Title

Computational Prediction of Transcriptional Influence

Author

Cary, Michael Patrick

Publication Date

2020

Computational Prediction of Transcriptional Influence

by
Michael Cary

DISSERTATION
Submitted in partial satisfaction of the requirements for degree of
DOCTOR OF PHILOSOPHY

in

Biological and Medical Informatics

in the

GRADUATE DIVISION
of the
UNIVERSITY OF CALIFORNIA, SAN FRANCISCO

Approved:

DocuSigned by:
Cynthia J. Kenyon ______________________ Cynthia J. Kenyon
3FBD570EDEFD4CD...
Chair

DocuSigned by:
Hiten D. M ______________________ Hiten D. Madhani
[illegible]A4D0...
DocuSigned by:
Hana El-Samad ______________________ Hana El-Samad
[illegible]4DC...
DocuSigned by:
Hao Li ______________________ Hao Li
C2CA172BE8684DD...

____
Committee Members

DEDICATION

To Kelly, without whom none of this would have been possible.

ABSTRACT

Computational Prediction of Transcriptional Influence

Michael P. Cary

Genome-wide expression measurements remain difficult to interpret. Two major challenges lie in drawing firm conclusions from hundreds or even thousands of significantly changing genes, and in deriving hypotheses from the data that merit further testing. Identifying the degree to which each gene regulator acts to increase or decrease the expression of each gene, a concept I refer to as *transcriptional influence*, would greatly increase our ability to make sense of these data.

This work describes a new method to calculate the transcriptional influence that each regulatory motif in a *de novo* predicted set has on each gene represented in a gene expression measurement platform, using only a compendium of data from the platform and genome sequence information. The method uses independent component analysis (ICA) first to generate genetic regulatory modules, and then to predict DNA sequence motifs (putative regulatory sites) that are enriched in these modules. In a final step, the relative membership of each gene in each gene module and the enrichment of each sequence motif in each module are used to predict the relative influence of each sequence motif on each gene.

The power of these predictions is demonstrated in the analysis of microarray data for several *C. elegans* variants, including *isp-1* and *hif-1* mutants. *isp-1* mutations extend lifespan through the HIF-1 transcription factor, but there is no meaningful overlap among significant genes in *hif-1* and *isp-1* microarray datasets. In contrast, our method reveals extensive similarity in gene expression at a deeper level. Moreover, a regulatory motif predicted to have a strong influence in both datasets matches the canonical HIF-1 binding site.

TABLE OF CONTENTS

TABLE OF FIGURES

Chapter 1: Introduction

Advances in genome-wide experimental techniques, such as RNA, protein or metabolic profiling, have had a radical impact on biological research in recent years. These high-throughput techniques provide researchers with a wealth of data about a biological sample, such as the abundance of messenger RNA for nearly every gene in the genome, or the abundance of thousands of different proteins.

Concomitant with the widespread adoption of these technologies has been the development of bioinformatic techniques aimed at helping researchers make sense of the data they generate. Early work in this area focused mainly on developing statistical methods to help researchers decide which observed differences from a data-rich assay were most likely to be real, and which may have arisen simply due to random technical or biological variation not associated with a condition of interest (Cristoni & Bernardi 2004; Kerr & Churchill 2001; Satagopan & Panageas 2003). This work, combined with continued technological advances, has led to larger and larger sets of “significant” molecules from high-throughput assays. This in turn has led to a new challenge: from a list of hundreds or thousands of significant molecules, how does one determine which changes are germane to the biological questions at hand (Murray 2000)?

The challenge of interpreting results from high-throughput assays has spawned a flurry of research aimed at providing meaningful contexts for sets of molecules, especially genes and the proteins or non-coding RNAs that they encode (Bussemaker et al 2007; Curtis et al 2005;

Sivachenko et al 2007; Troyanskaya 2005). For example, efforts to provide functional context include the development and application of controlled vocabularies to gene annotation (Ashburner et al 2000; Jupe et al 2014; Mayer et al 2014), and the generation of pathway or network maps that show how genes and gene products interact with each other (Croft et al 2011; Dahlquist et al 2002; Kanehisa & Goto 2000).

An implicit and widespread assumption in much of this work is that most biological processes involve a relatively small, discrete set of molecules that together execute some function. I will refer to this as the *assumption of extreme specialization*, because it views most gene products as serving in very specific capacities (though many of these are unknown). For many cellular processes, there is abundant evidence that this assumption is valid. For example, some processes are carried out by macromolecular complexes. Many of these, for example, the ribosome, have been studied in great detail, such that the identity of the molecules comprising them, their physical structure and orientation, and their contribution to the overall function of the complex is known in some detail (Bedford et al 2010; Chakravarthy et al 2005).

There are, however, cellular processes in which the assumption of extreme specialization may not hold. Protein kinases provide a good illustration, as they may be much more promiscuous (i.e., capable of phosphorylating many different protein targets) than originally appreciated (Lienhard 2008). These many substrates might reflect a diversity of downstream functions, or possibly simply as excess kinase substrates that titrate down their activity (Kim et al 2011). Furthermore, the human kinome appears to be resilient to perturbation and able to rapidly circumvent the effect of highly specific kinase inhibitors (Graves et al 2013). These and other findings suggest that the set of human kinases may form an extremely highly connected network, and that the effect of specific phosphorylations on kinase function may range from neutral (no effect) to highly activating or highly inhibiting. Thus, the common motif in signaling pathways of a linear kinase cascade, in which a protein kinase acts to phosphorylate another protein kinase, which thereby becomes activated and phosphorylates yet another, may be an over-simplification of the biological reality – dozens or even hundreds of other kinases (and other proteins) may exert effects on the components of the pathway and transduction of the signal.

Another area in which the assumption of extreme specialization may not hold is that of regulatory interactions between transcription factors and DNA. Many models of small-scale genetic regulatory networks have been proposed in recent years, each typically involving a handful of transcription factors and dozens to hundreds of target genes (Lee & Tzou 2009). For example, Davidson, *et al.*, developed a model for the specification of the endomesoderm in the sea urchin embryo that contained 45 genes, and Hartemink, *et al.*, developed a model of the pheromone response in *S. cerevisiae* that contained 32 genes. Such models are useful for performing *in silico* experiments on the modeled system. These can lead to knowledge not easily obtained by traditional experimental methods, such as the discovery of emergent properties not apparent from studies of individual genes or proteins.

However, one criticism of small-scale genetic regulatory-network models is that they represent the function of gene regulators in isolation from genes and proteins not included in the model. There is growing experimental evidence, e.g. from *in vivo* protein-DNA binding assays such as ChIP-seq or ChIP-on-chip assays, that many, if not most, transcription factors bind proximally to a large fraction of the genome (Van Nostrand & Kim 2013). These assays also show that transcription factors bind to different genomic regions with a wide range of affinities, and that some genomic regions appear to be bound with high affinity by many different transcription factors. Thus, the true genetic regulatory network of the cell may be both highly connected and highly variable in the degree to which each gene regulator influences its target genes (Biggin 2011). A global set of predictions for the influence of each gene regulator on each gene (a concept I refer to as *transcriptional influence*) could therefore prove more useful in interpreting genome-wide expression measurements than methods that yield only sets of small-scale, largely disjoint genetic regulatory networks or pathway maps. Such predictions could also serve as the foundation for a global genetic regulatory network. This raises the question, "How can predict one predict transcriptional influence on a genome-wide scale?"

Here, I present a method that takes a significant step toward the accurate prediction of transcriptional influence genome-wide and toward the prediction of a global genetic regulatory network from a large compendium of gene expression measurements (Chapter 2). The method uses independent component analysis (ICA) first to generate genetic regulatory modules, and then to predict DNA sequence motifs (putative regulatory sites) that are enriched in these modules. In a final step, the relative membership of each gene in each gene module and the enrichment of each sequence motif in each module are used to predict the relative influence of each sequence motif on each gene.

While ICA has been applied to gene module prediction before, our method contains several innovations that result in significantly higher quality gene modules. The first of these is a data preprocessing optimization step. Investigations by other researchers in this area largely made use of established protocols for microarray preprocessing. Such protocols, however, were optimized for a different use case – the detection of significantly changing genes between two conditions or different points in a time series. Prior to beginning our work, it was not clear to us whether such protocols would also be optimal for the use of detecting gene modules and, indeed, we found that they were not.

A second major innovation in the gene module prediction part of our work lies in identifying the optimal number of gene modules to extract from a compendium of microarray data. ICA is a powerful source separation algorithm, but it is unable to predict the number of true latent sources in a dataset. Information theoretic approaches exist to help make this determination, but these rely on the assumption that samples comprising the dataset are independent and identically distributed (i.e., pulled randomly from the space of possible sample points). This assumption is clearly not valid for microarray data, in which most samples are replicated several times and a large fraction of the samples correspond to the organism or tissue in a wild type state. We used a “brute force” approach to source determination by using several different module quality metrics to evaluate the results of performing ICA with various numbers of extracted components. All of the quality metrics we applied gave similar results, and indicated that extracting just over 200 gene modules (that is, co-regulated gene sets) was optimal for the microarray compendium we analyzed. This is far less than the typical approach to ICA (in which the maximum number of components are extracted, 1386 in this case) would have generated. Thus, the gene modules we predicted are likely much more enriched for "true" gene modules than those that would be produced by other ICA-based methods.

Another innovation in our work lies in discretizing the independent components produced by ICA. While others have used a fixed threshold approach (e.g., setting genes with weights greater than 3 or less than -3 to be "in" the gene module), we found that this approach was suboptimal. We trained an artificial neural network on a large set of simulated data to find the optimal discretization threshold for each independent component based on the skewness and kurtosis of the component's weight distribution. We show that this method out-performs the fixed threshold approach.

Because our main motivation in developing an accurate, genome-wide model of gene regulation was to use it to aid the interpretation of experimental data, we tested the utility of our predictions with several case studies. We found that our predictions could identify key (genetically-validated) regulators that were not be revealed by traditional methods of gene expression analysis. We anticipate that our method will help us understand key regulatory mechanisms of biological processes we are currently investigating, such as the process of aging.

Application of our predictions to the interpretation of longevity data sets and other planned research activities are discussed in more detail in Chapter 3, Future Directions.

References

2004. The ENCODE (ENCyclopedia Of DNA Elements) Project. *Science (New York, N.Y* 306:636-40

Aittokallio T, Kurki M, Nevalainen O, Nikula T, West A, Lahesmaa R. 2003. Computational strategies for analyzing data in gene expression microarray experiments. *J Bioinform Comput Biol* 1:541-86

Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, et al. 2000. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nature genetics* 25:25-9

Bansal M, Belcastro V, Ambesi-Impiombato A, di Bernardo D. 2007. How to infer gene networks from expression profiles. *Molecular systems biology* 3:78

Bar-Joseph Z, Gerber GK, Lee TI, Rinaldi NJ, Yoo JY, et al. 2003. Computational discovery of gene modules and regulatory networks. *Nat Biotechnol* 21:1337-42

Barrett T, Troup DB, Wilhite SE, Ledoux P, Evangelista C, et al. 2011. NCBI GEO: archive for functional genomics data sets--10 years on. *Nucleic acids research* 39:D1005-10

Bedford L, Paine S, Sheppard PW, Mayer RJ, Roelofs J. 2010. Assembly, structure, and function of the 26S proteasome. *Trends Cell Biol* 20:391-401

Bell EL, Klimova TA, Eisenbart J, Schumacker PT, Chandel NS. 2007. Mitochondrial reactive oxygen species trigger hypoxia-inducible factor-dependent extension of the replicative life span during hypoxia. *Molecular and cellular biology* 27:5737-45

Bergmann S, Ihmels J, Barkai N. 2003. Iterative signature algorithm for the analysis of large-scale gene expression data. *Phys Rev E Stat Nonlin Soft Matter Phys* 67:031902

Biggin MD. 2011. Animal transcription networks as highly connected, quantitative continua. *Developmental cell* 21:611-26

Bolstad BM, Irizarry RA, Astrand M, Speed TP. 2003. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics (Oxford, England)* 19:185-93

Bussemaker HJ, Li H, Siggia ED. 2000. Building a dictionary for genomes: identification of presumptive regulatory sites by statistical analysis. *Proc Natl Acad Sci U S A* 97:10096-100

Bussemaker HJ, Ward LD, Boorsma A. 2007. Dissecting complex transcriptional responses using pathway-level scores based on prior information. *BMC bioinformatics* 8 Suppl 6:S6

Chakravarthy S, Park YJ, Chodaparambil J, Edayathumangalam RS, Luger K. 2005. Structure and dynamic properties of nucleosome core particles. *FEBS Lett* 579:895-8

Chatziioannou A, Moulos P, Kolisis FN. 2009. Gene ARMADA: an integrated multi-analysis platform for microarray data implemented in MATLAB. *BMC bioinformatics* 10:354

Chiappetta P, Roubaud MC, Torresani B. 2004. Blind source separation and the analysis of microarray data. *J Comput Biol* 11:1090-109

Comon P. 1994. Independent Component Analysis: a new concept? *Signal Processing* 36:287-314

Cristina D, Cary M, Lunceford A, Clarke C, Kenyon C. 2009. A regulated response to impaired respiration slows behavioral rates and increases lifespan in Caenorhabditis elegans. *PLoS genetics* 5:e1000450

Cristoni S, Bernardi LR. 2004. Bioinformatics in mass spectrometry data analysis for proteomics studies. *Expert Rev Proteomics* 1:469-83

Croft D, O'Kelly G, Wu G, Haw R, Gillespie M, et al. 2011. Reactome: a database of reactions, pathways and biological processes. *Nucleic acids research* 39:D691-7

Curtis RK, Oresic M, Vidal-Puig A. 2005. Pathways to the analysis of microarray data. *Trends Biotechnol* 23:429-35

Dahlquist KD, Salomonis N, Vranizan K, Lawlor SC, Conklin BR. 2002. GenMAPP, a new tool for viewing and analyzing microarray data on biological pathways. *Nature genetics* 31:19-20

