Decoupling Linear and Nonlinear Associations of Gene Expression

Thesis
By Alan Itakura

In Partial Fulfillment of the Requirements
For the Degree of
Master of Science

King Abdullah University of Science and Technology
Thuwal, Kingdom of Saudi Arabia
May 2013
The dissertation/thesis of Alan Itakura is approved by the examination committee.

Committee Chairperson: Tim Ravasi
Committee Co-Chair: Xin Gao
Committee Member: Taewoo Ryu
The FANTOM consortium has generated a large gene expression dataset of different cell lines and tissue cultures using the single-molecule sequencing technology of HeliscopeCAGE. This provides a unique opportunity to investigate novel associations between gene expression over time and different cell types. Here, we create a MatLab wrapper for a powerful and computationally intensive set of statistics known as Maximal Information Coefficient, and then calculate this statistic for a large, comprehensive dataset containing gene expression of a variety of differentiating tissues. We then distinguish between linear and nonlinear associations, and then create gene association networks. Following this analysis, we are then able to identify clusters of linear gene associations that then associate nonlinearly with other clusters of linearity, providing insight to much more complex connections between gene expression patterns than previously anticipated.
ACKNOWLEDGEMENTS

I would first like to thank Professor Timothy Ravasi for allowing me to work in his lab. I would like to thank Carlo Cannistraci for his guidance and mentorship. I would also like to thank Prof Ravasi, Prof Xin Gao, and Dr. Tae Woo for their input towards my thesis.

I am in great debt to Marcin Rogowski for his support with MatLab and, more importantly, actually enabling the project to progress. Lastly, I would like friends that have kept me sane throughout this process, including Ala’a Ragab, Sarah Ahmahdali, Gao Ge, Fuad Jamour, and Marcin Rogowski. And of course, my greatest gratitude to my mother, who has provided me support in all my endeavors.
# TABLE OF CONTENTS

EXAMINATION COMMITTEE APPROVALS FORM.......................... 2

ABSTRACT......................................................................................... 3

ACKNOWLEDGEMENTS ........................................................................ 4

TABLE OF CONTENTS..................................................................... 5

LIST OF ABBREVIATIONS................................................................. 7

LIST OF ILLUSTRATIONS................................................................. 8

LIST OF TABLES.............................................................................. 9

1. INTRODUCTION.............................................................................. 10
   1.1 HISTORY OF FANTOM............................................................ 10
   1.2 HELICOPECAGE VS RNA-SEQ.......................................... 10
   1.3 MEASURING LINEARITY AND NONLINEARITY................... 14
   1.4 MAXIMAL INFORMATION COEFFICIENT (MIC)................. 15

2. METHODOLOGY ............................................................................ 17
   2.1 DATASET CURATION............................................................. 17
   2.2 OPTIMIZATION OF THE MINE APPLICATION .................... 19
   2.3 IMPLEMENTATION OF THE MINE STATISTIC.................... 22
   2.4 OUTPUT CHARACTERIZATION............................................ 22
   2.5 STATISTICAL SIGNIFICANCE.............................................. 22
   2.6 THRESHOLDING AND EDGE ASSIGNMENT...................... 23
   2.7 GRAPH VISUALIZATION...................................................... 25

3. RESULTS....................................................................................... 28
   3.1 MERGED DATASET.............................................................. 28
   3.2 CHARACTERIZATION OF MATLAB WRAPPERS................... 28
   3.3 DETECTING CLUSTERS OF LINEARITY.............................. 32
   3.4 PERFECT LINEARITY............................................................. 46
3.5 NONLINEAR ASSOCIATIONS........................................ 47

4. DISCUSSION................................................................. 57
  4.1 IMPLEMENTATION AND OPTIMIZATION ...................... 57
  4.2 VISUAL ANALYSIS OF NETWORKS WITH LINEAR AND NONLINEAR ASSOCIATION........................................ 58
  4.3 NONLINEAR MODULES: UBL3 AND INCIDENT NODES .... 61
  4.4 GENERAL CONCLUSIONS OF THE NONLINEAR NETWORK .... 62
  4.5 FURTHER WORK.......................................................... 63
REFERENCES........................................................................... 65
# LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIC</td>
<td>Maximum Information Coefficient</td>
</tr>
<tr>
<td>RNA</td>
<td>Ribonucleic Acid</td>
</tr>
<tr>
<td>FANTOM</td>
<td>Functional Annotation of the Mammalian Genome Consortium</td>
</tr>
<tr>
<td>cDNA</td>
<td>complementary deoxyribonucleic acid</td>
</tr>
<tr>
<td>CAGE</td>
<td>Cap analysis gene expression</td>
</tr>
<tr>
<td>MINE</td>
<td>Maximal information-based nonparametric exploration</td>
</tr>
<tr>
<td>TPM</td>
<td>tags per million</td>
</tr>
<tr>
<td>DAVID</td>
<td>Database for Annotation, Visualization and Integrated Discover</td>
</tr>
<tr>
<td>RMSE</td>
<td>Root mean square error</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1 – HeliScope CAGE workflow ................................................................. 12
Figure 2 – Comparison of association characterizations ....................................... 16
Figure 3 – Computation times of MINE wrappers ............................................... 29
Figure 4 – Extrapolated computation times of MINE wrappers ............................ 30
Figure 5 – Degree distribution ............................................................................. 33
Figure 6 – Network of associations greater than 0.85 with modularity of 5 .......... 35
Figure 7 – Network of associations greater than 0.85 with modularity of 0.5 ....... 37
Figure 8 – Select gene expression plots for linear associations ............................ 40
Figure 9 – Select gene expression plots for linear associations ............................ 43
Figure 10 – Select gene expression plots for nonlinear associations .................... 45
Figure 11 – An example of perfect linearity .......................................................... 47
Figure 12 – Giant component of network composed of nonlinear associations ...... 48
Figure 13 – Select gene expression plots for nonlinear associations involving kinases 50
Figure 14 – Second largest component of nonlinear network ............................. 52
Figure 15 – Gene expression plots of Ubl3 and incident nodes ............................. 54
Figure 16 – Select gene expression plots of second largest component ................. 55
LIST OF TABLES

Table 1 – The classes of differentiation series ................................................................. 18
Table 2 – Combined dataset metrics ............................................................................. 28
Table 3 – Computation times of MINE wrappers with increasing number of genes .......... 29
Table 4 – Computation times with increasing time points ................................................. 31
Table 5 – Associations at different threshold values ....................................................... 32
Table 6 – List of genes with select nonlinear associations .............................................. 39
Table 7 – The individual differentiation series and corresponding color ....................... 39
Table 8 – Gene ontology enrichment of red module ....................................................... 49
1. Introduction

1.1 History of FANTOM

The Functional Annotation of the Mammalian Genome Consortium (FANTOM) began in 2000 as a group effort to annotate the cDNA clones produced by the Mouse Encyclopedia Project (Kawai et al. 2001). Since then, there have been 4 follow-up efforts, each focusing on transcriptome analysis. The FANTOM project consistently employs the latest experimental and computational techniques to remain at the forefront of transcriptome analysis.

