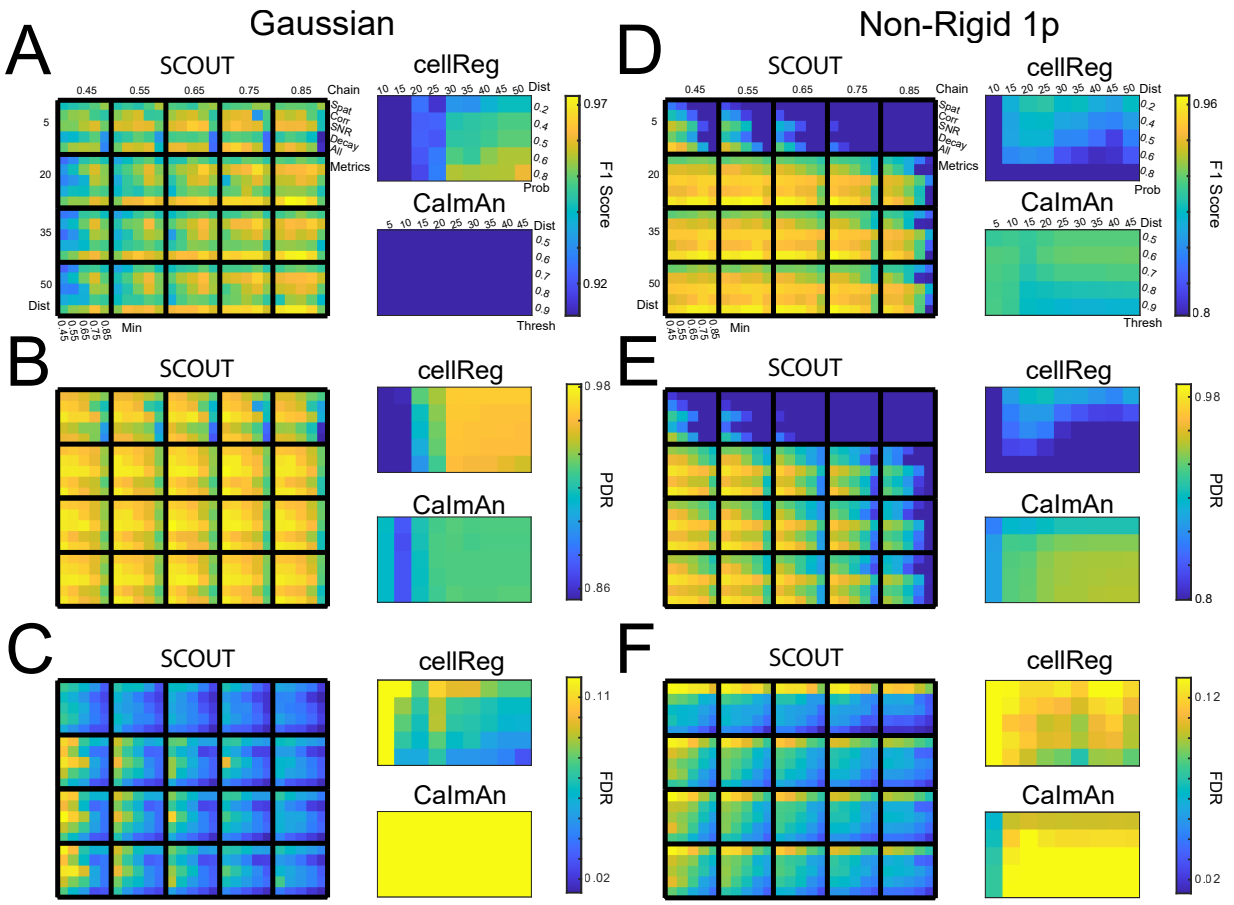

Figure A.1: Comparison of F1, PDR and FDR scores on the Gaussian and Non-Rigid 1p datasets. **A-E**: Each heatmap depicts a different metric for determining cell tracking quality: PDR the percentage of tracked neurons through all recordings of those present in each recording; FDR the percentage of tracked neurons containing at least one identification error; F1 Score defined as $F1 = \frac{2\times PDR\times(1-FDR)}{(1-FDR+PDR)}$. Parameters are labeled in **A**(**D**) and extend to **B-C**(**E-F**). Parameters for SCOUT include max_dist (labeled Dist), min_prob (labeled min), and chain_prob (labeled chain). The vertical change in each box corresponds to which metrics were used in the computation (spatial only, correlation, SNR, decay, or all metrics), the horizontal change in each box corresponds to variation in the min_prob parameter. Vertical change across boxes corresponds to variation in the max_dist parameter, and horizontal changes across boxes correspond to the chain_prob parameter. CaImAn and cellReg have a common maximum centroid distance (Dist) parameter, as well as threshold parameters (Prob, Thresh) governing acceptance levels for tracked cells. Results for Gaussian dataset on left (**A-C**), results for Non-Rigid 1p dataset on right (**D-F**).