Dalma-Weiszhausz DD, Warrington J, Tanimoto EY, Miyada CG. 2006. The affymetrix GeneChip platform: an overview. *Methods in enzymology* 410:3-28

Dupuy D, Li QR, Deplancke B, Boxem M, Hao T, et al. 2004. A first version of the Caenorhabditis elegans Promoterome. *Genome Res* 14:2169-75

Durinck S, Moreau Y, Kasprzyk A, Davis S, De Moor B, et al. 2005. BioMart and Bioconductor: a powerful link between biological databases and microarray data analysis. *Bioinformatics (Oxford, England)* 21:3439-40

Durinck S, Spellman PT, Birney E, Huber W. 2009. Mapping identifiers for the integration of genomic datasets with the R/Bioconductor package biomaRt. *Nat Protoc* 4:1184-91

Eklund AC, Szallasi Z. 2008. Correction of technical bias in clinical microarray data improves concordance with known biological information. *Genome biology* 9:R26

Engreitz JM, Daigle BJ, Jr., Marshall JJ, Altman RB. 2010. Independent component analysis: mining microarray data for fundamental human gene expression modules. *J Biomed Inform* 43:932-44

Frigyesi A, Veerla S, Lindgren D, Hoglund M. 2006. Independent component analysis reveals new and biologically significant structures in micro array data. *BMC bioinformatics* 7:290

Furuyama T, Nakazawa T, Nakano I, Mori N. 2000. Identification of the differential distribution patterns of mRNAs and consensus binding sequences for mouse DAF-16 homologues. *Biochem J* 349:629-34

Gautier L, Cope L, Bolstad BM, Irizarry RA. 2004. affy--analysis of Affymetrix GeneChip data at the probe level. *Bioinformatics (Oxford, England)* 20:307-15

Gerstein MB, Lu ZJ, Van Nostrand EL, Cheng C, Arshinoff BI, et al. 2010. Integrative analysis of the Caenorhabditis elegans genome by the modENCODE project. *Science (New York, N.Y* 330:1775-87

Gong T, Xuan J, Wang C, Li H, Hoffman E, et al. 2007. Gene module identification from microarray data using nonnegative independent component analysis. *Gene Regul Syst Bio* 1:349-63

Graves LM, Duncan JS, Whittle MC, Johnson GL. 2013. The dynamic nature of the kinome. *Biochem J* 450:1-8

Hunt-Newbury R, Viveiros R, Johnsen R, Mah A, Anastas D, et al. 2007. High-throughput in vivo analysis of gene expression in Caenorhabditis elegans. *PLoS Biol* 5:e237

Hwang AB, Lee SJ. 2011. Regulation of life span by mitochondrial respiration: the HIF-1 and ROS connection. *Aging (Albany NY)* 3:304-10

Hyvarinen A, Oja E. 2000. Independent component analysis: algorithms and applications. *Neural Netw* 13:411-30

Hyvärinen A, Oja E. 2000. Independent Component Analysis: Algorithms and Application. *Neural Networks* 13:411-30

Ihmels J, Bergmann S, Barkai N. 2004. Defining transcription modules using large-scale gene expression data. *Bioinformatics (Oxford, England)* 20:1993-2003

Jupe S, Jassal B, Williams M, Wu G. 2014. A controlled vocabulary for pathway entities and events. *Database (Oxford)* 2014

Kanehisa M, Goto S. 2000. KEGG: kyoto encyclopedia of genes and genomes. *Nucleic acids research* 28:27-30

Kellis M, Wold B, Snyder MP, Bernstein BE, Kundaje A, et al. 2014. Defining functional DNA elements in the human genome. *Proc Natl Acad Sci U S A* 111:6131-8

Kerr MK, Churchill GA. 2001. Statistical design and the analysis of gene expression microarray data. *Genet Res* 77:123-8

Kim SK, Lund J, Kiraly M, Duke K, Jiang M, et al. 2001. A gene expression map for Caenorhabditis elegans. *Science (New York, N.Y* 293:2087-92

Kim Y, Paroush Z, Nairz K, Hafen E, Jimenez G, Shvartsman SY. 2011. Substrate-dependent control of MAPK phosphorylation in vivo. *Molecular systems biology* 7:467

Kong W, Vanderburg CR, Gunshin H, Rogers JT, Huang X. 2008. A review of independent component analysis application to microarray gene expression data. *Biotechniques* 45:501-20

Kouns NA, Nakielna J, Behensky F, Krause MW, Kostrouch Z, Kostrouchova M. 2011. NHR-23 dependent collagen and hedgehog-related genes required for molting. *Biochem Biophys Res Commun* 413:515-20

Leach M. 2004. Gene expression informatics. *Methods Mol Biol* 258:153-65

Lee HC, Lai K, Lorenc MT, Imelfort M, Duran C, Edwards D. 2012. Bioinformatics tools and databases for analysis of next-generation sequence data. *Brief Funct Genomics* 11:12-24

Lee SI, Batzoglou S. 2003. Application of independent component analysis to microarrays. *Genome biology* 4:R76

Lee SJ, Hwang AB, Kenyon C. 2010. Inhibition of respiration extends C. elegans life span via reactive oxygen species that increase HIF-1 activity. *Curr Biol* 20:2131-6

Lee WP, Tzou WS. 2009. Computational methods for discovering gene networks from expression data. *Brief Bioinform* 10:408-23

Li H, Sun Y, Zhan M. 2007. The discovery of transcriptional modules by a two-stage matrix decomposition approach. *Bioinformatics (Oxford, England)* 23:473-9

Liebermeister W. 2002. Linear modes of gene expression determined by independent component analysis. *Bioinformatics (Oxford, England)* 18:51-60

Lienhard GE. 2008. Non-functional phosphorylations? *Trends in biochemical sciences* 33:351-2

Mayer G, Jones AR, Binz PA, Deutsch EW, Orchard S, et al. 2014. Controlled vocabularies and ontologies in proteomics: overview, principles and practice. *Biochim Biophys Acta* 1844:98-107

Michoel T, De Smet R, Joshi A, Marchal K, Van de Peer Y. 2009. Reverse-engineering transcriptional modules from gene expression data. *Ann N Y Acad Sci* 1158:36-43

Murphy CT, McCarroll SA, Bargmann CI, Fraser A, Kamath RS, et al. 2003. Genes that act downstream of DAF-16 to influence the lifespan of Caenorhabditis elegans. *Nature* 424:277-83

Murray AW. 2000. Whither genomics? *Genome biology* 1:COMMENT003

Niu W, Lu ZJ, Zhong M, Sarov M, Murray JI, et al. 2011. Diverse transcription factor binding features revealed by genome-wide ChIP-seq in C. elegans. *Genome Res* 21:245-54

Pham TH, Satou K, Ho TB. 2004. Mining yeast transcriptional regulatory modules from factor DNA-binding sites and gene expression data. *Genome Inform* 15:287-95

Roy S, Ernst J, Kharchenko PV, Kheradpour P, Negre N, et al. 2010. Identification of functional elements and regulatory circuits by Drosophila modENCODE. *Science (New York, N.Y* 330:1787-97

Rubinstein R, Simon I. 2005. MILANO--custom annotation of microarray results using automatic literature searches. *BMC bioinformatics* 6:12

Rustici G, Kolesnikov N, Brandizi M, Burdett T, Dylag M, et al. ArrayExpress update--trends in database growth and links to data analysis tools. *Nucleic acids research* 41:D987-90

Satagopan JM, Panageas KS. 2003. A statistical perspective on gene expression data analysis. *Stat Med* 22:481-99

Segal E, Battle A, Koller D. 2003a. Decomposing gene expression into cellular processes. *Pac Symp Biocomput*:89-100

Segal E, Shapira M, Regev A, Pe'er D, Botstein D, et al. 2003b. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature genetics* 34:166-76

Segal E, Yelensky R, Koller D. 2003c. Genome-wide discovery of transcriptional modules from DNA sequence and gene expression. *Bioinformatics (Oxford, England)* 19 Suppl 1:i273-82

Shen C, Nettleton D, Jiang M, Kim SK, Powell-Coffman JA. 2005. Roles of the HIF-1 hypoxia-inducible factor during hypoxia response in Caenorhabditis elegans. *The Journal of biological chemistry* 280:20580-8

SImes RJ. 1986. An Improved Bonferroni Procedure for Multiple Tests of Significance. *Biometrika* 73:751-4

Sivachenko AY, Yuryev A, Daraselia N, Mazo I. 2007. Molecular networks in microarray analysis. *J Bioinform Comput Biol* 5:429-56

Smyth GK. 2005. Limma: linear models for microarray data. In *Bioinformatics and Computational Biology Solutions using R and Bioconductor*, ed. R Gentleman, V Carey, S Dudoit, R Irizarry, W Huber, pp. 397-420. New York: Springer

Soinov LA, Krestyaninova MA, Brazma A. 2003. Towards reconstruction of gene networks from expression data by supervised learning. *Genome biology* 4:R6

Thomas-Chollier M, Defrance M, Medina-Rivera A, Sand O, Herrmann C, et al. RSAT 2011: regulatory sequence analysis tools. *Nucleic acids research* 39:W86-91

Troyanskaya OG. 2005. Putting microarrays in a context: integrated analysis of diverse biological data. *Brief Bioinform* 6:34-43

van Helden J, Andre B, Collado-Vides J. 1998. Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *J Mol Biol* 281:827-42

Van Nostrand EL, Kim SK. 2013. Integrative analysis of C. elegans modENCODE ChIP-seq data sets to infer gene regulatory interactions. *Genome Res* 23:941-53

Vermeirssen V, Joshi A, Michoel T, Bonnet E, Casneuf T, Van de Peer Y. 2009. Transcription regulatory networks in Caenorhabditis elegans inferred through reverse-engineering of gene expression profiles constitute biological hypotheses for metazoan development. *Mol Biosyst* 5:1817-30

Wang Y, Joshi T, Zhang XS, Xu D, Chen L. 2006. Inferring gene regulatory networks from multiple microarray datasets. *Bioinformatics (Oxford, England)* 22:2413-20

Xia X, McClelland M, Wang Y. 2005. WebArray: an online platform for microarray data analysis. *BMC bioinformatics* 6:306

Zaslaver A, Baugh LR, Sternberg PW. Metazoan operons accelerate recovery from growth-arrested states. *Cell* 145:981-92

Zhang P, Judy M, Lee SJ, Kenyon C. Direct and indirect gene regulation by a life-extending FOXO protein in C. elegans: roles for GATA factors and lipid gene regulators. *Cell Metab* 17:85-100

Zhang Y, Szustakowski J, Schinke M. 2009. Bioinformatics analysis of microarray data. *Methods Mol Biol* 573:259-84

Chapter 2: Predicting the influence of regulatory sequences on gene expression

Abstract

Identifying the genes and gene regulators that specify and execute essential biological activities is a fundamental goal in biology. In principle, gene expression data coupled with knowledge of transcription factor binding sites should advance this goal, yet recent findings indicate that most transcription factors bind proximally to a much wider array of genes *in vivo* than they are predicted to regulate. Thus more nuanced models may be needed to predict genetic control circuitry and regulatory factors. Toward this end, we have developed a powerful method to calculate the transcriptional influence that each regulatory motif in a *de novo* predicted set has on each gene represented in a gene expression measurement platform, using only a compendium of data from the platform and genome sequence information. We demonstrate the power of our predictions of transcriptional influence by using them to analyze microarray data for several *C. elegans* variants, including *isp-1* and *hif-1* mutants. *isp-1* mutations extend lifespan through the HIF-1 transcription factor, but we observed no meaningful overlap among significant genes in *hif-1* and *isp-1* microarray datasets. In contrast, our method reveals extensive similarity in gene expression at a deeper level. Moreover, a regulatory motif predicted to have a strong influence in both datasets matches the canonical HIF-1 binding site.

Introduction

Despite the widespread availability of bioinformatic tools for interpreting genome-wide transcript measurements(Aittokallio et al 2003; Chatziioannou et al 2009; Leach 2004; Lee et al 2012; Rubinstein & Simon 2005; Xia et al 2005; Zhang et al 2009), extracting testable hypotheses from such data remains difficult. In our view, this difficulty stems largely from a lack of specific knowledge about the relationships between gene expression regulators, such as transcription factors, and the sets of genes they control. A desire to generate such knowledge has motivated large-scale efforts, such as the ENCODE and modENCODE projects, which collect *in vivo* binding data for known gene regulators(2004; Gerstein et al 2010; Kellis et al 2014; Niu et al 2011; Roy et al 2010). A surprising result from these projects is that many, if not most, transcription factors bind proximally to a much wider array of genes than expected, including genes not thought to be under their control, and often to regions lacking canonical binding sites(Van Nostrand & Kim 2013).

These findings reinforce the concept that accurate models of gene regulation must include more than just the presence or absence of transcription-factor binding. The degree of influence that a factor has on the expression level of each of its target genes, a concept we refer to as transcriptional influence, must also be represented. Toward this end, we have developed a method to predict transcriptional influence between a large set of regulatory sequence motifs, predicted *de novo*, and each gene represented in an extensive expression platform, using only a compendium of data from the platform and genome sequence as input.