The most recent FANTOM consortium, FANTOM5, is a multifaceted approach that aims to define the transcriptional regulation networks that determine every human cell type. Notably, each FANTOM ‘round’ has been able to utilize the latest developments in high throughput sequencing technology to allow for a greater breadth and accuracy of insight into the human transcriptome. In order to define the transcriptional networks of type of cell, FANTOM5 looks at snapshots of 200 cancer cell lines, 200 primary cell types, and 30 time courses of cell differentiation (http://fantom.gsc.riken.jp/).

1.2 HeliscopeCAGE vs. RNA-seq

FANTOM5 uses a new combination of techniques termed as HeliScopeCage, an obvious combination of capped-expression gene expression (CAGE) analysis and Heliscope true single molecule sequencer (Kanamori-Katayama et al. 2011).

CAGE analysis identifies transcription start-sites by sequencing short tags originating from 5’ end of mRNA. First, mRNAs are reverse-transcribed using either a poly-A tail based primer, or random priming. Then, RNAs are isolated by biotinylating their 5’
methylguanylate cap, and subsequently pulling them down with streptavidin, and sequencing them. When coupled with high-throughput sequencing for in-depth coverage, this technique is known as ‘deepCAGE’. By counting the number of CAGE tags, one can estimate the cellular concentration of mRNA for a particular gene (Hoon & Hayashizaki 2008). And because this technology captures the 5'-end of the transcript enable the accurate mapping of the proximal promoter used to drive the transcript expression.

Originally, all CAGE analysis was dependent on a step of amplification as a part of sequencing. This amplification step introduces potential biases caused by extreme base composition, such as high guanine cytosine content (Aird et al. 2011). The most obvious way of overcoming this issue is by bypassing the step completely, something that 3rd generation sequencing, known as single molecule sequencing, has been able to do. FANTOM5 consortium employed the HeliScope Genetic Analysis System for single-molecule sequencing, having been hailed as the first true single-molecule sequencer to market. The approach is based on detecting individual incorporation of labeled nucleotides on a synthesizing strand (Thompson and Steinmann, 2010).

In HeliScopeCAGE, CAGE procedures were simplified to be compatible with the HeliScope sequencer. Firstly, whole RNA is extracted from the cell. The RNA is then reverse-transcribed to cDNA using random primers, creating cDNA-RNA hybrids. The hybrids that include the 5’ end of the transcript are isolated by first biotinylating the 5’ cap and then pulled them down using streptavidin. The cDNA strand is then eluted, and a poly-adenosine tail is added to the tags. This allows them to be then pulled down by immobilized poly-thymine oligonucleotides at randomly distributed points within a flow cell (Figure 1). Fluorescently-labeled and blocked nucleotides are then washed over, one
nucleotide type at time, imaged, and then the fluorescent label removed, leaving the incorporated nucleotide. Consequently, for every imaging round, the particular location where fluorescence is detected indicates that the particular nucleotide washed over was incorporated into the immobilized synthesizing strand. By repeating the synthesizing, washing, imaging, and cleaving steps multiple times (typically more than 25 times, every immobilized cDNA will be gradually synthesized and the sequence recorded (Kanamori-Katayama et al. 2011).

**Figure 1** – HeliScope CAGE. A) Reverse transcription using random primers. B) The 5’ mRNA cap is biotinylated. C) Removal of single stranded RNA using RNAses. D) Biotinylated RNA is pulled down using streptavidin beads. E) cDNA is eluted. F) Poly-adenosine tails are added to the cDNA. G) Fixed oligo-T (dT50) pulled down the cDNA by the poly-A tails. H) Begin sequencing (Adapted from Kanamori-Katayama et al. 2011).
This method of gene expression is very robust and sensitive, able to detect across 5 orders of magnitude (1 tag count per million (tpm) to 10,000 tpm) in a reproducible manner (Kanamori-Katayama et al. 2011). Furthermore, HeliscopeCAGE is outperforms other CAGE analysis techniques paired with 2\textsuperscript{nd} generation sequencing in terms of depths of sequencing, the amount of input RNA required, sensitivity, and amplification bias.

RNAseq, or RNA sequencing, is conceptually simple: fractionate the whole transcriptome, add adaptors than permit sequencing, and sequence. It offers a high resolution, quantitative approach to analyzing the transcriptome. Compared to microarrays, RNAseq is much more quantitative, and is not dependent on prior genome knowledge to characterize the transcriptome (Wang et al. 2009).

Of all the methods to measure gene expression, HeliScopeCage and RNA-seq are the most powerful. Both CAGE and RNA-seq have their respective advantages and disadvantages. Because CAGE only measures one area of transcription, it is better suited to measure novel transcription; on the other hand, the signal of RNA-seq can occur anywhere along the length of the transcript which complicates the analysis, but allows for identification of alternative splice variants (Wang et al. 2009). Because HeliscopeCAGE measures the tags of the same part of a transcript, it is a more accurate quantitative measurement than RNA-seq, which samples transcripts multiple times along their length. As a consequence, CAGE analysis permits comparison of gene expression between loci without normalization (Kanamori-Katayama et al. 2011).

For the FANTOM5 project, both RNA-seq and CAGE are used complimentary. For the time course data utilized in this project, HeliscopeCAGE is the choice methodology.
1.3 Measuring linearity and nonlinearity

Classically, most investigations seek only linear associations, as it is the easiest to define and to explain conceptually (Yi 2011). Nonetheless, biological systems are always proving to be more intricate and complicated than initially believed. Nonlinearity associations in biological systems are becoming increasingly recognized as a common phenomenon (Fujita et al. 2008). There are several methods to detect associations and measure dependence of co-expressed genes. Reshef and colleagues (2011) provide a thorough comparison of some predominant methods, including spatially adaptive smoothing splines and loess nonparametric regression. Spatially adaptive smoothing splines are approximations of a function using a series of polynomial functions fitted to particular subsections of data (Fan & Yao 2005). Loess nonparametric regression score variables based on how well they fit an estimated curve (Cleveland & Devlin 1988). Like spatially adaptive smoothing splines, the estimated curve is dependent on localities of values. Typically, the local polynomials are 1\textsuperscript{st} and 2\textsuperscript{nd} degree, while in splines, the polynomial degree can be much higher and there is more emphasis on how the boundaries of each local polynomial are determined (Hastie et al. 2009).