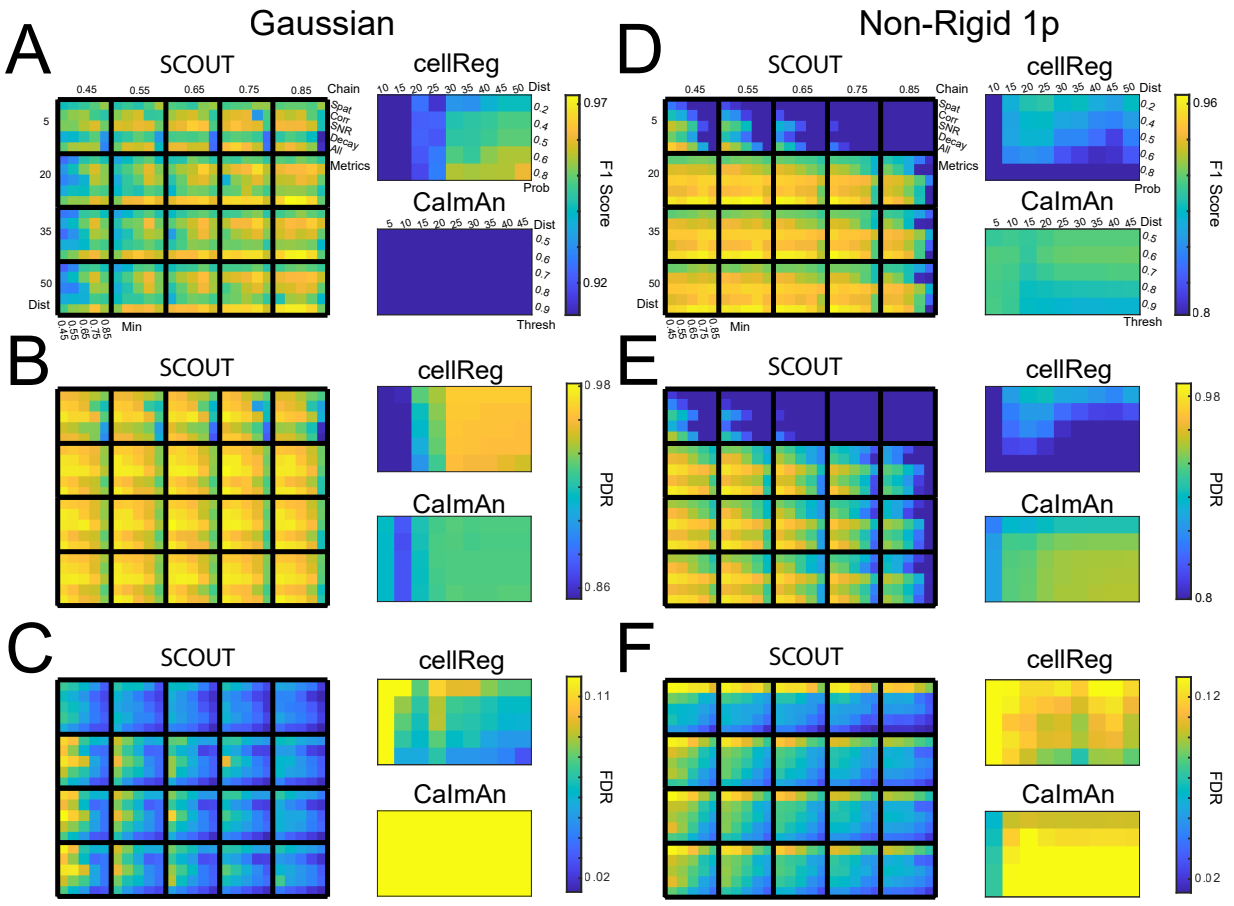

Figure A.2: Comparison of F1, PDR, and FDR scores on the Non-Rigid 2p and Individual Shift datasets. **A-C**: F1 scores (**A**), PDR (**B**), and FDR (**C**) for SCOUT, cellReg, and CaImAn across a variety of parameters for the Non-Rigid 1p dataset. Parameter labels (**A**) extend to **B-C**. **D-F**: F1 scores (**D**), PDR (**E**), and FDR (**F**) for SCOUT, cellReg, and CaImAn across a variety of parameters for the Non-Rigid 1p dataset. Parameter labels (**D**) extend to **E-F**. Results for the Non-Rigid 2p dataset on the left (**A-C**), results for the Individual Shift dataset on the right (**D-F**).