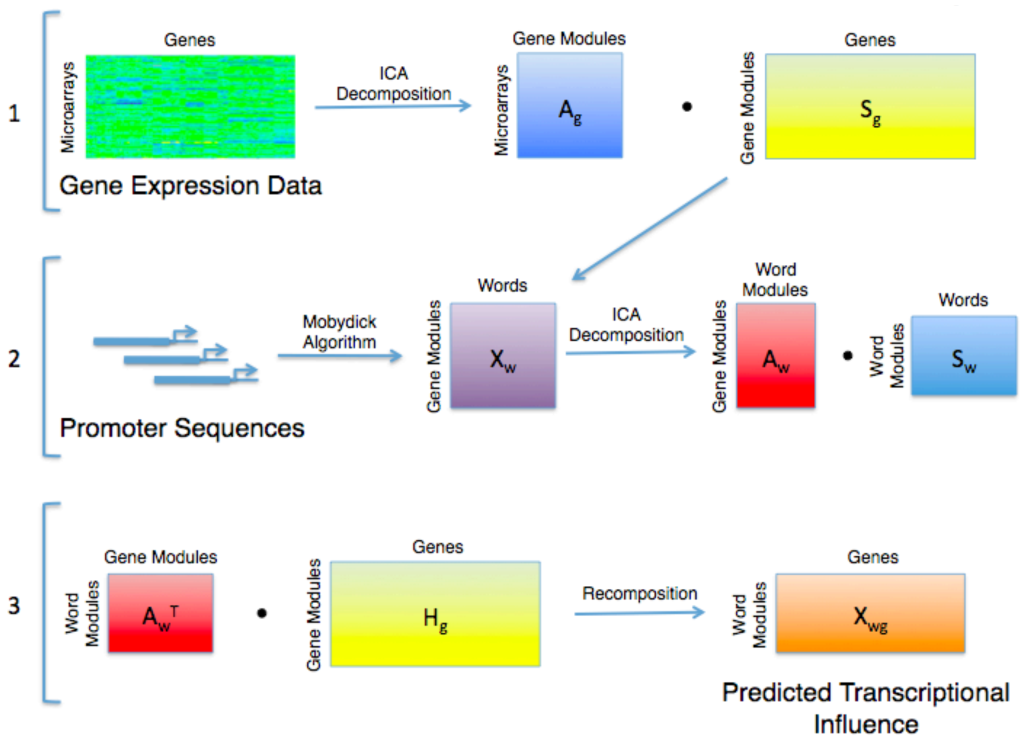


Figure 2.1. Diagram of transcriptional influence prediction

The three main steps of our method. In step 1, a matrix of gene expression data, Xg, is decomposed using independent component analysis (ICA), producing a gene module definition matrix, Sg, and a matrix indicating the weight of each module (set of co-regulated genes) in each gene-expression measurement, Ag. In step 2, a dictionary of potential regulatory sequence elements is constructed using the Mobydick algorithm. The words of this dictionary are assessed for enrichment in the promoters of the genes comprising each gene module, and p-values for word / gene module pairs are used to construct a new matrix, Xw. ICA decomposition of this matrix produces a word module matrix, Sw, and a matrix indicating the weight of each word module in each gene module, Aw. In step 3, the product of Hg (a transformation of Sg, see text) and the transpose of Aw is calculated. We interpret the values in the resulting matrix as the transcriptional influences of each word module on each gene within the context of the gene expression data provided as input to step 1.

Our method consists of three main steps (Fig. 2.1). In the first, we predict gene transcription modules (sets of co-regulated genes) using independent component analysis (ICA) of a large compendium of expression data. ICA has been applied to gene module prediction before(Engreitz et al 2010; Gong et al 2007; Lee & Batzoglou 2003; Li et al 2007; Liebermeister 2002), but we have refined the process in a way that improves the results substantially according to several different measures. These predicted gene modules serve as an intermediate data structure in our algorithm for transcriptional influence, but they are revealing in their own right, and provide new insights into properties of gene expression, some of which we present here.

In the second step, we calculate the module-wise enrichment or deprivation of "words" (oligonucleotide sequences from a dictionary compiled using annotated DNA sequence) in the promoter regions of module genes. We create a matrix with this data and perform ICA to generate word modules; that is, sets of words that appear together (or are absent together) in the promoters of predicted gene modules. We interpret word modules, which generally comprise closely related oligonucleotide sequences, as DNA sequences with shared regulatory potential, e.g. transcription factor binding sites. Finally, we determine the matrix product of the gene module source matrix from step 1 and the word module mixing matrix resulting from step 2, which yields a matrix relating each word module to each gene, i.e., the predicted transcriptional influence of each potential regulatory sequence on each gene.

We validate our method by using transcriptional influence predictions to analyze gene expression data from several experiments, including *C. elegans isp-1* (respiratory chain) and *hif-1* (hypoxia-inducible transcription factor) mutant microarrays. Mutation of the *isp-1* gene extends lifespan in a *hif-1*-dependent fashion(Bell et al 2007; Hwang & Lee 2011; Lee et al

2010), but the significant gene sets from microarray data for these two mutants have little in common. In contrast, our method reveals that both conditions do impact common sets of downstream genes. Moreover, using our transcriptional influence predictions to analyze the data, we find that several word modules are predicted to exert strong effects in both data sets, and that the words comprising one of these modules match the canonical HIF-1 binding site.

Results

Optimization of genetic regulatory module prediction

Our algorithm relies on accurate predictions of genetic regulatory modules. A large body of gene expression data is publicly available(Barrett et al 2011; Rustici et al) and has enabled computational prediction of gene modules (co-regulated genes) by several groups(Bansal et al 2007; Bar-Joseph et al 2003; Bergmann et al 2003; Engreitz et al 2010; Ihmels et al 2004; Kim et al 2001; Michoel et al 2009; Pham et al 2004; Segal et al 2003a; Segal et al 2003b; Segal et al 2003c; Soinov et al 2003; Vermeirssen et al 2009; Wang et al 2006). Preliminary experimentation with published methods led us to choose ICA for performing module prediction, as modules predicted with ICA yielded stronger oligonucleotide enrichment in promoter regions than did modules predicted with the other methods we tested (Fig. 2.2e and additional data not shown; see Lee & Batzoglou(Lee & Batzoglou 2003) for additional comparisons of ICA to other methods).

Briefly, ICA is a blind source separation method that attempts to “unmix” a signal comprising additive subcomponents by separating it into statistically independent source signals(Comon 1994; Hyvärinen & Oja 2000). In the common notation, a data matrix, ***X***, comprising multiple observations of a multidimensional variable, ***x***, is decomposed into two new matrices, a mixing matrix, ***A***, and a source matrix, ***S***:

$$\boldsymbol{X} = \boldsymbol{AS} \qquad (1)$$

The ***A*** matrix contains the weight of each independent component in each observation, and the ***S*** matrix contains the weight of each element of ***x*** in each independent component. In the context of gene expression analysis, the elements of ***x*** correspond to genes, the observations correspond to genome-wide gene expression measurements, such as microarrays, and the independent components are interpreted as gene modules (essentially, genes whose expression levels change similarly across multiple arrays). The values in the ***S*** matrix correspond to the relative levels of inclusion of each gene in each gene module(Kong et al 2008; Liebermeister 2002).

Our preliminary investigation indicated that the performance of ICA was sensitive both to data preprocessing and to the number of components extracted. Therefore, we first sought to optimize gene module prediction through ICA, evaluating results using biological information,

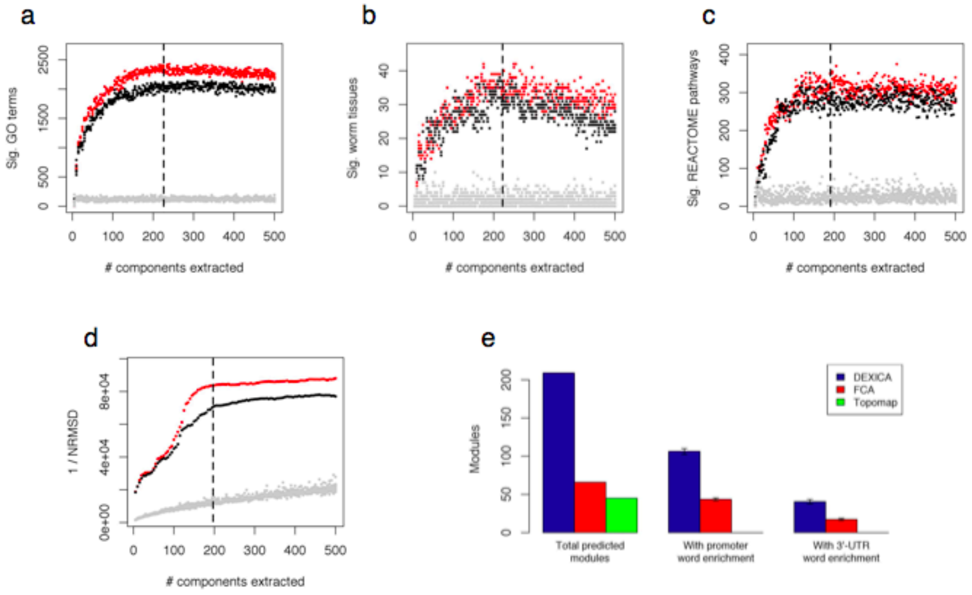


Figure 2.2. Gene module prediction

To determine the optimal preprocessing method and the optimal number of components (gene modules) to extract from a gene expression compendium of *C. elegans* microarray data, we determined the number of Gene Ontology terms (a), *C. elegans* tissues (b), and REACTOME pathways (c) that were significantly enriched in at least one gene module. We also quantified the ability of each predicted set of gene modules to accurately represent data from a different, two-color gene expression platform (d). Black points show results from a compendium produced using a previously published preprocessing procedure[15]; red points show results for the best alternative preprocessing method that we tested. Dashed lines indicate the point on the x-axis of each graph at which loess regression curves showed the greatest difference between red points and results from randomized controls (grey points). (e) The total number of predicted modules and the number of modules with significant regulatory word enrichment for our method (DEXICA), another ICA-based method, FCA[15], and gene sets from the *C. elegans* gene expression topomap[25]; error bars indicate s.d. between repeat runs of DEXICA / FCA.

including Gene Ontology (GO) term enrichment(Ashburner et al 2000), REACTOME pathway enrichment(Croft et al 2011), and tissue-specific expression enrichment in predicted gene modules. We applied our optimization strategy to a compendium of 1386 *C. elegans* Affymetrix arrays(Dalma-Weiszhausz et al 2006), which we obtained from the Gene Expression Omnibus

(GEO) database(Barrett et al 2011). Our preliminary results indicated that applying dimension reduction procedures on the data matrix prior to performing ICA reduced the number of biologically significant components in the end result (data not shown), so we chose to optimize ICA of the full data matrix of 1386 arrays. We found that the highest quality modules were produced when we omitted between-experiment quantile normalization from the preprocessing procedure (see Methods) and when the number of extracted components (i.e., gene expression modules, or sets of co-regulated genes) ranged from 191 to 226 (Fig. 2.2a-c).

Because each of the module quality metrics we used relied on prior knowledge, which could be incomplete or inaccurate, we also gauged module quality by quantifying the ability of each predicted set of modules to represent expression data from a different gene expression measurement platform (a two-color microarray; see Methods), with the expectation that more accurate gene modules would be able to capture such data more closely than less accurate modules. This measure of module quality produced results similar to those produced by the other measures, with the optimal number of extracted components occurring at 197 (Fig. 2.2d).

With the exception of representing data from an alternative platform, all of our module quality measures required translation of the independent components generated by ICA into discrete sets of genes, a process we refer to as discretization. Typically, each component (gene module) is discretized into two sets of genes that are regulated in opposite directions. We refer to these two sets as “hemi-modules”, one set consisting of genes with highly positive weights and another consisting of genes with highly negative weights in the independent component.

Others have used a fixed threshold approach to discretization(Chiappetta et al 2004; Engreitz et al 2010; Lee & Batzoglou 2003), for example, defining genes with weights exceeding +/- 3 standard deviations from the component mean to be “in” each hemi-module, and this is the approach we applied in figures 2.2a-c. However, we found that individual modules showed maximum annotation enrichment at different thresholds, suggesting that a ‘one-size-fits-all’ approach to discretization is sub-optimal. An alternative approach to discretization that we attempted (described in Frigyesi *et al.*(Frigyesi et al 2006)) failed to converge in many cases. Therefore, to increase discretization accuracy, we trained an artificial neural network (ANN) to predict thresholds for discretization of each component from the skewness and kurtosis of its weight distribution (see Supplemental Methods). Using this artificial neural network for discretization in our optimization process produced similar results qualitatively (Fig. 2.3a-c), but resulted in a greater number of significant annotations across the range of parameters tested than did threshold discretization ($p < 2.2e-16$, Fig. 2.3d). Therefore, we used ANN discretization in the subsequent steps of our algorithm.

The mean optimum number of extracted components, determined using the module quality measures we applied, was similar for both threshold discretization and ANN discretization (209, and 209.33, respectively), and we chose 209 as the final number of components to extract from the *C. elegans* Affymetrix microarray compendium. We refer to our process of first optimizing and then executing gene module prediction from a non-dimension-reduced gene expression compendium using ICA as DEXICA, for deep extraction independent component analysis.