Notably, these approaches do not definitively identify many other potentially interesting relationships that are not functional; in other words, they lack generality (Figure 2). Other tests, like Spearman’s rank correlation coefficient, are limited by the types of relationship \textit{(i.e. monotonicity)}. As a consequence, do not treat all types of association equally and are said to lack the quality of equitability.
1.4 Maximal information coefficient (MIC)

MIC is the flagship statistic of set of statistics deemed Maximal Information-based Nonparametric Exploration (MINE) (Reshef et al. 2011). The statistic we are most interested in is MIC, as it is able to identify associations in a manner that is both general and equitable. MIC calculates the mutual information, which is essentially a measurement of how much one variable explains the other, for particular subsections, or ‘bins’, of a scatterplot of the two variables. The number and size of these bins are optimized to maximize the mutual information score. Altogether, this allows the MIC statistic to capture a wide range of associations, regardless of the type of relationship, and treat them all equally (Reshef et al. 2011). In other terms, this means that MIC will detect non-functional relationships, such as ellipses and bifurcations, with the same accuracy as simple linear functions (Figure 2).
Figure 2 – Comparison of association characterizations A) The MIC coefficient scores for the different patterns at different levels of noise (the more blurry a line, the more noise). B) The scores given by smoothing spline. C) The scores given by nonparametric regression (Loess) (Adapted from Reshef et al. 2011).
2. Methodology

2.1 Dataset Curation

The data used was obtained from the FANTOM5 consortium, and consist of the gene expression 26 different tissue types across 3 types of differentiation series classes: human tissue differentiation series *in vitro*, mouse tissue differentiation series *in vitro*, and mouse *in vivo* tissue differentiation series (Table 1). From here, ‘class of differentiation series’ will refer to the above, while ‘differentiation series’ refers to an individual subset of data (*i.e.* macrophage response to lipopolysaccharide or mesenchymal stem cell differentiation to adipocyte and osteocytes’).
There are several reasons why we analyzed each class of differentiation series separately. Firstly, the gene symbols and Refseq IDs between the mouse and human are not identical, which complicates comparisons between genes of the different species. Secondly, the conclusions that can be made from in vitro differentiation series are not as potent as those that can be made from the in vivo differentiation series (Yang & Xiong 2012). Lastly,

There are several reasons why we analyzed each class of differentiation series separately. Firstly, the gene symbols and Refseq IDs between the mouse and human are not identical, which complicates comparisons between genes of the different species. Secondly, the conclusions that can be made from in vitro differentiation series are not as potent as those that can be made from the in vivo differentiation series (Yang & Xiong 2012). Lastly,

<table>
<thead>
<tr>
<th>Dataset #</th>
<th>Human tissue differentiation series in vitro</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Macrophage response to lipopolysaccharide</td>
</tr>
<tr>
<td>2</td>
<td>Osteosarcoma model of calcification/bone deposition</td>
</tr>
<tr>
<td>3</td>
<td>Retinal pigment transition to mesenchyme</td>
</tr>
<tr>
<td>4</td>
<td>Preadipocyte differentiation to adipocyte</td>
</tr>
<tr>
<td>5</td>
<td>Primary aortic smooth muscle cells response to interleukin1-beta and fibroblast growth factor</td>
</tr>
<tr>
<td>6</td>
<td>Embryonic stem cell differentiation to CD34+ hematopoietic stem cells</td>
</tr>
<tr>
<td>7</td>
<td>Melanocytic differentiation</td>
</tr>
<tr>
<td>8</td>
<td>Cardiomyocyte differentiation</td>
</tr>
<tr>
<td>9</td>
<td>Erythcytic differentiation</td>
</tr>
<tr>
<td>10</td>
<td>Primary lymphatic endothelial cells response to vascular endothelial growth factor C</td>
</tr>
<tr>
<td>11</td>
<td>Breast cancer line response to epidermal growth factor and heregulin</td>
</tr>
<tr>
<td>12</td>
<td>Human primary myoblast differentiation to myotube</td>
</tr>
<tr>
<td>13</td>
<td>Induced pluripotent stem cell to neuron differentiation timecourse</td>
</tr>
<tr>
<td>14</td>
<td>Primary mesenchymal stem cell differentiation to adipocyte</td>
</tr>
<tr>
<td>15</td>
<td>Stem cell response to Rinderpest infection</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset #</th>
<th>Mouse tissue differentiation series in vitro</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ST2 mesenchymal stem cell differentiation to adipocyte and osteocyte</td>
</tr>
<tr>
<td>2</td>
<td>EBF model of T-cell differentiation timecourse</td>
</tr>
<tr>
<td>3</td>
<td>Embryonic stem cells differentiated to neurons</td>
</tr>
<tr>
<td>4</td>
<td>Models of erythrocytic differentiation</td>
</tr>
<tr>
<td>5</td>
<td>Mouse primary lunge epithelial cell differentiation to ciliated epithelial cells</td>
</tr>
<tr>
<td>6</td>
<td>Bone marrow derived macrophage response to tuberculosis infection</td>
</tr>
<tr>
<td>7</td>
<td>Trophoblast stem cell differentiation to trophoblasts</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset #</th>
<th>Mouse tissue differentiation series in vivo</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cerebellum development</td>
</tr>
<tr>
<td>2</td>
<td>Hematopoiesis stem cell development</td>
</tr>
<tr>
<td>3</td>
<td>Liver regeneration</td>
</tr>
<tr>
<td>4</td>
<td>Visual cortex development</td>
</tr>
</tbody>
</table>

**Table 1** – The individual datasets that make up the three classes of differentiation series.
these classes of differentiation series were already present in the organization of the FANTOM5 consortium.

For each group of time courses, the original text files were compiled into one large continuous matrix in MatLab, with each differentiation series (i.e. macrophage response to lipopolysaccharide) being assigned a number in order to track them later in our analysis. Consequently, some gene transcripts may be expressed in one differentiation series, and not in another. Because we are seeking associations between genes, we filtered for loci/mRNA RefSeq IDs present within each differentiation series of each class, and only considered these genes for the remainder of our analysis.