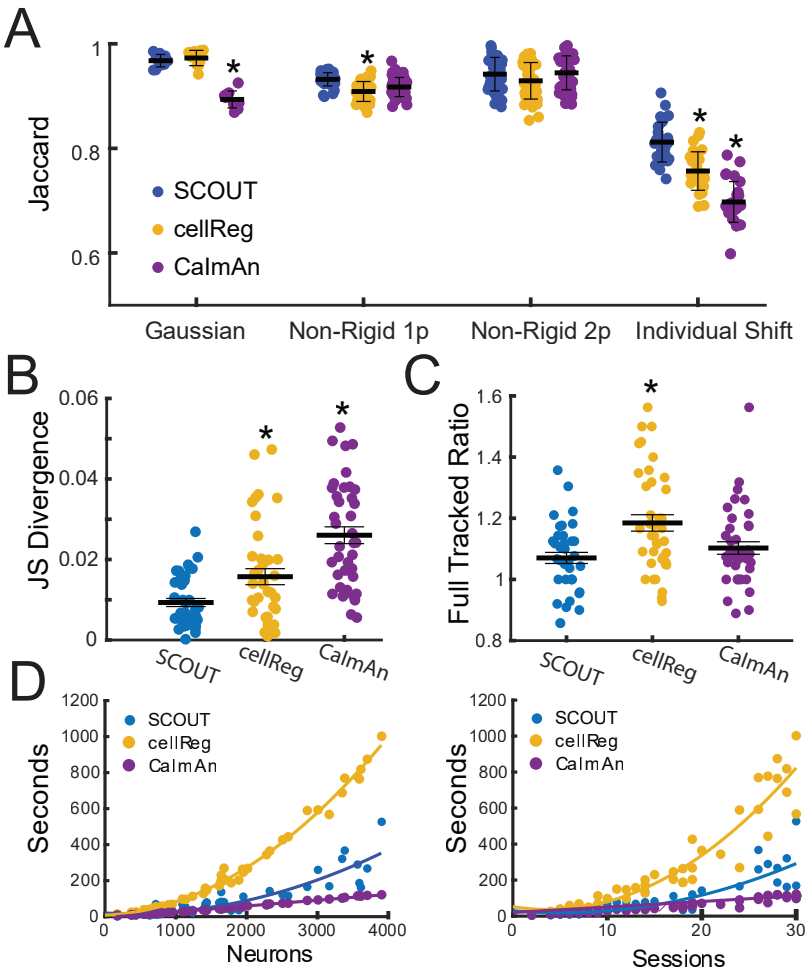

Figure A.3: Supplemental statistical comparisons of methods, Related to Figure 3. **A**: Maximal Jaccard similarity (y-axis) on modified versions of the simulated datasets (x-axis), in which 30% of neurons were removed from analysis prior to cell tracking. Statistical results denote pairwise comparison with SCOUT and are labeled with black asterisks (ANOVA, Bonferroni multiple comparisons). **B**: JS divergence (y-axis) between cluster size distributions of inferred cell registers for each method (x-axis), on the Non-Rigid 1p dataset after removal of 30% of neurons, based on parameters producing highest F1 scores. **C**: Full tracking ratios (y-axis, the percentage of available neurons tracked through all sessions) on the modified Non-Rigid 1p dataset based on parameters producing highest F1 scores. **D**: Algorithm runtime comparison between SCOUT, cellReg and CaImAn. Testing was performed on the Non-Rigid 1p dataset. We performed 50 tests in the following manner: 1) a random recording was identified; 2) recording sessions were duplicated to construct up to 30 sessions; 3) SCOUT, cellReg and CaImAn were run with global session registration disabled, and without the JS metric (SCOUT only) to provide the most unbiased results. Tests were run with 24 cores on a 2.2 Ghz CPU. Results are shown based on the seconds required to complete cell tracking (y-axis), and the total number of neurons (left) or sessions (right) in the recording. SCOUT and cellReg results were fit with a quadratic curve, and CaImAn with a linear curve.