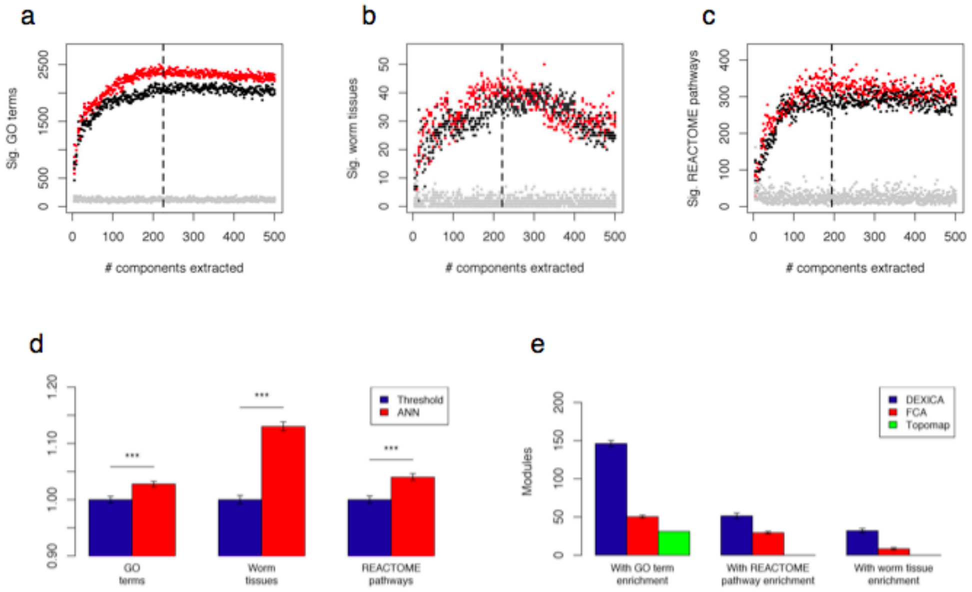


Figure 2.3. Gene module prediction optimization using ANN discretization

When searching for optimal parameters for gene module prediction (see Figure 2.2), we applied two different module discretization methods.   Figure 2.2 shows results using a fixed-threshold method (threshold = +/- 3), whereas here, a-c show results using an artificial neural network-based approach for generating discrete sets of genes for Gene Ontology term enrichment (a), *C. elegans* tissue enrichment (b), and REACTOME pathway enrichment (c). As in Figure 2.2a-d, black points show results from a compendium produced using a previously published preprocessing procedure[15]; red points show results for the best alternative preprocessing method that we tested; dashed lines indicate the point on the x-axis of each graph at which loess regression curves showed the greatest difference between red points and results from randomized control modules (grey points). d shows the mean performance of ANN discretization relative to fixed threshold discretization for all data points in a-c and Figure 2.2a-c. Error bars indicate SEM; all comparisons are statistically significant (*** = p < 2.2e-16). e shows the number of modules with significant enrichment for GO terms, *C. elegans* tissue annotations, and REACTOME pathway annotations for modules produced by DEXICA and FCA, and gene sets from the *C. elegans* gene expression topomap[25]; error bars in e indicate s.d. between repeat runs of DEXICA / FCA.

Gene module validation

To test the prediction that the independent components generated by DEXICA correspond to genetic regulatory modules, we checked each module for enrichment of regulatory sequences in the promoter regions and 3' untranslated regions (3'-UTRs) of module genes. To do this, we first generated a list of potential regulatory oligonucleotide sequences (called 'words') by applying the Mobydick algorithm(Bussemaker et al 2000) to the set of all predicted *C. elegans* promoter regions, which we defined as the region extending from the transcription start site to 2000 base pairs upstream. (Many *C. elegans* regulatory sequences are located in this interval; however, we note that this method will exclude potential promoter sequences located exclusively upstream or downstream of this region.) We generated a second oligonucleotide list using the set of all predicted *C. elegans* 3'-UTRs (see Supplementary Methods). We then calculated the statistical significance of the over- or under-representation of genes bearing each word in each gene module (see Methods), using the hypergeometric test and the Simes method(SImes 1986) for multiple hypothesis testing (alpha level = 0.05), to determine the number of significant modules. Across multiple runs of DEXICA, the mean number of gene modules containing significant promoter words and 3'-UTR words was 106.3 and 40.6, respectively, which was significantly greater than that produced by other module prediction methods we tested (p < 2.2e-16, Fig. 2.2e).

Because the ICA algorithm that we employed during module prediction, *fastICA*, converges to a final solution from a random starting point(Hyvarinen & Oja 2000), small differences typically exist in the output of different runs of the algorithm; these differences can be seen in the vertical spread of data points in figures 2.2a-d, and in the error bars of figure 2.2e. While others have reconciled such differences through a clustering approach applied to the output of numerous runs of the algorithm (so called "iterated ICA")(Engreitz et al 2010; Frigyesi et al 2006), when applied to our *C. elegans* Affymetrix compendium, we found that many of the final components generated by this method were highly correlated to one another, indicating non-independence and potential redundancy among the components (data not shown). We therefore sought to choose a single, high quality, *fastICA* run output to use as predicted gene

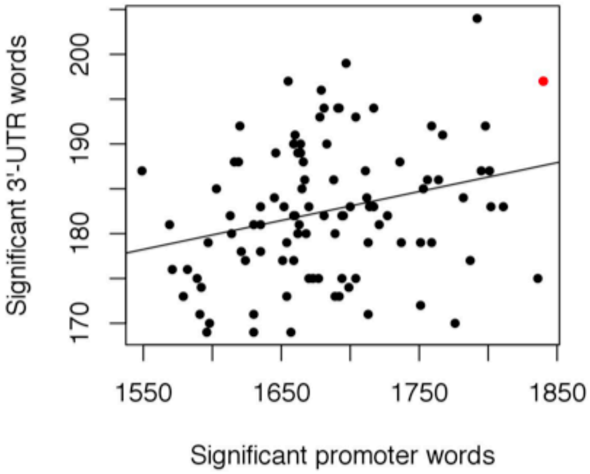


Figure 2.4. Significant words in predicted gene modules

We generated 100 sets of gene modules from the *C. elegans* microarray compendium using the optimal parameters indicated by the tests in Figures 2.2a-d and 2.3a-c. Among these module sets, there is a significant correlation between the total number of promoter words that are significant in at least one module and the number of 3'-UTR words that are significant in at least one module (R = 0.27, p = 6.5e-3). As our final *C. elegans* module set, we chose the set with the best mean rank in these two criteria (indicated by red point in figure).

modules. Because we considered word enrichment the most trustworthy measure of module quality, and because we observed a significant correlation (R = 0.27, p = 6.5E-3) between the total number of significant promoter words and the total number of significant 3'-UTR words in the results of different ICA runs with the same parameters (Fig. 2.4), as our final module set, we chose the run from a set of 100 with the best average rank in terms of significant promoter words and significant 3'-UTR words. This set ranked first in significant promoter words and third in significant 3'-UTR words.

Global properties of gene expression revealed by predicted gene modules

Gene modules are sets of genes that are co-expressed. Unexpectedly, during our analysis of 3'-UTR word enrichment within gene modules, we observed that some modules appeared to be enriched for genes with long 3'-UTRs. To determine if this trend was statistically significant, we calculated the mean 3'-UTR length of each hemi-module and determined a p-value for length bias using Student's t-test (Fig. 2.5a). Of the 418 hemi-modules, 65 contained a significant ($q < 0.1$) bias toward long 3'-UTR genes and 58 contained a bias toward short 3'-UTR genes.

To see if other gene properties were enriched in specific gene modules, we tested each hemi-module for over- and under-enrichment of genes appearing in operons and for genes with multiple splice forms. 21 hemi-modules were significantly enriched and 205 hemi-modules were significantly under-enriched for operon genes, and 81 hemi-modules were enriched and 80 hemi-modules were under-enriched for genes with multiple splice variants (Fig. 2.5b-c).

Control tests performed on the same module set but with randomly scrambled gene IDs produced no significant modules for any of the gene properties we tested (Fig. 2.6). Taken together, these results suggest that genetic regulatory modules tend to comprise genes with gross similarities in gene structure. This association, in turn, raises the possibility that these

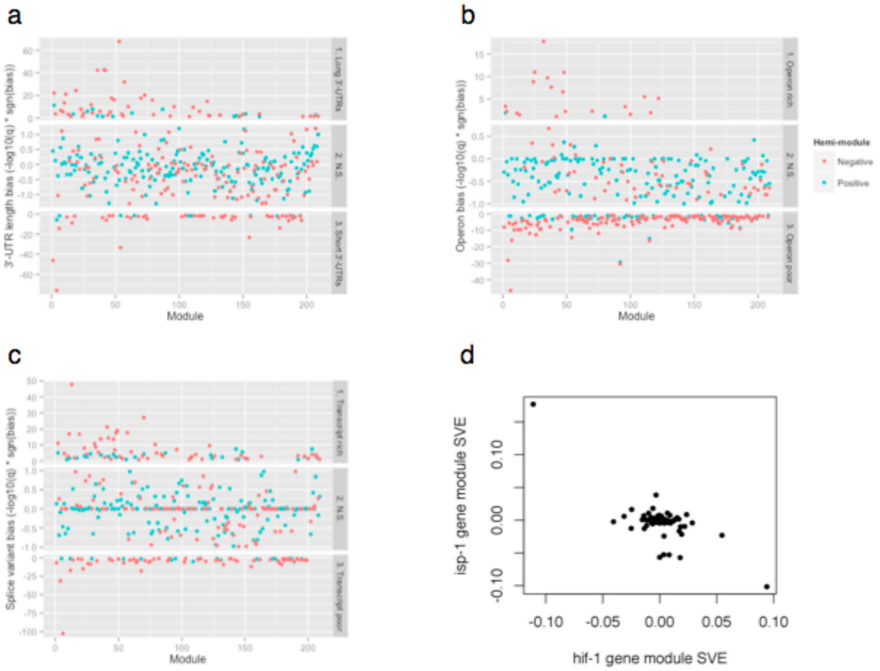


Figure 2.5. Biological implications of predicted gene modules

Discretization of predicted gene modules produces two sets of genes per module, which we refer to, based on the signs of their weights in the $\boldsymbol{S_g}$ matrix, as the positive and negative hemi-modules. We tested each hemi-module for biases in 3'-UTR length (A), in the number of genes transcribed in operons (B), and in the number of genes with multiple annotated splice variants (C). The y-axis in A-C indicates the strength and direction of bias; values shown are the $-\log_{10}$ of q-values, multiplied by the direction of the bias. D shows the signed variance explained (SVE) of the gene modules for two sets of gene expression fold changes, *hif-1(-)* vs. wild type, and *isp-1(-)* vs. wild type. SVE for these two datasets are significantly negatively correlated (R = -0.74, p = 2.4e-37). That is, genes turned up in *isp-1(-)* mutants are likely to be turned down in *hif-1(-)* mutants.

shared structural features (long 3’-UTRs, etc.) house important biological information, either for

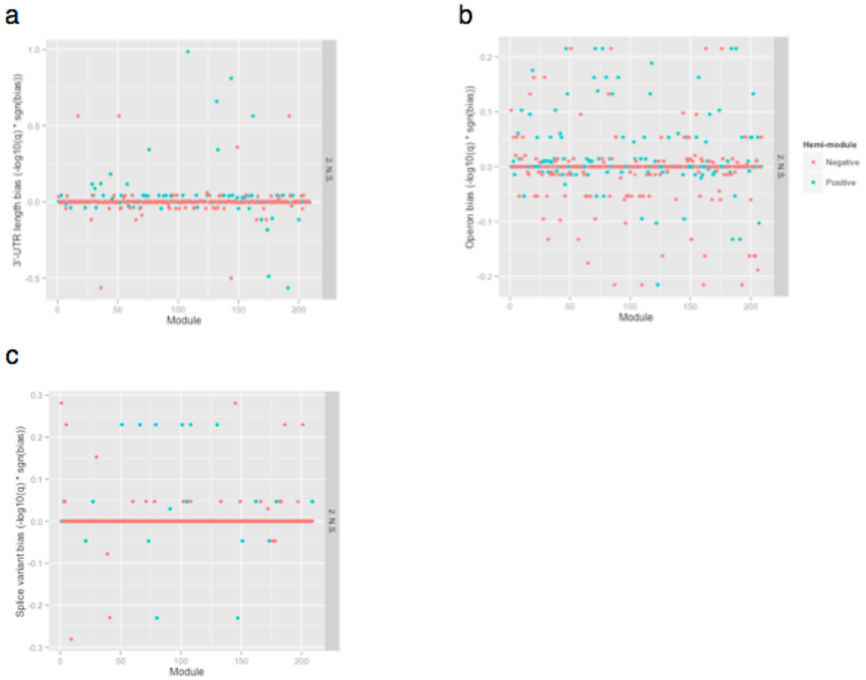
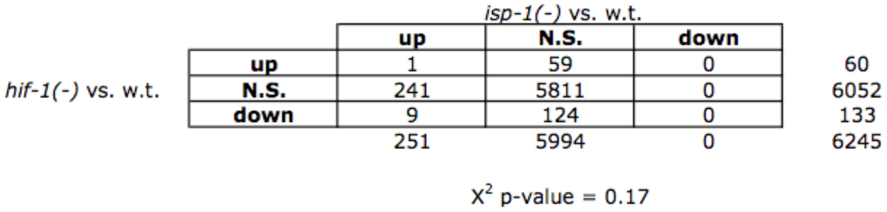


Figure 2.6. Randomized controls for module bias tests

We permuted the gene IDs in the gene module definition matrix, S_g, then repeated the statistical tests for 3-UTR length bias (A), operon gene content bias (B), and splice variant number bias (C). As in Fig. 2.5, the y-axis of each graph indicates the strength and direction of bias; values shown are the $–\log_{10}$ of q-values, multiplied by the direction of the bias.

gene regulation or gene function. Consistent with this idea, genes within operons are enriched in the set of *C. elegans* genes switched on during recovery from growth-arrested states(Zaslaver et al).