2.2 Optimization of the MINE Application

The MINE application, used to calculate MIC and its related statistics, can be downloaded from exploredata.net. The original application is written in Java. In addition to the executable jarfile they provide, they also provide ‘wrappers’, which allow the original Java application to be used in the context of different programs, such as R and Python programming languages.

Because of the popularity of MatLab among bioinformaticians, we searched the literature to determine if other users of the statistic have employed the MINE application in MatLab. Our search identified a MatLab wrapper developed and used by a lab in Norway. The PhD student and his advisor graciously allowed us to use it (Eikeland, 2012).
The Matlab wrapper was adapted to work in Windows Operating System and modified to optimize the speed of the computation. In all cases, the MINE statistic was run at default parameters.

Upon applying it to our dataset, we quickly realized that the size of our dataset was a major obstacle. Firstly, the Java portion of the MatLab wrapper would fail due to memory requirements; secondly, the calculation time using the Java wrapper was prohibitively slow. These problems were both caused by the Java code, as when the MatLab feeds the input data into Java, the Java file makes all gene comparisons simultaneously which overwhelms the memory storage of a computer.

In an attempt to resolve this, three different approaches were taken. The first (version 1) would compare one variable against all others, produce an output, and then this output was saved to a matrix. This would be repeated (using a ‘for’ loop) for every single variable pair. This approach avoids running the Java script for all variable comparisons at once, and solved the memory problem. However, this approach then causes multiple redundant comparisons to be made (i.e. variable 1 is compared to all other variables, including variable 4. When variable 4 is compared to all other variables, it is again compared to variable 1 unnecessarily). This is a consequence of the intrinsic properties of the Java program: there are only options to run the code by comparing one variable against another, one variable against all other variables, or comparing all variables against all other variables.

The second approach (version 2) is very similar to version 1 in that it compares one variable at a time to every other, but reduces the number of redundant calculations by
deleting every variable that has already been compared thereby circumventing the built-in limitations of the Java code.

The final approach (version C) is to use a C-based code rather than a Java-based code. By doing this, we avoid using the memory-limited Java script to address the issue of memory usage. This MatLab wrapper the C-based implementation of the MINE statistics was created specifically to reduce the amount of memory used (Albanese et al. 2012). However, it is important to note that the calculations might not be identical, as the source code is not provided, so Albanese and colleagues have implemented the algorithms provided. In order to quantify the potential difference between the C-based implementation of MINE and the Java-based implementation of MINE, we calculate the root-mean square error (RMSE) as:

$$RMSE = \sqrt{\frac{\sum (new \ result - old \ result)^2}{range \ of \ results}}$$

where the range of results is equal to 1, which derives from the possible range of MIC values.

For every version of the MatLab the final output was compared to sample datasets obtained used in the original MINE paper in order to sure that alterations in the method computation do not affect the computation itself.

Memory efficiency is not the only concern with these calculations; the large number of variables makes time a constraint as well. The computation time of each MatLab wrapper was determined for increasing sizes of matrices in order to determine which version would be the fastest. We characterized both dimensions of the input matrices (i.e. the
number of variables, or genes, and the number of time points) to gain a better understanding how they contribute to the running time. Truncated matrices of our largest dataset and randomly generated matrices were used at scales less than the final matrices. Linear and quadratic curves were fitted to the data in order to give a rough extrapolation of how long the computation on the complete dataset would take.

2.3 Implementation of the MINE statistic

Despite our best attempts to optimize the code, the scale of our datasets meant that the number of computations exceeds what one machine is able to process in a reasonable amount of time. As a consequence, we parallelized the C-based code to run on over 80 machines containing 500 processing cores total. For the in vitro mouse dataset, it took about 12 hours to run, while the largest dataset took approximately 24 hours total.

2.4 Output Characterization

Every gene was compared against itself; consequently, the output matrices would be $n$ by $n$ in size, $n$ being the number of genes. The matrices would be mirrored along the diagonal of the matrix. We filled the diagonals with zeros, as we want to avoid perfect correlation/association of self-comparison. Consequently, we are left with $n(n-1)/2$ values, with $n$ being the number of genes.

2.5 Statistical Significance

The two most accessible methods to address the issue of multiple testing are the Bonferoni and Benjamini Hochberg corrections (Hochberg & Benjamini, 1990). Unfortunately, due to the number of comparisons we are performing (Table 2), Bonferroni would reduce the $p$-value necessary for significance to a value far too
conservative (Green & Britten, 1998). Additionally, the $p$-values for the MIC test are not readily available for the whole spectrum of values. On exploredata.net, the home of the MIC statistic, values for different sample sizes are provided, but only up to a certain MIC value. This renders us unable to perform either correction, as the $p$-values are required to do so. Nonetheless, we can deduce from the table of $p$-values that the MIC statistics for the associations that we consider have very low uncorrected $p$-values ($\leq 5 \times 10^{-6}$). Consequently, for our analysis, we perform no correction to the MIC statistic $p$-value. While this statistically incorrect, our intention is to have just narrowed a list of associations that might be interesting, rather than a confirmed list. Furthermore, the analysis carried on later is qualitative, and requires individual inspection of particular sub-networks and associations prior to making conclusions. Additionally, the thresholding we apply to classify linear or nonlinear associations is high enough that we can safely assume the MIC values we consider are statistically significant.

For the Pearson correlation coefficient ($\rho$), we perform the more conservative Bonferroni correction.

### 2.6 Thresholding and Edge Assignment

With hundreds of millions of associations for each class of differentiation series, we look to filter for the most linear and nonlinear associations in order to reduce the number of genes and associations in our analysis. Because we are aiming to create an interaction network, genes will be the nodes, and the associations will be the edges.

To filter and define nonlinear associations, we calculated the difference between the MIC and Pearson coefficient, reasoning that if the MIC detects all associations, and the
Pearson detects linear associations, then the difference between the two will be the nonlinear associations. We term this value ‘delta’ ($\Delta$).

$$\Delta = MIC - \rho$$

For nonlinear associations, we filter for associations between two genes that have a specified cut-off value $\Delta_{\text{nonlinear}}^*$ large enough to say that we are looking at the most linear associations; and we also set a cut-off $\rho_{\text{nonlinear}}^*$ greater than or equal to 0, which avoids the scenario of a negative $\rho$ value being added to the MIC value.

$$\left\{ \begin{array}{l} 0.85 \leq \Delta_{\text{nonlinear}}^* \leq 1 \\ \rho_{\text{nonlinear}}^* \geq 0 \end{array} \right.$$  

The genes that fulfill these criteria genes will act as nodes with nonlinear associations and the edge weight will be:

$$w_{\text{nonlinear}} = 1 - \Delta_{\text{nonlinear}}^*$$

This implies that nonlinear association will be quantitatively characterized in the reconstructed network of association with very low weights.