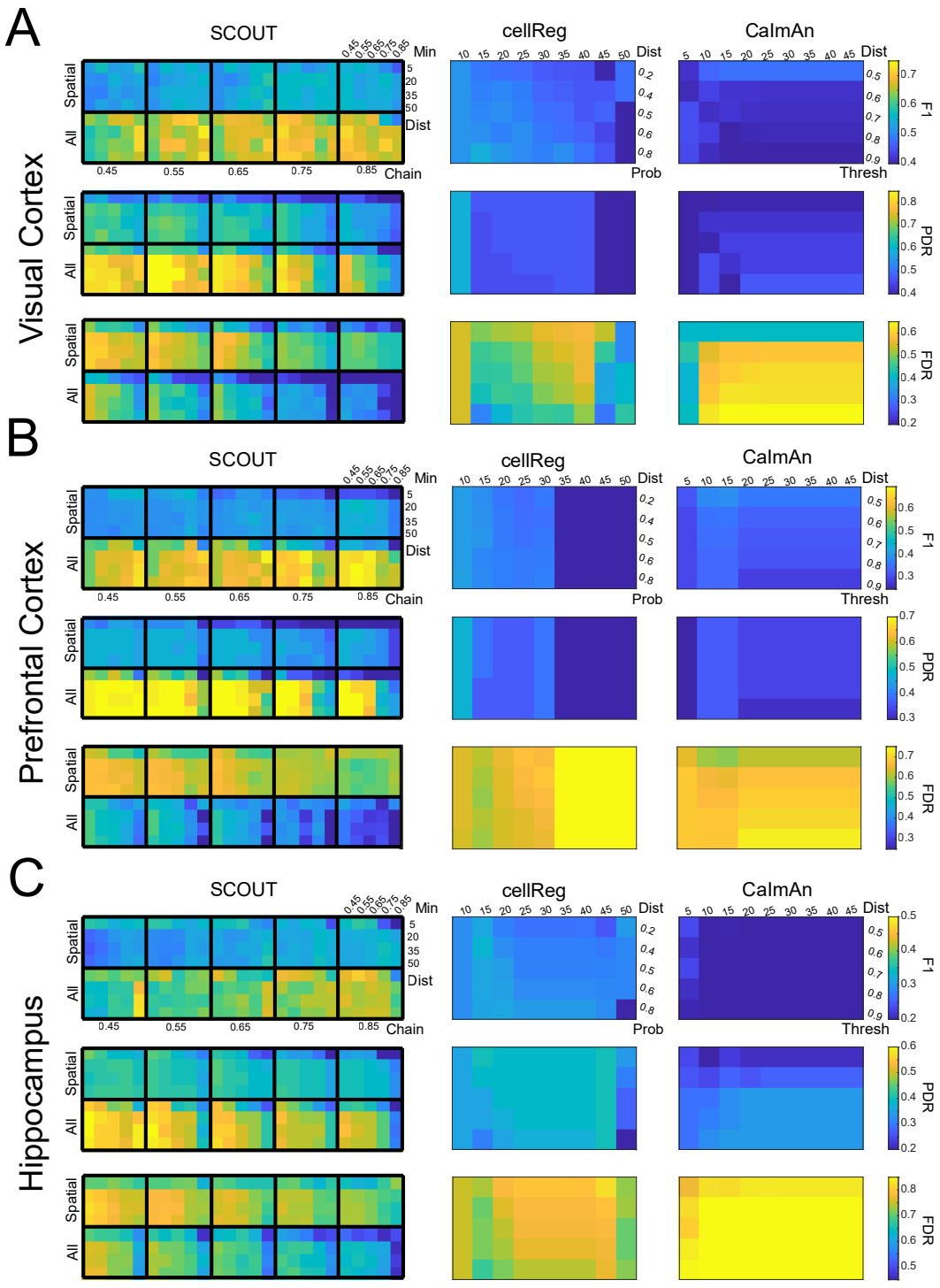

Figure A.4: SCOUT cell tracking performance on 1-photon in vivo recordings. **A**: F1 scores (top), PDR (middle), and FDR (bottom) for SCOUT, cellReg, and CaImAn across a variety of parameters (labeled top), for the visual cortex recording. **B**: F1 scores (top), PDR (middle), and FDR (bottom) for SCOUT, cellReg, and CaImAn across a variety of parameters (labeled top) for the prefrontal cortex recording. Multiple cellReg parameters resulted in errors, the associated F1 and PDR were set to 0, and FDR to 1. **C**: F1 scores (top), PDR (middle), and FDR (bottom) for SCOUT, cellReg, and CaImAn across a variety of parameters (labeled top) for the hippocampus recording.