To test whether our predicted gene modules could provide biological insights into gene expression data, we used them to analyze published microarray datasets for *C. elegans* carrying mutations in *isp-1* (iron-sulfur protein, respiratory complex III)(Cristina et al 2009) and *hif-1* (hypoxia-inducible transcription factor)(Shen et al 2005). Reduction-of-function *isp-1* mutations extend lifespan in a *hif-1*-dependent fashion(Bell et al 2007; Hwang & Lee 2011; Lee et al 2010), but, unexpectedly, we found that the overlap among the significant genes of microarray measurements comparing each mutant to wild type was not statistically significant (X^2 test p-value = 0.17; Fig. 2.7). We computed the relative amount of variance that each gene module explains in the *hif-1* and *isp-1* microarray data sets, then multiplied the resulting values by -1 in those cases in which a module definition vector was negatively correlated with a set of microarray fold changes. The resulting quantity, which we refer to as signed variance explained (SVE), provides an indication of both the strength and the direction of change of each module in each microarray data set. Using this method, we observed a very strong correlation in gene module activities for *isp-1* mutants and those for *hif-1* mutants (Fig. 2.5d, R = -0.74, p = 4.1e-39). The correlation was negative, consistent with the interpretation that the life extension



		isp-1(-) vs. w.t.			
		up	**N.S.**	**down**	
hif-1(-) vs. w.t.	**up**	1	59	0	60
	N.S.	241	5811	0	6052
	down	9	124	0	133
		251	5994	0	6245

X^2 p-value = 0.17

Figure 2.7. Contingency table for isp-1(-), hif-1(-) microarray comparison

The *isp-1(-)* and *hif-1(-)* datasets we used in this study were conducted on different microarray platforms, and both datasets contained a substantial number of flagged (excluded due to low quality) data points, such that the number of complete pair-wise observations between the two sets was only 6245. The table shows the overlap between the number of significantly up-regulated, significantly down-regulated, and non-significant genes in the two datasets. The X^2 p-value for this table is 0.17; thus, there is not a significant degree of overlap among the significant genes of these two datasets.

observed when *isp-1* activity is reduced requires activation of gene expression by HIF-1. The strong gene module correlation between these datasets despite a lack of similarity among their most differentially expressed genes suggests that the role of HIF-1 in regulating the lifespan of *isp*-1 mutants may be to instigate small but coordinated expression changes in many genes, most of which fail significance tests for differential expression in one or both datasets. In general, it would be interesting to learn to what extent this situation, which would not be detected by many genetic or bioinformatic methods, has arisen during the evolution of gene circuits.

Generation of word modules

In our algorithm, a word is a predicted regulatory sequence. We observed that sets of significantly enriched words within gene module promoters often contain word pairs that are reverse compliments of each other. This increased our confidence that the independent components generated by DEXICA correspond to genetic regulatory modules and led us to hypothesize that ICA of a matrix comprising a significance level of each word in each hemi-module would reveal sets of words that "travel together" in the space of gene module promoters, i.e., regulatory sequence motifs.

To test this hypothesis, we created a new matrix, $\boldsymbol{X_w}$, comprising enrichment p-values for each word / hemi-module combination (see Methods). We then performed ICA on this matrix multiple times, varying the number of extracted components each time. We refer to the

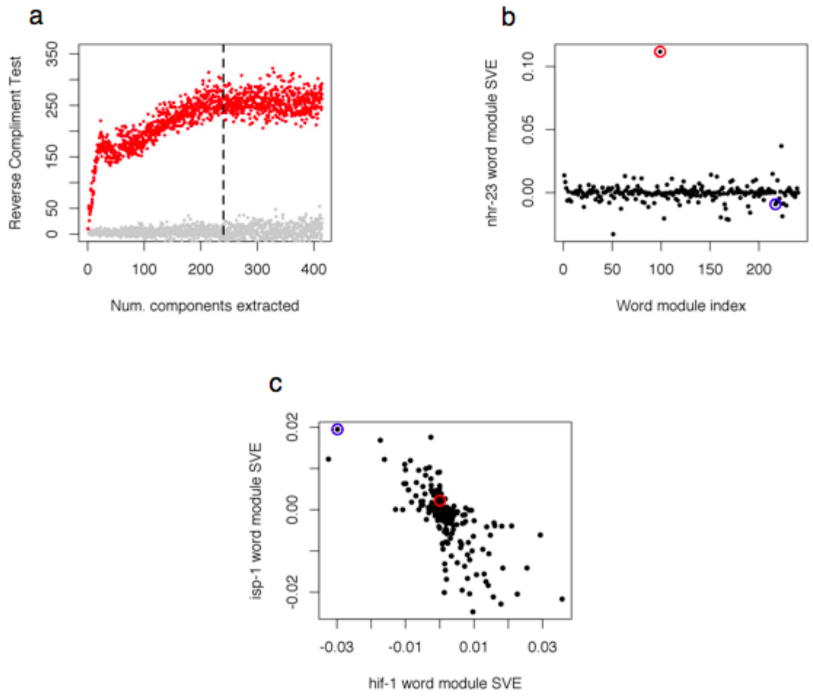


Figure 2.8. Word modules

(A) To determine the optimal number of word modules to extract from the word / gene module matrix, we varied the number of extracted components and assessed the total number of reverse complement word pairs that appeared in each set of discretized word modules. This value reached the largest increase over randomized controls (grey points) at 240, indicated by the dashed line. B) We calculated gene module signed variance explained (SVE) for gene expression fold changes in *nhr-23(RNAi)* samples compared to wild type from previously published data; gene module 99 accounts for over 10% of the variance explained by the word module set. C) We calculated SVE of gene expression fold changes from *hif-1(-)* vs. wild type and *isp-1(-)* vs. wild type using previously published data. SVE for these two data sets are significantly negatively correlated (R = -0.72, p = 4.1e-39). The red circles in (B) and (C) highlight word module 99; the blue circles highlight word module 217.

resulting independent components as word modules. To assess the quality of a set of word modules, we counted the total number of reverse compliment pairs that occurred within the same word module after ANN discretization. We found that this number achieved a maximum

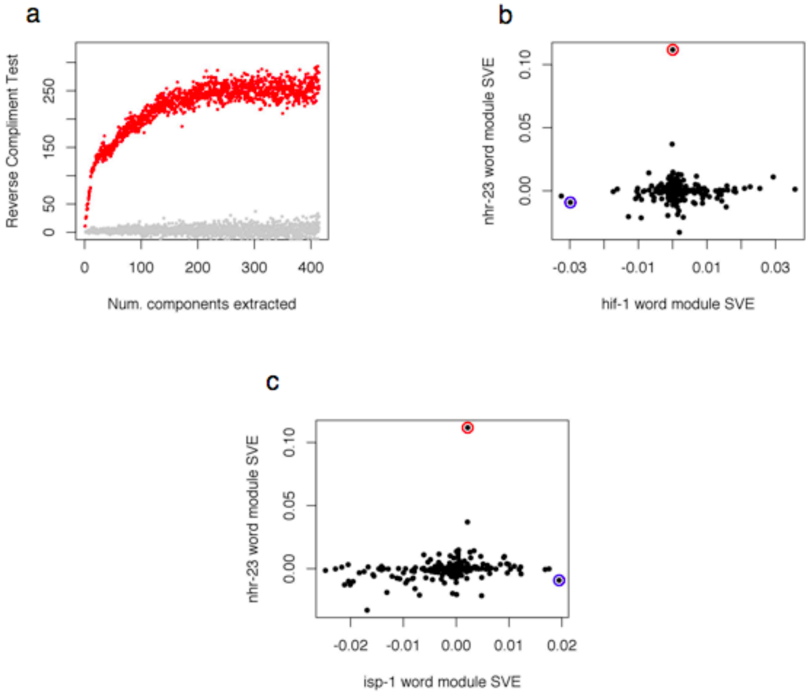


Figure 2.9. Word module control tests

a) Similar to Fig. 2.8a, in which we applied ANN discretization, we varied the number of extracted components and assessed the total number of reverse complement word pairs that appeared in each set of discretized word modules using threshold discretization (+/- 3 s.d.). b) Similar to Fig. 2.8c, we compared *nhr-23(-)* SVE to *hif-1(-)* SVE (b) and to *isp-1(-)* SVE (c). As in Fig. 2.8b-c, red circles highlight word module 99 and blue circles highlight word module 217.

value when 240 word modules were extracted (Fig. 2.8a). A similar result was observed using threshold discretization (Fig. 2.9a).

To test whether word modules generated in this manner resembled known transcription factor binding sites, we examined whether any word modules showed strong similarity to the canonical binding sites of four well-characterized transcription factors: DAF-16, HSF-1, NHR-23, and HIF-1. The canonical binding site of the first of these, DAF-16, was not present in the promoter word dictionary generated by Mobydick, which precluded it from membership in a word module, but the canonical binding sites for each of the other three factors were both present in the dictionary and among the top words in specific word modules (Table 1). Thus, ICA of word enrichment p-values in sets of co-regulated genes has the ability to recover sets of words with strong similarity to known transcription factor binding sites.

Prediction of transcriptional influence

ICA of the word / hemi-module p-value matrix, $\boldsymbol{X_w}$, produces a word module source matrix, $\boldsymbol{S_w}$, and a word module mixing matrix, $\boldsymbol{A_w}$ (Figure 2.1). Because the $\boldsymbol{A_w}$ matrix describes the relative enrichment of each word module in the promoters of each hemi-module, and the values in the gene module source matrix, $\boldsymbol{S_g}$, describe the degree of inclusion of each gene in each gene module, we hypothesized that the matrix product of these two matrices, appropriately transformed (see Methods), would reveal the relative transcriptional influence of each word module; that is, each potential regulatory sequence, on each gene. However, one property of ICA decomposition that must be taken into account before performing this operation is that the signs assigned to each independent component are arbitrary. For example, binding sites for a particular transcription factor that always activates gene expression may be enriched in a hemi-module termed “positive” in one gene module, and also enriched in a hemi-module termed “negative” in another gene module.

Sign ambiguity in gene modules, if left unresolved, could lead to inaccurate transcriptional influence predictions as this ambiguity would be transferred to the addends whose sums yield the values in the transcriptional influence matrix; addends with incorrectly (from the perspective of biological reality) opposing signs would serve to cancel each other in the summation step. To resolve sign ambiguity in gene modules, we transformed the gene module source matrix, $\boldsymbol{S_g}$, into a matrix with a separate row for each hemi-module and a column for each gene, $\boldsymbol{H_g}$. We used the absolute value of each gene's weight in each hemi-module to fill this matrix, placing zeros in cells of the matrix for each gene that was not a member of the corresponding hemi-module after ANN discretization. We then determined the matrix product of this matrix with the transpose of a normalized version of the $\boldsymbol{A_w}$ matrix. This produced a matrix, $\boldsymbol{X_{wg}}$, with word modules in rows and genes in columns, with each value in the matrix reflecting the weight of the word module across all of the hemi-modules in which the gene is a member. We interpret this value as the degree of transcriptional influence that each word module exerts on each gene in the data comprising the gene expression compendium.

To test whether our predictions of transcriptional influence agree with known transcription factor / gene associations, we compared the strongest genes associated with each word module to transcription factor target genes identified by the modENCODE project. Of the 240 word modules we predicted, 239 shared a statistically significant overlap with at least one of the transcription factors in the modENCODE data set (hypergeometric test, $q < 0.1$).

Transcriptional influence analysis of gene expression data

To test the utility of our transcriptional influence predictions in analyzing gene expression data, we first applied them to the analysis of a transcription factor perturbation experiment conducted on the same platform used to generate our predicted gene modules. For this test, we chose an experiment that measured gene expression changes induced by RNAi of the *C. elegans* nuclear hormone receptor *nhr-23* for two reasons, 1) the experiment was excluded from inclusion in the compendium because only 6 array hybridizations had been performed (we required at least 8 arrays from each experiment for inclusion in the compendium, see Methods), and, 2) the binding site of *nhr-23* has been characterized.

Calculation of signed variance explained (SVE) of the word modules for the fold changes between *nhr-23* and control samples using the $\boldsymbol{X_{wg}}$ matrix reveals that word module 99 shows by far the strongest change in activity (fig. 2.8b). This is the same word module that we identified when we looked for word-module matches to the NHR-23 binding site directly (discussed above). The three words most strongly associated with word module 99 have similar weights, ranging from 18.3 – 20.7; the fourth strongest word has a weight of 7.7. Each of the top three words matches the known *nhr-23* binding site, either directly or via reverse complementarity, suggesting that the word and gene associations defined by this word module are accurate (Table 1). To determine if the canonical binding site for NHR-23 would also be recovered using a standard method for transcription factor binding site prediction, we applied the RSA-Tools(Thomas-Chollier et al) *oligo-analysis* program (http://rsat.ulb.ac.be/oligo-analysis_form.cgi) to the most differentially expressed genes in the *nhr-23* dataset. None of the oligonucleotide sequences found to be enriched in the promoters of these genes by *oligo-analysis* matched the canonical NHR-23 binding site (see Supplemental data).

We next chose to test whether transcriptional influence predicted from microarray data generated with the Affymetrix platform could be useful for interpreting data from a different expression platform. For this test, we reanalyzed the *hif-1* and *isp-1* mutant gene expression fold changes described above. Calculation of signed variance explained for these fold revealed that word module 217 was the strongest positive word module in *isp-1* mutants, and the second strongest negative word module in *hif-1* mutants. Similar to the result with *nhr-23*, the top three words in word module 217 showed similar strengths, ranging from 26.8 – 28.9 and each of these words matched the known binding site of HIF-1. We also applied the *oligo-analysis* program to the most differentially expressed genes for both the *isp-1* and *hif-1* datasets. Similar to the result with the *nhr-23* dataset, none of the oligonucleotides reported to be enriched by this program matched the canonical HIF-1 binding site (see Supplemental data). Again, these results suggest that our method is able to detect functional transcription factor binding sites that other methods, which typically examine only the most strongly differentially expressed genes, may miss.

Discussion

Improvements in gene expression measurement technology have advanced to the point where subtle changes in gene expression between two conditions can be detected reliably, and a researcher is often faced with making sense of thousands of significant genes following statistical analysis of expression data. A common practice is to choose an arbitrary fold-change threshold, e.g., 2-fold, to limit a large list of significant genes to a more manageable size, but the biological justification for such a threshold is unclear, given that a small difference in the expression level of a gene, or the coordinated expression of many such genes, could potentially have a large impact on cellular physiology. To wit, our application of a common transcription factor binding site detection method to the top genes from the three different microarray experiments we examined (from *nhr-23*, *isp-1*, and *hif-1* mutants) did not recover the binding sites of the transcription factors expected to be most influential in each data set.

Our method of analyzing gene expression data using transcriptional influence does not rely on fold change thresholds or on thresholds of statistical significance. Instead, the entire spectrum of fold changes from a gene expression experiment informs the activity level of each word module. This not only allows one to look for biologically significant patterns comprising genes with small fold changes (as we infer occurs for HIF-1's regulation of genes in *isp-1* mutants), it also allows one to apply transcriptional influence predictions to analyze gene expression data from a single contrast, e.g. a single two-color microarray or an RNA-Seq experiment with a single control sample and a single test sample. This level of sensitivity could be useful for conducting large-scale gene expression screens, as both cost and effort would be reduced dramatically.