For linear associations, an association would be considered if its $\rho_{\text{linear}}^*$ was greater than or equal to the specified cut-off value.

We kept the threshold values the same for linear and nonlinear associations. Additionally, the $\Delta_{\text{linear}}^*$ for the same association ideally would be relatively small, indicating that $\rho$ is in agreement with MIC.

$$\left\{ \begin{array}{l} 0 \leq \Delta_{\text{linear}}^* \leq 0.1 \\ \rho_{\text{linear}}^* \geq 0.85 \end{array} \right.$$
The two genes would be the nodes, and the edge would be assigned an edge linear weight:

\[ w_{linear} = \rho_{linear} \]

This implies that linear association will be quantitatively characterized in the reconstructed network of association with very high weights.

This edge weighting scheme allows for linearity and nonlinearity to be comparable in values: linear connections have a weighting nearer 1, while a nonlinear edge will have a weight nearer 0. This is important for network analysis later.

### 2.7 Graph Visualization

To visualize networks, we opted for the program Gephi (Bastian et al. 2009). Firstly, it has a built-in method to calculate modularity and other insightful statistics. Secondly, it has tools to allow filtering of nodes or edges based on different properties, such as edge weight or degree. Thirdly, it can handle large networks and perform the aforementioned functions in a timeframe that is not prohibitive.

The associations that met the criteria described in the previous sections were then concatenated into a matrix, along with the gene name, the Ref Seq ID, and the edge weight. This information was then converted into a .gml file (graph modeling file). A .gml file defines the nodes and their properties (i.e. name, node ID), and the edges and their properties (i.e. between which nodes, weight). This file was then imported into Gephi, which then be used to visualize the network.

The initial step is to hide all isolated nodes (degree less than two). This serves to remove any components consisting of only two nodes (i.e. an association between two genes that
are not associated with any other genes). This new filtered network was then moved in to a new workspace for further analysis. Modularity partitioning was different resolutions in order to detect communities of different sizes. The modularity was calculated using a repetitive algorithm developed by Blondel and colleagues (2008). In order to visualize modularity, each module would be assigned a random color.

Lastly, Force Atlas 2, a force-directed spacing algorithm, is applied. This essentially spaces out nodes proportional to the edge weights that exist between them. For instance, the space between two nodes that share an edge with weight of 1 would be smaller than the space between two nodes with an edge weight of 0.01. This attraction-repulsion based layout allows for a readable shape to be applied to the network (Jacome et al. 2012).

In order ease the analysis of our network, we focus on one to two modules at a time, choosing to move these to a new workspace and apply the modularity analysis and force atlas layout again. If an interesting network motif was recognized, such as a small cluster of nodes linearly associated with each other with a high degree of nonlinear associations to other nodes, then those nodes and all neighboring nodes would be extracted and moved to a new workspace. This analysis is operator-based, and is also dependent on the particular graph being analyzed. For the layout of this focused graph, we used the Fruchterman-Reingold layout algorithm to result in a network that distributes the nodes evenly, thus attaining an aesthetic layout (Fruchterman & Reingold 1991).

Selected associations were plotted. In some cases, a log scale is used because of the different range of magnitudes genes can be expressed at (i.e. one gene might be expressed with a range of 1-100 tpm, while another will have a range from 1-10000 tpm).
We used the Database for Annotation, Visualization and Integrated Discover (DAVID) to search for enriched gene ontology terms and pathways. All tests were done using the whole mouse genome as a background. A gene ontology term or pathway was considered enriched if had a significant Benjamini-corrected $p$-value (Huang et al. 2009a; Huang et al. 2009b).
3. Results

3.1 Merged Datasets

The datasets of each of the three different differentiation series were merged and the common genes between them were filtered for. The metrics of the merged datasets can be seen in table 2. Note the number of comparisons for each data set approaches 250 million. This becomes important when we start to perform intensive statistics.

<table>
<thead>
<tr>
<th></th>
<th>Datasets</th>
<th>Common Genes</th>
<th>Combined Time Points</th>
<th>Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human tissue <em>in vitro</em></td>
<td>15</td>
<td>21775</td>
<td>747</td>
<td>237064425</td>
</tr>
<tr>
<td>Mouse tissue <em>in vitro</em></td>
<td>7</td>
<td>18287</td>
<td>532</td>
<td>167198041</td>
</tr>
<tr>
<td>Mouse tissue <em>in vivo</em></td>
<td>4</td>
<td>17194</td>
<td>110</td>
<td>147808221</td>
</tr>
</tbody>
</table>

Table 2 – The basic characteristics of the classes of differentiation series, post-filtering for common genes.

3.2 Characterization of MatLab wrappers

We first measured the computation time of different versions of the Matlab wrapper for an increasing number of variables (Table 3). Notably, the original version, version 0, fails to complete the MINE calculations for the 2000x200 dataset after ~1000 variables compared due to insufficient memory (Figure 3; Table 3); however, when it does not fail, it is much faster than the other Java-based wrappers, version 1 and 2. Version 1 is the slowest, most likely because of the redundant calculations it computes, while version 2 looks to be faster than version 1, but more comparable to version 1 than to version 0. The C-based wrapper, version C, appears to have very similar calculation times while avoiding the memory limitations encountered by version 0 (Figure 3; Table 3).
Table 3 – The computation time of each version of the MINE statistic implementation and MatLab wrapper for varying sizes of input matrices. Truncated versions of the *in vitro* tissue differentiation of human tissues were used. The red value indicates that the calculation failed due to memory requirements.

<table>
<thead>
<tr>
<th>Version</th>
<th>25x25</th>
<th>100x100</th>
<th>1000x110</th>
<th>2000x200</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version 0</td>
<td>0.015167</td>
<td>0.493</td>
<td>62.9935</td>
<td>763.55</td>
</tr>
<tr>
<td>Version 1</td>
<td>0.188833</td>
<td>2.038667</td>
<td>138.7108</td>
<td>2154.787</td>
</tr>
<tr>
<td>Version 2</td>
<td>0.1775</td>
<td>1.661</td>
<td>107.0073</td>
<td>1351.616</td>
</tr>
<tr>
<td>Version C</td>
<td>0.0075</td>
<td>0.47708</td>
<td>60.1760</td>
<td>769.073</td>
</tr>
</tbody>
</table>

Importantly, version C does not include the Pearson coefficient, while the MatLab wrappers for the Java code do. Nonetheless, this is a relatively quick calculation when compared to the MINE statistics, and would not add more than 3 minutes for each, even at datasets of our size.
These preliminary computations on truncated data are already quite time consuming, suggesting that processing the computed datasets, with up to 20000 variables (genes) and up to 800 time points will be extremely time consuming.