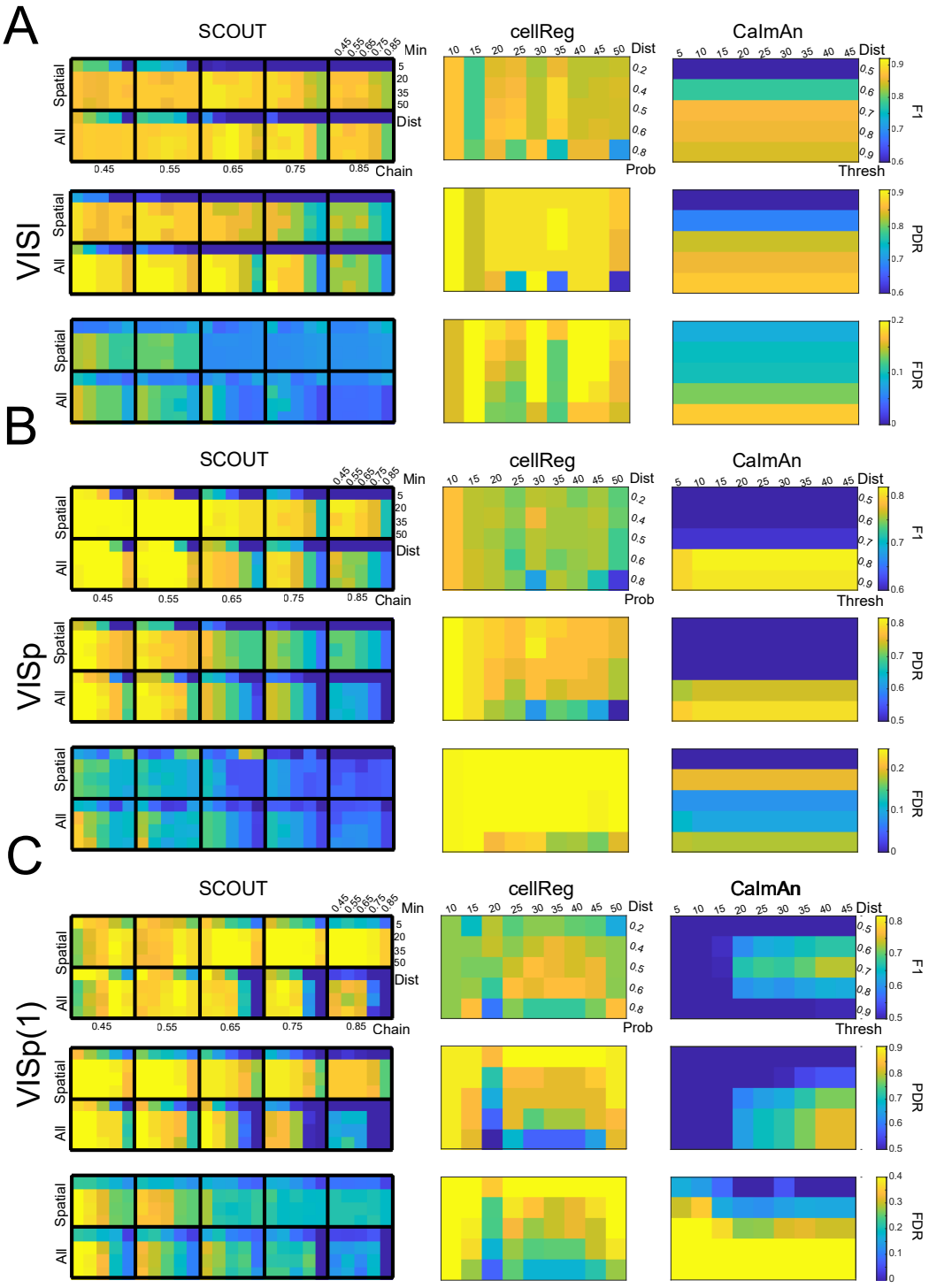

Figure A.5: SCOUT cell tracking performance on the 2-photon in vivo recordings. **A**: F1 scores (top), PDR (middle), and FDR (bottom) for SCOUT, cellReg, and CaImAn across a variety of parameters (labeled top) for the VISi recording. **B**: F1 scores (top), PDR (middle), and FDR (bottom) for SCOUT, cellReg, and CaImAn across a variety of parameters (labeled top) for the VISrl recording. **C**: F1 scores (top), PDR (middle), and FDR (bottom) for SCOUT, cellReg, and CaImAn across a variety of parameters (labeled top) for the VISp recording.