Whereas ICA has been applied to the prediction of gene modules before, we could find no examples in the literature of its application to the recovery of transcription factor binding sites from co-regulated genes. The successful recovery of known binding sites via ICA, based on a library of potential regulatory sequences created with genomic sequence data, serves as a proof of principle of the utility of this algorithm in this capacity. Combined with the improved ability to discretize independent components provided by our artificial neural network approach, we expect that our results will spur exploration into additional applications of ICA to the analysis of biological data. In addition, it will stimulate many specific, testable hypotheses about the roles of specific transcription factors in biological processes. For example, had we not known previously that HIF-1 regulated life extension in *isp-1* mutants, we would have generated this hypothesis upon observing strong representation of the same gene and word modules in the *isp-1* and *hif-1* microarray datasets.

A potential limitation of our method is that estimations for transcriptional influence can only be made for transcription factors that have differing levels of activity among the experimental conditions represented in the compendium. A transcription factor with a constant level of activity across all compendium microarrays would be invisible to our method. As new areas of research are explored and new experiments are published, however, transcriptional influence can be recalculated to add new word modules or to improve influence estimations for word modules already present.

While three of the four canonical transcription factor binding sites that we examined had good matches to word modules, both in terms of the oligo nucleotide sequences comprising the word modules and in terms of the predicted transcriptional influence of the word modules, neither of the exact matches to the canonical binding site of the fourth factor, DAF-16 [which has the canonical binding site T(G/A)TTTAC(Furuyama et al 2000; Murphy et al 2003)], was present in the promoter word dictionary created by Mobydick. A word module comprising longer, similar sequences to the canonical DAF-16 site was generated by our algorithm, but few genes were predicted to be strongly influenced by this word module and those that were did not match known DAF-16 target genes. While the lack of strong transcriptional influence predictions between known DAF-16 target genes and a word module that resembles the DAF-16 binding site could be due to insufficient sampling of perturbations involving this factor in the compendium, we believe a more likely explanation stems from our method for calculating word enrichment among module promoters. We calculated the p-value for module-wise enrichment based simply on the presence or absence of each word in each gene's promoter. Thus, words that are present many times in a gene's promoter do not contribute anything more to the p-value calculation than words that are present only once. The canonical DAF-16 binding site occurs in approximately 50% of all 2k-bp gene promoters, but in the 12 genes with the largest expression changes in *daf-2* mutants, in which DAF-16 becomes activated, Zhang et al.(Zhang et al), found that the mean number of occurrences of the DAF-16 binding site is 5.1. Thus, a single copy of the DAF-16 binding site may be insufficient to confer regulation by this factor. A modification of our method that uses gene-wise promoter word enrichment rather than presence vs. absence may prove more to be more accurate for predicting the transcriptional influence of factors with highly abundant binding sites, such as DAF-16. This is a question we plan to address in future work.

In addition to its utility in analyzing gene expression data, the transcriptional influence matrix can be used to identify transcription factor target genes, providing that word modules can successfully be mapped to transcription factors. In our comparison of word module target genes to genes identified as potential targets of transcription factors by the modENCODE project, we found that most word modules shared statistically significant overlap with multiple transcription factors. We attribute this to the general binding promiscuity that transcription factors in the modENCODE datasets demonstrate. Future work based on matching binding site sequences, rather than target genes, may prove to be more fruitful for mapping word modules to specific transcription factor(s) on a large scale.

Methods

Compendium construction

To build our compendium of 1386 *C. elegans* Affymetrix arrays, we first downloaded all CEL files with the appropriate platform ID (GPL200) from the GEO database available on March 1, 2014, excluding those for which the organism was not *C. elegans* and the sample type was not RNA. We excluded arrays from experiments for which fewer than 8 hybridizations were performed in order to mitigate the effect that under-sampled conditions might have on predicted modules. We then performed a quality control step using the quality assessment functions provided in the *simpleAffy* (v2.40.0) R package (http://bioinformatics.picr.man.ac.uk/simpleaffy/), discarding arrays that did not meet the quality thresholds recommended in the *simpleAffy* documentation.

We generated expression values for probesets separately for each experiment (determined by GEO series IDs) using the RMA preprocessing procedure provided in the *affy* (v1.40.0) R package (Gautier et al 2004), then used the *bias* (v0.0.5) R package(Eklund & Szallasi 2008) to remove intensity-dependent biases in expression levels. We then concatenated the expression matrices for each experiment into a single matrix. Next, we either performed between-experiment quantile normalization(Bolstad et al 2003) on the entire matrix using the *limma* (v3.18.13) R package(Smyth 2005), or omitted this step, depending on preprocessing method to be tested. Finally, we scaled and centered the arrays and centered the genes such that the mean of each row and column were zero and the standard deviation of each array was 1.

Conducting ICA

To conduct ICA of the gene expression matrix, we used the *fastICA* (v1.2-0) R package (http://CRAN.R-project.org/package=fastICA) with default parameters except for the “method” parameter, which we set to “C” to increase computational speed, and the “row.norm” parameter, which we set to ”TRUE” in order to balance the total compendium variance between genes with subtle changes in expression values and those with large changes in expression values. We used the same parameters to conduct ICA of the word / module p-value matrix.

Discretization of independent components

To convert independent components to discrete sets of genes, we employed two methods. In the first, for each component, we assigned all genes with a weight <= -3 to the negative hemi-module, and all genes with a weight >= 3 to the positive hemi-module. In the second, we created an artificial neural network using the *neuralnet* (v1.32) R package (http://CRAN.R-project.org/package=neuralnet) to predict positive and negative discretization thresholds for each independent component, based on the component’s skewness and kurtosis (see

Supplemental Methods), then assigned genes whose weights exceeded these thresholds to the corresponding hemi-modules.

Obtaining gene annotations and additional microarray data

To obtain GO term and REACTOME pathway annotations for genes we used the *biomaRt* (v2.18.0) R package(Durinck et al 2005; Durinck et al 2009), using the *ensembl* mart for data retrieval. To obtain tissue annotations for *C. elegans* genes, we downloaded all available data from the GFP Worm database (http://gfpweb.aecom.yu.edu/)(Hunt-Newbury et al 2007), which contains annotated expression patterns of promoter::GFP fusion constructs; in total, this dataset provided annotations for 1821 genes across 89 tissue types (*n.b.,* we considered the same tissue in different development stages to be distinct tissue types). To obtain expression data from a different platform for use in optimization of gene module prediction, we downloaded the fold change matrices for all GEO series conducted on the Washington University *C. elegans* 22k 60-mer array (GEO platform ID: GPL4038), a two-color spotted array platform, and concatenated these column-wise into a single matrix. To obtain microarray data for *nhr-23(RNAi)*, we downloaded gene fold changes for the GEO series GSE32031, which contains three control samples and three *nhr-23(RNAi)* samples(Kouns et al 2011); gene fold changes were calculated using the GEO2R web service (http://www.ncbi.nlm.nih.gov/geo/geo2r/). To obtain fold changes for *isp-1* mutants, we used data previously published by our group in which *isp-1(qm150)* mutants were compared to wild type controls(Cristina et al 2009). To obtain fold changes for *hif-1* mutants, we used the *maanova* (v1.33.2) R package (http://research.jax.org/faculty/churchill) and data previously published by Shen, et al.(Shen et al 2005), to calculate the induced gene fold changes upon mutation of *hif-1*.

Optimizing gene module prediction

To optimize gene module prediction, we performed ICA with different parameters, varying the number of extracted components from 5 to 500 by increments of 5 and varying the compendium between one generated with between-experiment quantile normalization and one generated without this step. For each parameter combination, we repeated ICA 5 times, for a total of 1000 ICA runs.

We tested the biological validity of the independent components generated by each ICA run by determining the number of annotations that were enriched in at least one hemi-module. To make this determination, we first calculated a p-value for the enrichment of genes associated with each annotation term in each hemi-module using the hypergeometric test. We then applied the Simes method(SImes 1986) for multiple hypothesis testing (alpha = 0.05) to the set of p-values for each annotation term; failure of this test indicates that at least one of the p-values is truly significant. To verify the accuracy of our module quality statistics, we repeated all tests using module definition matrices in which gene IDs had been randomly shuffled.

To test the ability of a set of independent components to represent data from a different microarray platform, we first projected the data from the second platform onto the independent components (see below). This operation produces a mixing matrix, which may be interpreted as describing the weight of each independent component in each of the projected microarrays. We then attempted to recover the original data by determining the dot product of the module definition matrix and the mixing matrix. We compared this matrix with the original matrix and calculated the root mean squared deviation (RMSD) between the two. We normalized this value by dividing by the range of values between the two matrices, resulting in NRMSD.

Projection onto independent components and calculation of SVE

To project a data vector, ***x***, such as a set of gene expression fold changes, onto a set of modules, we used the scalar projection method, in which a mixing vector, ***a***, is calculated from the dot product of the data vector and the transpose of the unit vectors comprising the module definitions. The resulting mixing vector, ***a***, provides an indication of the weight of each module definition vector in the projected data, ***x***. Projection of a data matrix, ***X***, which generates a mixing matrix, ***A***, was carried out using the same procedure.

To calculate signed variance explained (SVE), we calculated the relative variance explained (VE) for each module from ***a*** then multiplied these values, which are strictly positive, by -1 in each case where ***a*** < 0 to obtain SVE.

Statistical testing of module 3’-UTR length bias

We observed that *C. elegans* 3’-UTR lengths are approximately log-normally distributed (Figure 2.10). Therefore, we chose to use the log of each 3’-UTR length in our calculations to allow the use of parametric statistical tests, such as Student’s t-test. For those genes with multiple annotated 3’-UTRs, we determined the log of the individual 3’-UTR lengths and used the mean of these numbers for the gene’s 3’-UTR length.

In our statistical test for 3’-UTR length biases in predicted modules, we first calculated the weighted mean *C. elegans* 3’-UTR length. We weighted each gene’s contribution to this mean by the frequency with which it appears in our predicted modules in order to adjust for different propensities for module inclusion by different genes. We then used one-sample t-tests to calculate p-values for whether the mean 3’-UTR length of each hemi-module differs significantly from the weighted mean *C. elegans* 3’-UTR length. We used the Benjamini-Hochberg procedure on these p-values to control the false discovery rate at a level of 0.1.

Generation of word / hemi-module p-value matrix

To generate a matrix for use in word module prediction via ICA, we first created gene sets from the module definition matrix, $\boldsymbol{S_g}$, using ANN discretization. This produced two gene sets (which we refer to as hemi-modules) per gene module, for a total of 418. We then calculated a hypergeometric probability for each word in each hemi-module, using the frequency of genes bearing a particular word in their promoter in the hemi-module, the frequency of such genes in the compendium, the number of genes in the hemi-module, and the number of genes not in the hemi-module as the *q, m, k,* and *n* input parameters, respectively, to the *phyper()* function of the *stats* (v3.0.3) R package (http://www.R-project.org/).

We used these p-values to populate a matrix with a column for each hemi-module and a row for each word in our promoter dictionary. For under-represented words, we entered the *log(p-value x 2)* in the matrix, and for over-represented words we entered the *–log(p-value x 2).* The multiplication by 2 corrected for two-tailed testing.

Prediction of transcriptional influence

Because word modules are created based upon genes assigned to hemi-modules, the final step in our process of predicting transcriptional influence required us to transform the gene / module definition matrix, $\boldsymbol{S_g}$, into a gene / hemi-module matrix., $\boldsymbol{H_g}$. This operation consisted of performing a row-wise concatenation of two copies of the module definition matrix, such that each hemi-module appeared in its own row. We then set the weights of all non-hemi-module genes within each row to zero, to prevent such genes from exerting an effect on transcriptional influence.

Comparing transcriptional influence predictions to modENCODE data

To compare transcriptional influence predictions to known transcription factor / DNA binding interactions, we downloaded all *C. elegans* transcription factor binding data from the modENCODE database (http://www.modencode.org/) available on March 1, 2014. This dataset comprised 69,860 DNA binding sites from 136 submissions. For each DNA binding site, we annotated a gene as being a target of a transcription factor if the midpoint of one of the transcription factor's binding sites occurred within the gene's 2-kb promoter region. We then calculated a hypergeometric p-value for each word module / transcription factor combination by comparing the genes with the 100 largest weights for a word module in the transcriptional influence matrix, $\boldsymbol{X}_{wg}$, to the set of genes annotated as transcription factor targets. We then calculated q-values from the resulting set of hypergeometric p-values, and considered a result significant if $q < 0.1$.

Supplemental methods

Construction of Mobydick dictionaries

To construct promoter and 3'-UTR dictionaries, we ran the Mobydick(Bussemaker et al 2000) program once on the complete set of *C. elegans* promoters, using DNA sequence from the transcription start site to 2000 b.p. upstream for each gene, and again on the complete set of 3'-UTRs with lengths of at least 25 n.t. Sequences were obtained using the *biomaRt* (ver 2.14.0) R package(Durinck et al 2005; Durinck et al 2009). Application of Mobydick to promoter sequences produced a dictionary of 5230 words, and application to 3'-UTR sequences produced a dictionary of 968 words.

Calculation of significance of 3'-UTR word enrichment

Because 3'-UTRs differ in length, and because gene modules show a tendency toward inclusion of genes with similar length 3'-UTRs, calculation of the enrichment of 3'-UTR words in module genes required a length-normalization step. To achieve this, we applied the method described van Helden, et al.(van Helden et al 1998) Briefly, we determined the per nucleotide frequency of each word in the entire set of 3'-UTRs, then used the binomial distribution to determine whether each word occurs more often than expected by random chance in a sequence, given the number of occurrences of the word in the sequence and the sequence length. We then applied the Holm-Bonferroni correction to the resulting p-values and marked all words with a corrected p-value < 0.5 as present in the 3'-UTR.