In order to provide some estimation of how long it will take to run these, we’ve plotted these computation times,

![Graph A](image1)

**Figure 4** – Extrapolation of times of computation (in hours) of version 1 (solid), version 2 (dashed), and version C (dotted) plotted with respect of number of genes using linear lines and quadratic curves of best fit. **A)** Quadratic curves of best fit (solid: $y_1 = 1.596e-05x^2 + -0.01438x + 0.8042; R=.999$, dash: $y_2 = 9.665e-06x^2 + -0.008305x + 0.4665; R=.999$ dotted: $y_c = 5.51e-06x^2 + -0.004745x + 0.2621; R=.999$) **B)** Linear lines of best fit (solid: $y_1=0.01722x + -3.89, R=0.905$; dashed: $y_2 =0.01083x + -2.375, R=0.911$; dotted: $y_3 = 0.006164*x + -1.358, R=0.911$) up to 800 time points will be extremely time consuming. In order to provide some estimation of how long it will take to run these, we’ve plotted these computation times,
and fit models to them. The two types of models we have considered, linear and quadratic, appear to be the most appropriate (as well as the most conservative) (Figure 4). Unfortunately, because of the time it takes to run these demos, the sample sizes that we are able to conduct on are several orders of magnitude smaller than our actual data sets making our approximations at these larger magnitudes very rough – nonetheless, even in the most optimistic scenario with a linear estimation, the time to complete a dataset of 20,000 using version C would be 122 hours (about 5 days). If the scaling is more quadratic behavior, the time would be 2100 hours (87 days) (Figure 4).

Even more troubling, these estimations do not appropriately account for a larger number of time points, which also appear to cause significant increase in computation time. All together, these results suggest that time of computation looks to be a severely limiting factor for our complete datasets.

<table>
<thead>
<tr>
<th></th>
<th>100x10</th>
<th>100x25</th>
<th>100x50</th>
<th>100x100</th>
<th>100x250</th>
<th>100x500</th>
<th>100x1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version 0</td>
<td>0.0350</td>
<td>0.04731</td>
<td>0.15970</td>
<td>0.65977</td>
<td>5.05695</td>
<td>21.9094</td>
<td>96.36277</td>
</tr>
<tr>
<td>Version 1</td>
<td>0.4680</td>
<td>0.60557</td>
<td>0.86858</td>
<td>2.03604</td>
<td>11.0614</td>
<td>45.199</td>
<td>192.712</td>
</tr>
<tr>
<td>Version 2</td>
<td>0.4105</td>
<td>0.53030</td>
<td>0.74097</td>
<td>1.39987</td>
<td>6.35585</td>
<td>23.8212</td>
<td>99.86045</td>
</tr>
<tr>
<td>Version C</td>
<td>0.0038</td>
<td>0.01884</td>
<td>0.14905</td>
<td>0.61744</td>
<td>3.81364</td>
<td>14.9572</td>
<td>58.00592</td>
</tr>
</tbody>
</table>

Table 4 – The computation times for different versions of the MatLab wrapper (rows) for data with an increasing number of time points (columns)

It is important to note that version C was created by Albanese and colleagues (2012) by replicating the algorithms provided by the original MINE publication (Reshef et al. 2011), and is not a true copy of the original Java code. As a consequence, we checked the fidelity of the translation by calculating the root mean square error for the MIC of version C and version 1. For the whole in vitro mouse differentiation series, the normalized
RMSE was a mere 0.3298%, indicating that the MIC values are identical for all intents and purposes. In conclusion, given these estimations, we decided to perform the computation using version C and running the code using 500 processing cores available in the Visualization facility, solving the problem in an average time of two days for each dataset.

### 3.3 Detecting clusters of linearity

Once we completed the MIC, $\rho$ and $\Delta$ computations, we looked at the strongest linear and nonlinear associations. From here, we focused our analysis on the on the *in vitro* mouse tissue differentiation dataset. By considering multiple thresholds for strong linearity and nonlinearity, we could choose a reasonable number of total edges and nodes for our network. Cut-offs greater than or equal to 0.9 results in very few associations, and consequently sparse graphs that make further visual analysis difficult. Conversely, thresholds of 0.8 results in too many associations – we settle opt for cutoffs of 0.85 as a comfortable in-between (Table 5). This somewhat arbitrary limitation was imposed to ease visual analysis.

<table>
<thead>
<tr>
<th>Type of association</th>
<th>Threshold</th>
<th># of associations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>$\rho \geq 0.8$</td>
<td>14081</td>
</tr>
<tr>
<td>Nonlinear</td>
<td>$\Delta \geq 0.8$</td>
<td>6740</td>
</tr>
<tr>
<td>Linear</td>
<td>$\rho \geq 0.85$</td>
<td>3400</td>
</tr>
<tr>
<td>Nonlinear</td>
<td>$\Delta \geq 0.85$</td>
<td>2175</td>
</tr>
<tr>
<td>Linear</td>
<td>$\rho \geq 0.9$</td>
<td>283</td>
</tr>
<tr>
<td>Nonlinear</td>
<td>$\Delta \geq 0.9$</td>
<td>534</td>
</tr>
<tr>
<td>Linear</td>
<td>$\rho \geq 0.95$</td>
<td>14</td>
</tr>
<tr>
<td>Nonlinear</td>
<td>$\Delta \geq 0.95$</td>
<td>121</td>
</tr>
<tr>
<td>Linear</td>
<td>$\rho \geq 0.99$</td>
<td>0</td>
</tr>
<tr>
<td>Nonlinear</td>
<td>$\Delta \geq 0.99$</td>
<td>6</td>
</tr>
</tbody>
</table>

**Table 5** – The number of associations (edges) generated by different $\Delta$ and $\rho$ thresholds on the *in vitro* mouse tissue differentiation combined dataset.
Using cutoffs of $\Delta \geq 0.85$ and $\rho \geq 0.85$ on the \textit{in vitro} mouse tissue differentiation series, we generate an undirected network with 3400 linear associations and 2175 nonlinear associations between 1048 nodes (Table 5). The linear associations are 0.002\% of all associations, while the nonlinear make up 0.001\% of all associations. The whole network has an average degree of 10.639, and a degree distribution suggestive of a scale-free distribution characteristic of real world networks (Figure 5; Barabási & Oltvai 2004).

88\% of the graph is part of one giant component, the rest are predominantly small components of 2-3 nodes. The giant component has an average degree distribution of 12.148 (Figure 5), as the whole network contains many small components containing only 2-3 nodes. Because most of the small components consist of either nonlinear or linear associations, we opt to focus our analysis on the giant component.

![Degree Distribution](image)

**Figure 5** – Degree distribution of linear and nonlinear network of \textit{in vitro} mouse tissue differentiation series.

We then apply a modularity algorithm (Blondel \textit{et al.} 2008) at a resolution of 5, which should generate relatively large modules. While there does appear to be three defined modules, these modules are still not entirely linear within themselves, and still contain numerous nonlinear associations (Figure 6A).
We manipulate the network further in an attempt to tease out linear modules connected by nonlinear associations by looking at the red and blue modules. Because the red module is smaller in size, we use this as a limiting factor, pulling out all nodes within the blue module that are incident to nodes within the red module (Figure 6B).
Figure 6 – Nonlinear and linear associations greater than 0.85 with a modularity resolution of 5 applied. Thicker lines are linear associations and thinner lines are nonlinear associations. A) The giant component composed of 5 modules, spatially distributed using the force atlas method described in the methodology. B) A focused view of the red module of the giant component with only the incident nodes of the blue component (now light blue) included. The layout is done using Fruchterman-Reingold, with some manual curation. C) The same network as in B, but with modularity with resolution of 1 applied.
Despite this narrowing approach, there still does not appear to be a clear divide between our modules. Interedges, which are edges that occur between modules, are both linear and nonlinear; similarly, intraedges, which are edges that occur within a module, are both linear and nonlinear (Figure 6B). This suggests that either or measure of modularity or the resolution we are considering is less than ideal.

As we can see, the clusters are still not clear, and the grey edges (which indicate interedges) are numerous between the modules without a clear pattern. It is likely that the resolution of 5 for modularity is too coarse. To determine if a finer resolution might be helpful at a smaller scale, we apply modularity with resolution 3 to this new network, which detects 5 communities (Figure 6C). We can see that new modules emerge, some of which are ‘shared’ by the original two modules (namely the purple module). This again suggests that our initial application of modularity at resolution 5 was again far too coarse.
Figure 7 – Nonlinear and linear associations with Δ and ρ greater than 0.85 with modularity applied at a resolution of 0.5. Thicker lines are linear associations and thinner lines are nonlinear associations. A) The giant component with modularity applied at a resolution of 0.5. Every color represents a different module. The layout is done using force atlas. B) Highlighted modules, the red having enrichment for DNA repair pathways and the blue having nonlinear interactions with the red module. C) The red and blue module from B in detail, with all other modules removed and modularity applied with a resolution of 1. The blue module from B is the light blue module in C. The Fruchterman-Reingold layout is used, with some manual curation.
We also apply the modularity algorithm as described in the methodology to the giant component at a resolution of 0.5, generating 20 modules (Figure 7A). For each module, we looked to see if there was any gene ontology term or pathway enrichment.

One module was significantly enriched for the Biological Process Gene Ontology term for DNA damage repair, the cadherin signaling pathway, and the Biocarta pathway of the role of BRCA1, BRCA2, and ATR in cancer susceptibility (Figure 7B, 7C). Upon identifying this enriched module, we sought out other modules that showed a high degree of nonlinear relationship with the enriched module. This identified one other module, which we isolate along with the original enriched module (Figure 7C).

The resulting subcomponent contains 79 nodes, 488 edges and an average degree of 12.35 per node. The red module, which is enriched for the aforementioned terms, is highly interconnected, while the blue module does not exhibit such extensive interconnectedness (Figure 7C). In addition, there is a defined third module present with only nonlinear associations to the blue module. It is originally considered identified to be part of the blue module, but upon reapplication of modularity, it is recognized as a separate module.

Most of the nodes of the red module have nonlinear associations with only a few nodes of the blue module (Figure 7C). In other words, only a select few blue nodes (Table 6) have multiple nonlinear edges with nodes from the red module, while many of the red module nodes have only one or two nonlinear associations with these blue nodes.
The association plots of the highly interconnected red module visually demonstrate that these genes correlate linearly with each other. Within each association plot, the same datasets retain the same relative levels of gene expression rather consistently. For instance, the gene expression of Fat1, Fars2, Fancf, Fbl, Rsu1, and Fance are all expressed in the same ratios, regardless at what scale and in what differentiation series they are expressed (Figure 8).

<table>
<thead>
<tr>
<th>Dataset #</th>
<th>Mouse tissue differentiation series in vitro</th>
<th>Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ST2 mesenchymal stem cell differentiation to adipocyte and osteocyte</td>
<td>black</td>
</tr>
<tr>
<td>2</td>
<td>EBF model of T-cell differentiation timecourse</td>
<td>blue</td>
</tr>
<tr>
<td>3</td>
<td>Embryonic stem cells differentiated to neurons</td>
<td>red</td>
</tr>
<tr>
<td>4</td>
<td>2E models of Erythrocytic differentiation</td>
<td>green</td>
</tr>
<tr>
<td>5</td>
<td>Mouse primary lunge epithelial cell differentiation to ciliated epithelial cells</td>
<td>yellow</td>
</tr>
<tr>
<td>6</td>
<td>Bone Marrow Derived Macrophage, TB infection</td>
<td>magenta</td>
</tr>
<tr>
<td>7</td>
<td>Trophoblast stem cell differentiation to trophoblasts</td>
<td>cyan</td>
</tr>
</tbody>
</table>

Table 7 – The individual differentiation series of the in vitro mouse tissue differentiation series, each assigned a different color.

The levels of expression in differentiation to T-cells (blue) and to neurons (red) are regularly higher than gene expression in other differentiation series (Figure 8). Some
associations appear to be more linear than others; this juxtaposition is even more evident on the log scale.

**Figure 8** – Select gene expression plots for linear associations within red module of Figure 6C, with regular scale on the left (A, C, E) and log scale on the right (B, D, F). Gene expression is in tag count per million (tpm). Different tissue differentiation series are represented by different colors (see table 7).
The linear edges occurring in the blue module do not have such uniform trends as those in the red (Figure 9). Most appear to be linear upon visual inspection, but this correlation is not as distinct as the linear associations in the red module; nonetheless, when plotted on a log scale, the linearity is apparent. Most points are condensed at low levels of expression, indicating that in most differentiation series, these genes are expressed at low levels.