Generation of artificial neural network for independent component dicretization

To create an artificial neural network for use in discretization of independent components, we first generated simulated data to use as test, training, and validation sets. We generated this data by first randomly permuting the expression values of 100 arrays comprising our *C. elegans* microarray compendium column-wise to create a background devoid of non-random signal, but with a similar gene expression value distribution to real data.

Into this random background we inserted simulated gene modules by first picking a gene to use as a module seed pattern, then changing the expression values of other genes such that they positively or negatively correlated with the expression values of this gene across all or a subset of arrays. We varied the number of genes comprising the simulated module, the strength of adherence of each gene to the seed pattern, the fraction of genes within the module with positive and negative correlation to the seed pattern, and the number of arrays in which this correlation existed. In all, we generated over 10,000 random modules and inserted them into separate sets of random background arrays, so that each array set would contain a single non-random module.

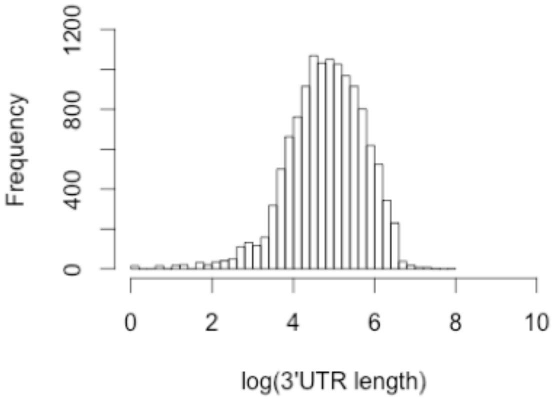


Figure 2.10. Distribution of C. elegans 3'-UTR lengths

The distribution of *C. elegans* 3'-UTR lengths is approximately log-normal.

We then attempted to recover each simulated module using ICA. We extracted a single component from each simulated array set and deemed the extraction successful if 3 of the top 5 most strongly weighted genes in this component were in the simulated module. For successful extractions, we calculated the optimal discretization thresholds for the positive and negative hemi-modules, as well as the skewness and kurtosis of the module definition vector using the *moments* (v0.13) R package (http://CRAN.R-project.org/package=moments).

Using this data, we trained an artificial network to predict the optimal discretization thresholds for an independent component from the skewness and kurtosis of its gene weights using the *neuralnet* (v1.32) R package (http://CRAN.R-project.org/package=neuralnet). We generated another simulated module set in the same manner as the first to use as a test set, and varied the architecture of the artificial neural network until the prediction performance reached a maximum value. This occurred when the artificial neural network contained two hidden layers, each with 11 nodes. We confirmed that the artificial neural network was not over-fit to the test

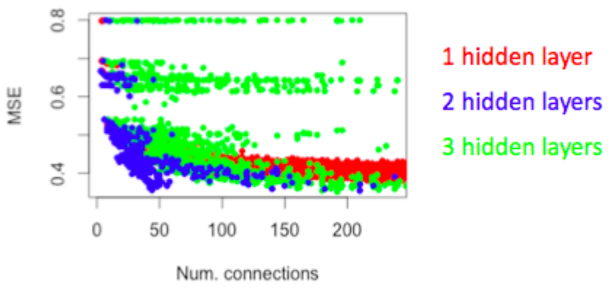


Figure 2.11. Optimization of artificial neural network for discretization

We tested the effect of various parameters on the prediction performance of artificial neural network discretization of simulated modules. Shown here are the results of varying the number of hidden layers in the network and the number of nodes in each layer; the x-axis shows the total number of connections in the network, the y-axis shows the mean squared error of each network. Colors indicate the number of hidden layers.

set by measuring its performance in a third set of simulated data, the validation set. Performance on this set was similar to that on the test set. The structure of this artificial neural network is shown in Figure 2.12; an R data file containing the artificial neural network is available for download on our website (http://kenyonlab.ucsf.edu/data/ann.11.11.Rdata).

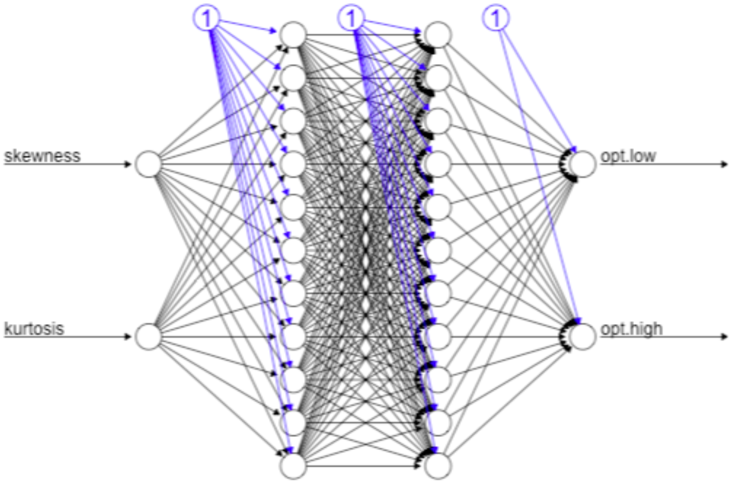


Figure 2.12. Schematic of optimized artificial neural network for IC discretization

The figure shows the structure of the best performing ANN generated by our optimization tests. The skewness and kurtosis of the source matrix weights (rows of the S matrix) of an independent component are used as input to the network, and the predicted optimal discretization thresholds are generated as output.

References

2004. The ENCODE (ENCyclopedia Of DNA Elements) Project. *Science (New York, N.Y* 306:636-40

Aittokallio T, Kurki M, Nevalainen O, Nikula T, West A, Lahesmaa R. 2003. Computational strategies for analyzing data in gene expression microarray experiments. *J Bioinform Comput Biol* 1:541-86

Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, et al. 2000. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nature genetics* 25:25-9

Bansal M, Belcastro V, Ambesi-Impiombato A, di Bernardo D. 2007. How to infer gene networks from expression profiles. *Molecular systems biology* 3:78

Bar-Joseph Z, Gerber GK, Lee TI, Rinaldi NJ, Yoo JY, et al. 2003. Computational discovery of gene modules and regulatory networks. *Nat Biotechnol* 21:1337-42

Barrett T, Troup DB, Wilhite SE, Ledoux P, Evangelista C, et al. 2011. NCBI GEO: archive for functional genomics data sets--10 years on. *Nucleic acids research* 39:D1005-10

Bell EL, Klimova TA, Eisenbart J, Schumacker PT, Chandel NS. 2007. Mitochondrial reactive oxygen species trigger hypoxia-inducible factor-dependent extension of the replicative life span during hypoxia. *Molecular and cellular biology* 27:5737-45

Bergmann S, Ihmels J, Barkai N. 2003. Iterative signature algorithm for the analysis of large-scale gene expression data. *Phys Rev E Stat Nonlin Soft Matter Phys* 67:031902

Bolstad BM, Irizarry RA, Astrand M, Speed TP. 2003. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics (Oxford, England)* 19:185-93

Bussemaker HJ, Li H, Siggia ED. 2000. Building a dictionary for genomes: identification of presumptive regulatory sites by statistical analysis. *Proc Natl Acad Sci U S A* 97:10096-100

Chatziioannou A, Moulos P, Kolisis FN. 2009. Gene ARMADA: an integrated multi-analysis platform for microarray data implemented in MATLAB. *BMC bioinformatics* 10:354

Chiappetta P, Roubaud MC, Torresani B. 2004. Blind source separation and the analysis of microarray data. *J Comput Biol* 11:1090-109

Comon P. 1994. Independent Component Analysis: a new concept? *Signal Processing* 36:287-314

Cristina D, Cary M, Lunceford A, Clarke C, Kenyon C. 2009. A regulated response to impaired respiration slows behavioral rates and increases lifespan in Caenorhabditis elegans. *PLoS genetics* 5:e1000450

Croft D, O'Kelly G, Wu G, Haw R, Gillespie M, et al. 2011. Reactome: a database of reactions, pathways and biological processes. *Nucleic acids research* 39:D691-7

Dalma-Weiszhausz DD, Warrington J, Tanimoto EY, Miyada CG. 2006. The affymetrix GeneChip platform: an overview. *Methods in enzymology* 410:3-28

Durinck S, Moreau Y, Kasprzyk A, Davis S, De Moor B, et al. 2005. BioMart and Bioconductor: a powerful link between biological databases and microarray data analysis. *Bioinformatics (Oxford, England)* 21:3439-40

Durinck S, Spellman PT, Birney E, Huber W. 2009. Mapping identifiers for the integration of genomic datasets with the R/Bioconductor package biomaRt. *Nat Protoc* 4:1184-91

Eklund AC, Szallasi Z. 2008. Correction of technical bias in clinical microarray data improves concordance with known biological information. *Genome biology* 9:R26

Engreitz JM, Daigle BJ, Jr., Marshall JJ, Altman RB. 2010. Independent component analysis: mining microarray data for fundamental human gene expression modules. *J Biomed Inform* 43:932-44

Frigyesi A, Veerla S, Lindgren D, Hoglund M. 2006. Independent component analysis reveals new and biologically significant structures in micro array data. *BMC bioinformatics* 7:290

Furuyama T, Nakazawa T, Nakano I, Mori N. 2000. Identification of the differential distribution patterns of mRNAs and consensus binding sequences for mouse DAF-16 homologues. *Biochem J* 349:629-34

Gautier L, Cope L, Bolstad BM, Irizarry RA. 2004. affy--analysis of Affymetrix GeneChip data at the probe level. *Bioinformatics (Oxford, England)* 20:307-15

Gerstein MB, Lu ZJ, Van Nostrand EL, Cheng C, Arshinoff BI, et al. 2010. Integrative analysis of the Caenorhabditis elegans genome by the modENCODE project. *Science (New York, N.Y* 330:1775-87

Gong T, Xuan J, Wang C, Li H, Hoffman E, et al. 2007. Gene module identification from microarray data using nonnegative independent component analysis. *Gene Regul Syst Bio* 1:349-63

Hunt-Newbury R, Viveiros R, Johnsen R, Mah A, Anastas D, et al. 2007. High-throughput in vivo analysis of gene expression in Caenorhabditis elegans. *PLoS Biol* 5:e237

Hwang AB, Lee SJ. 2011. Regulation of life span by mitochondrial respiration: the HIF-1 and ROS connection. *Aging (Albany NY)* 3:304-10

Hyvarinen A, Oja E. 2000. Independent component analysis: algorithms and applications. *Neural Netw* 13:411-30

Hyvärinen A, Oja E. 2000. Independent Component Analysis: Algorithms and Application. *Neural Networks* 13:411-30

Ihmels J, Bergmann S, Barkai N. 2004. Defining transcription modules using large-scale gene expression data. *Bioinformatics (Oxford, England)* 20:1993-2003

Kellis M, Wold B, Snyder MP, Bernstein BE, Kundaje A, et al. 2014. Defining functional DNA elements in the human genome. *Proc Natl Acad Sci U S A* 111:6131-8

Kim SK, Lund J, Kiraly M, Duke K, Jiang M, et al. 2001. A gene expression map for Caenorhabditis elegans. *Science (New York, N.Y* 293:2087-92

Kong W, Vanderburg CR, Gunshin H, Rogers JT, Huang X. 2008. A review of independent component analysis application to microarray gene expression data. *Biotechniques* 45:501-20

Kouns NA, Nakielna J, Behensky F, Krause MW, Kostrouch Z, Kostrouchova M. 2011. NHR-23 dependent collagen and hedgehog-related genes required for molting. *Biochem Biophys Res Commun* 413:515-20

Leach M. 2004. Gene expression informatics. *Methods Mol Biol* 258:153-65

Lee HC, Lai K, Lorenc MT, Imelfort M, Duran C, Edwards D. 2012. Bioinformatics tools and databases for analysis of next-generation sequence data. *Brief Funct Genomics* 11:12-24

Lee SI, Batzoglou S. 2003. Application of independent component analysis to microarrays. *Genome biology* 4:R76

Lee SJ, Hwang AB, Kenyon C. 2010. Inhibition of respiration extends C. elegans life span via reactive oxygen species that increase HIF-1 activity. *Curr Biol* 20:2131-6

Li H, Sun Y, Zhan M. 2007. The discovery of transcriptional modules by a two-stage matrix decomposition approach. *Bioinformatics (Oxford, England)* 23:473-9

Liebermeister W. 2002. Linear modes of gene expression determined by independent component analysis. *Bioinformatics (Oxford, England)* 18:51-60

Michoel T, De Smet R, Joshi A, Marchal K, Van de Peer Y. 2009. Reverse-engineering transcriptional modules from gene expression data. *Ann N Y Acad Sci* 1158:36-43

Murphy CT, McCarroll SA, Bargmann CI, Fraser A, Kamath RS, et al. 2003. Genes that act downstream of DAF-16 to influence the lifespan of Caenorhabditis elegans. *Nature* 424:277-83

Niu W, Lu ZJ, Zhong M, Sarov M, Murray JI, et al. 2011. Diverse transcription factor binding features revealed by genome-wide ChIP-seq in C. elegans. *Genome Res* 21:245-54

Pham TH, Satou K, Ho TB. 2004. Mining yeast transcriptional regulatory modules from factor DNA-binding sites and gene expression data. *Genome Inform* 15:287-95

Roy S, Ernst J, Kharchenko PV, Kheradpour P, Negre N, et al. 2010. Identification of functional elements and regulatory circuits by Drosophila modENCODE. *Science (New York, N.Y* 330:1787-97

Rubinstein R, Simon I. 2005. MILANO--custom annotation of microarray results using automatic literature searches. *BMC bioinformatics* 6:12

Rustici G, Kolesnikov N, Brandizi M, Burdett T, Dylag M, et al. ArrayExpress update--trends in database growth and links to data analysis tools. *Nucleic acids research* 41:D987-90

Segal E, Battle A, Koller D. 2003a. Decomposing gene expression into cellular processes. *Pac Symp Biocomput*:89-100

Segal E, Shapira M, Regev A, Pe'er D, Botstein D, et al. 2003b. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature genetics* 34:166-76

Segal E, Yelensky R, Koller D. 2003c. Genome-wide discovery of transcriptional modules from DNA sequence and gene expression. *Bioinformatics (Oxford, England)* 19 Suppl 1:i273-82

Shen C, Nettleton D, Jiang M, Kim SK, Powell-Coffman JA. 2005. Roles of the HIF-1 hypoxia-inducible factor during hypoxia response in Caenorhabditis elegans. *The Journal of biological chemistry* 280:20580-8

SImes RJ. 1986. An Improved Bonferroni Procedure for Multiple Tests of Significance. *Biometrika* 73:751-4

Smyth GK. 2005. Limma: linear models for microarray data. In *Bioinformatics and Computational Biology Solutions using R and Bioconductor*, ed. R Gentleman, V Carey, S Dudoit, R Irizarry, W Huber, pp. 397-420. New York: Springer

Soinov LA, Krestyaninova MA, Brazma A. 2003. Towards reconstruction of gene networks from expression data by supervised learning. *Genome biology* 4:R6

Thomas-Chollier M, Defrance M, Medina-Rivera A, Sand O, Herrmann C, et al. RSAT 2011: regulatory sequence analysis tools. *Nucleic acids research* 39:W86-91

van Helden J, Andre B, Collado-Vides J. 1998. Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *J Mol Biol* 281:827-42

Van Nostrand EL, Kim SK. 2013. Integrative analysis of C. elegans modENCODE ChIP-seq data sets to infer gene regulatory interactions. *Genome Res* 23:941-53

Vermeirssen V, Joshi A, Michoel T, Bonnet E, Casneuf T, Van de Peer Y. 2009. Transcription regulatory networks in Caenorhabditis elegans inferred through reverse-engineering of gene expression profiles constitute biological hypotheses for metazoan development. *Mol Biosyst* 5:1817-30

Wang Y, Joshi T, Zhang XS, Xu D, Chen L. 2006. Inferring gene regulatory networks from multiple microarray datasets. *Bioinformatics (Oxford, England)* 22:2413-20

Xia X, McClelland M, Wang Y. 2005. WebArray: an online platform for microarray data analysis. *BMC bioinformatics* 6:306

Zaslaver A, Baugh LR, Sternberg PW. Metazoan operons accelerate recovery from growth-arrested states. *Cell* 145:981-92

Zhang P, Judy M, Lee SJ, Kenyon C. Direct and indirect gene regulation by a life-extending FOXO protein in C. elegans: roles for GATA factors and lipid gene regulators. *Cell Metab* 17:85-100

Zhang Y, Szustakowski J, Schinke M. 2009. Bioinformatics analysis of microarray data. *Methods Mol Biol* 573:259-84

Chapter 3: Future Directions

Accurate predictions of transcriptional influence will facilitate the construction of continuous genetic regulatory networks, but the utility of the work described in Chapter 2 could be far more reaching. The predicted gene modules, which serve as an intermediate data structure in the algorithm for predicting transcriptional influence, may themselves reveal new biological processes or provide more inclusive lists of participating genes for known processes. Both the gene modules and transcriptional influence predictions may be used to analyze existing data sets, possibly shedding new light on the biological questions those data sets address. In this chapter, I discuss these and other possible applications of the work described in Chapter 2, and describe what I view are the most important next steps that should be taken in this area of research.

Construction of an online resource for data analysis

The most pressing need, in my opinion, is to make the work described in Chapter 2 accessible to the wider research community through the construction of an online resource for data analysis. Without such a resource, neither the gene modules nor the transcriptional influence predictions will see widespread use. Such a resource should allow a user to enter gene expression data (i.e., a list of gene fold changes), or even a simple list of genes that are interesting to the user, and the resource should show the user which gene modules and which word modules are significant in their input data. In addition, the resource could inform the user of microarrays (and the underlying perturbations) that are highly similar to their input data in terms of gene and word modules. In the initial version of the resource, the pool of available microarrays for this kind of analysis could be limited to the compendium used to construct the gene and word modules, but future versions of the resource could include microarrays from other platforms and RNA-seq data.

Additional future improvements to the online resource could be to expand predicted modules to other species (see below), allow cross-species data analysis through gene orthology relationships, and expand the types of annotations able to be analyzed. The method used to construct word modules relies on a simple binary annotation matrix, in which genes appear in rows and words in columns; if a word appears in the promoter region of a gene, the corresponding cell of this matrix contains a 1, and a 0 otherwise. This same kind of matrix could be constructed for any kind of gene annotation. For example, a matrix of genes and tissue types could describe all known tissue-specific expression patterns of all genes. This matrix could be used to construct tissue modules (i.e., tissues with similar expression patterns) and, subsequently, allow users to see which tissue modules are significant in their input data. In the same way, modules could be constructed for GO annotations, REACTOME pathways, publications, and even authors (by first creating a matrix associating published authors with genes they have written about.) All of these types of modules could be returned to the user upon submission of their input data, possibly revealing important patterns in their data that were previously hidden.

Predicting gene modules for other species

In preliminary work, we used ICA to predict gene modules from yeast, fly, and mouse Affymetrix arrays. Working with several batches of 100 arrays each, we observed that the optimal preprocessing regime appears to differ between these different species. It is not clear whether this is due to fundamental differences in gene expression between these organisms, or due to technical differences in the arrays; each array platform, though all were manufactured by Affymetrix, contained significant design differences from the others. Thus, the first challenge in applying our method in another species is to identify the optimal data preprocessing regime to use for that species. We expect that an approach similar to the one we employed to optimize module prediction in *C. elegans* will be successful in optimizing module prediction in other species.

The second challenge in applying our method to different species, especially to mouse and human, arises from the fact that these organisms contain an order of magnitude more microarrays in public repositories than were available for *C. elegans* at the time we performed our analysis. For example, at present (August, 2014), there are over 40,000 microarray samples available for the most common mouse Affymetrix array, and over 50,000 for the most common human Affymetrix array, compared with just over 1700 for *C. elegans*. This presents a computational challenge – using the same algorithms and computers we used in our *C. elegans* analysis for all of the data in mouse or human would take many months of computing time and a prohibitive amount of RAM. Thus, either smaller sets of arrays would need to be selected for module construction or different algorithms would need to be employed for predicting gene modules with mouse or human Affymetrix arrays. A highly parallel version of ICA able to run on a computer cluster could meet this need, but this has not yet been developed.

Yet another challenge in applying our method to other organisms stems from the fact that gene regulation is likely more complex and may work at larger upstream distances from the transcription start site in other organisms. In *C. elegans*, promoter regions in the range of 1 – 2 kilobases are often sufficient to recapitulate the expression pattern of a gene when fused to a reporter, such as GFP(Dupuy et al 2004). This is not the case in more complex organisms, such as *D. melanogaster* (Roland Bainton, personal communication). Thus, larger upstream regions (and downstream regions as well) may be needed for successful capture of functional transcription factor binding sites for most genes in *D. melanogaster* or more complex organisms. This could significantly dampen the signal used for word module detection, as additional input DNA for a gene would likely contain many words that serve no biological function for the gene in question, but that do serve a regulatory function to a different gene. The application of algorithms that predict DNA accessibility to transcription factor binding in order to limit the word search region for each gene may help to mitigate this problem.

Exploring gene module homology

Predicting gene modules in multiple species would allow cross-species comparisons of gene modules and, potentially, of transcriptional influence as well (providing the challenges described above are overcome). Aside from shedding light on how specific biological processes differ between species and how those processes may have evolved, cross-species module comparisons may help to answer two types of questions of particular interest to myself and other members of the Kenyon lab.

The first of these pertains to the process of gene and word module prediction. Using homologous modules to translate information that is abundant for one species but scarce for a second could aid the prediction and refinement of gene and word modules in the second species. For example, the number of well-characterized *D. melanogaster* transcription factors with binding site descriptions in public databases far exceeds those of *C. elegans*. Matching a transcription factor to a word module in *D. melanogaster*, therefore, should present less of a challenge in many cases than it does in *C. elegans*. A putative regulator for a *C. elegans* word module could potentially be found by identifying the gene homologue of a regulator found in *D. melanogaster* for a homologous word module. In addition, homologous modules may serve as guides for adapting our algorithm to more complex organisms. For example, a set of well-conserved gene modules between *C. elegans* and a second species could be used to optimize the search space for promoter words in the second species. Such a search would rely on the assumption that conserved gene modules should be regulated similarly in different species; this assumption appears reasonable, but it would need to be validated prior to conducting such work.

The second type of research question that homologous modules may help answer pertains to the use of model organisms to study human diseases. Perturbations that mimic a human disease state or perturb a biological process that is pertinent to a human disease may alter the activity of both conserved and non-conserved gene and word modules. In these cases, researchers may wish to focus their attention on the genes comprising conserved modules, as these may be more relevant to the human disease being investigated. In addition, perturbations in a model organism could be translated first into perturbations of modules (both word and gene), and then into perturbations of homologous modules in humans. For example, the gene fold changes observed in extremely long-lived variants of *C. elegans* could be translated into activity levels of human modules with *C. elegans* homologues. The vast repository of human microarray data could then be searched for conditions that elicit similar module activities. The experimental conditions that such arrays tested serve as hypothesis for the question, “What perturbations in human cells give a similar result to longevity perturbations of *C. elegans?”*

Toward continuous genetic regulatory networks

There are two major remaining challenges to generating continuous genetic regulatory networks from transcriptional influence prediction matrices. The first of these lies in mapping word modules to gene regulators. In the work presented in Chapter 2, the word modules are hypothesized to correspond mainly to protein transcription factors, but additional type of regulators, such as micoRNAs (see below), could be included in the network. Mapping transcription factors to word modules is not a trivial exercise, especially in *C. elegans*, in which relatively few transcription factors have well-characterized binding sites. I currently see two approaches one could use to achieve this mapping.

The first of these would rely on transcription factors in other species with known binding sites. Each word module would be mapped to the closest binding site match in the second species, and then (as described above) *C. elegans* orthologs of this transcription factor would serve as hypothetical matches to the word module. Expression levels of hypothetical matches could then be compared to the activity levels of the word module in a set of expression data; strong correlations (either positive or negative) would serve to bolster the hypothesis that the regulator acts upon the word module.

The second approach to mapping word modules to regulators would make use of high-throughput protein-DNA binding assays, such as the ChIP-seq data provided by the modENCODE consortium(Gerstein et al 2010). Analysis of non-promiscuous genomic regions (i.e., regions that appear to be relatively specific to a particular factor) can be analyzed with transcription factor binding site analysis algorithms to derive binding sites for each regulator. These can then be compared to word modules to find potential matches. Combined with comparisons between genes predicted to be influenced by the word module and genes shown to have proximal binding by the transcription factor, and regulator expression / word module activity correlation, a “short list” of potential regulators can be generated for each word module.

The second remaining challenge to generating continuous genetic regulatory networks lies in partitioning the influence of each transcription factor on each gene into direct and indirect components, i.e., how much of a transcription factor’s influence on a gene arises from proximal binding to that gene and how much is due to the action of downstream intermediaries? At least two types of data could be used to help make this determination. The first of these is binding site analysis. Genes lacking promoter sequence matches to a word module are candidates for indirect regulation by the corresponding transcription factor, especially if the set of such genes for a word module have one or more promoter sequence motifs in common that putative direct targets lack. The second type of data comes from protein-DNA binding assays. Given DNA binding data for a transcription factor and the predicted influence of that factor on each gene, a classifier algorithm could be trained to discriminate between direct and indirect gene targets. One possible way to train such a classifier would be with a set of transcription factor perturbation experiments using the assumption that, in general, direct targets should show greater expression level changes than indirect targets; the classifier would be iteratively modified until these (predicted) sets showed the greatest difference in the perturbation experiments.

Generation of word modules for 3’-UTRs would add another dimension to the continuous genetic regulatory network. Generation of word modules for 3’-UTRs is relatively straight forward, though one extra step is required (specifically, 3-UTR length normalization of word frequencies, see Chapter 2). Since gene expression regulation in 3’-UTRs is thought to occur mainly through the action of microRNAs, many 3’-UTR word modules could potentially be identified via microRNA database searches. We have not yet attempted this, however, so there may be unforeseen difficulties in this approach.

Once a rudimentary continuous genetic regulatory network is constructed, additional data could be used to test and refine it. Ultimately, an algorithm to produce the most likely continuous genetic regulatory network, given a large set of training data, is envisioned. Such a network could be of great use to researchers, as it could allow them to see how specific perturbations alter the activity of different nodes in the network to produce the observed transcriptional output. This could greatly facilitate the interpretability of genome-wide expression measurements and lead to more fruitful hypothesis generation from these complex data.

References

Dupuy, D. et al. A first version of the Caenorhabditis elegans Promoterome. *Genome Res* **14**, 2169-75 (2004).

Gerstein, M.B. et al. Integrative analysis of the Caenorhabditis elegans genome by the modENCODE project. *Science* **330**, 1775-87.

Publishing Agreement

It is the policy of the University to encourage open access and broad distribution of all theses, dissertations, and manuscripts. The Graduate Division will facilitate the distribution of UCSF theses, dissertations, and manuscripts to the UCSF Library for open access and distribution. UCSF will make such theses, dissertations, and manuscripts accessible to the public and will take reasonable steps to preserve these works in perpetuity.

I hereby grant the non-exclusive, perpetual right to The Regents of the University of California to reproduce, publicly display, distribute, preserve, and publish copies of my thesis, dissertation, or manuscript in any form or media, now existing or later derived, including access online for teaching, research, and public service purposes.

DocuSigned by:
Michael Cary
4DE8613AE293424...
Author Signature

12/18/2020
Date