Even in the linear association between Kremen1 and Pik3c3, the range of the expression for Pik3c3 is 0-25 tpm, and there is still rather clear vertical stratification of some of the differentiation series: mesenchymal stem cell differentiation (black) has expression between 10-25 tpm, while epithelial cell differentiation has Pik3c3 expression between 3-8 tpm, and other tissue sets have even less (Figure 9C). This demonstrates the sensitivity of HelicosCage, and might suggest that small changes in Pik3c3 might have biological significance.

The association plot between Rela and Ccdc37 reveals some potential weaknesses in our measurement of linearity. The association between Rela and Ccdc37 is, according to the ρ value of 0.902, linear; however, the plot there appears to be a rather distinct bifurcation (MIC = 0.920) (Figure 9E).

The nonlinear associations have much more complex association plots, as expected. The genes from the blue module appear to have expression in mesenchymal stem cell differentiation (black) and epithelial cell differentiation (yellow) cluster together at relatively high levels of gene expression; conversely, the genes in the coming from the red module have high expression at low levels of the genes from the blue module (Figure 10). This high expression is owed primarily to the differentiation series to T-cells (blue)
and neurons (red), the same datasets that were responsible for linear correlation at high levels of gene expression in the red module (Figure 10). These nonlinear associations highlight instances where the two genes are both expressed in several differentiation series, but appear to have specific ratios depending in which differentiating tissues they are being expressed. In other words, in particular differentiating tissues, a gene will require a high expression relative to another, while in others this ratio is reversed.
Alternatively, we could analyze these plots while disregarding the different tissue types and make a general statement about the relationship of each pair of genes. In the

**Figure 9** – The gene expression plots for linear associations within blue module of Figure 7, with normal scale on the left and log scale on the right. Gene expression is in tag count per million (tpm). Different tissue differentiation series are represented by different colors (see table 7).
nonlinear associations, it appears that as gene 1 (i.e. Kremen1) increases in expression, gene 2 (i.e. Faskd3) decreases at a rate proportional to its value (Figure 10A).

At a log$_{10}$ scale, different patterns emerge. In the instance of Kremen1 and Faskd3, rather defined clusters correspond consistently to the different tissue differentiation series (Figure 10B). This is particularly relevant in this association plot because the clusters are generally above log$_{10}$ (1.5) (roughly 30 tpm), which is still a relevant level of transcription: any fewer transcripts and then it becomes rather unclear to make some biological relevant hypothesis, particularly when the scale of gene expression expands several orders of magnitude.
Figure 10 – The gene expression plots for nonlinear associations between the red and blue modules of Figure 7C, with normal scale on the left and log scale on the right. Gene expression is in tag count per million (tpm). Different tissue differentiation series are represented by different colors (see table 7).
3.4 Perfect linearity

When applying modularity at a resolution of 0.5, we also observe a second module that is enriched for antigen processing and presentation (Benjamini-corrected $p$-value of $4.08 \times 10^{-08}$ at a false detection rate of $5.77 \times 10^{-07}$). The genes responsible for this enrichment, H2-D1, H2-AA, H2-AB1, H2-DMA, and H13 are all part of histocompatibility complexes, while the other two genes in this module, H0 and H0, are histone genes (Figure 11A).

Oddly enough, this cluster of seven genes all have perfect linear association with each other ($\rho = 1, \text{MIC} = 1$). Despite the fact these genes are located on different chromosomes, they appear to have identical levels of gene expression in all tissue types. In the specific example of H2-Ab1 and H2-DMa, both are expressed in mainly two tissues, but at the same levels and the same time points, suggesting that there is either a machine error or the alignment is incorrect (Figure 11B). Upon further inspection, when considering both mRNA sequences and their upstream sequences, there is one area of significant homology – the upstream area of the H1fx gene to the first intron of H2-D1 (NCBI 2013a; NCBI 2013b). This is still counterintuitive, as one would not expect the HelicosCage to create tags almost 2000nt into the gene from the transcription start site; indeed, gene expression is generated from reads that map within 500bp from the 5’ end of an mRNA (Kanamori-Katayama et al. 2011). Furthermore, the likelihood the HelicosCage tags would always be taken from small range of homology (120bp, not continuous) is small.
3.5 Nonlinear associations

To focus on the interesting nonlinear interactions, we analyzed them independently of the linear interactions. We apply a cut-off of 0.85 to the Δ values generated from the mouse *in vitro* differentiation series to generate a list of strong nonlinear associations. Rather than assigning the edge weight equal to 1-Δ, we assign it simply as Δ, as we are no longer concerned about having a scale of measurement that encompasses both linearity and nonlinearity. This was then visualized using Gephi, resulting in a network of 945 nodes and 2061 edges. 767 nodes are present in the giant component (Figure 12).
This list of nonlinearly associated genes is significantly enriched for several gene ontology terms, including protein kinase activity and localization to numerous lumens (Table 8). We speculated that particular types of functions would be enriched in the genes with nonlinear associations, making the gene ontology molecular function term protein kinase activity particularly interesting. We highlight the protein kinase genes to see if they have any distinct qualities within the network. Interestingly, 81% of the strong
<table>
<thead>
<tr>
<th>Category</th>
<th>Term</th>
<th>Count</th>
<th>Benjamini</th>
<th>FDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>GOTERM_BP_FAT</td>
<td>GO:0007049--cell cycle</td>
<td>51</td>
<td>2.20E-04</td>
<td>1.87E-04</td>
</tr>
<tr>
<td>GOTERM_BP_FAT</td>
<td>GO:0006974--response to DNA damage stimulus</td>
<td>28</td>
<td>0.008485</td>
<td>0.014521</td>
</tr>
<tr>
<td>GOTERM_MF_FAT</td>
<td>GO:0004672--protein kinase activity</td>
<td>42</td>
<td>0.01277</td>
<td>0.03114</td>
</tr>
<tr>
<td>GOTERM_BP_FAT</td>
<td>GO:0006281--DNA repair</td>
<td>23</td>
<td>0.017248</td>
<td>0.044469</td>
</tr>
<tr>
<td>GOTERM_BP_FAT</td>
<td>GO:0033554--cellular response to stress</td>
<td>33</td>
<td>0.021297</td>
<td>0.07335</td>
</tr>
<tr>
<td>GOTERM_CC_FAT</td>
<td>GO:0070013--intracellular organelle lumen</td>
<td>66</td>
<td>0.034233</td>
<td>0.136685</td>
</tr>
<tr>
<td>GOTERM_CC_FAT</td>
<td>GO:0043233--organelle lumen</td>
<td>66</td>
<td>0.018743</td>
<td>0.148487</td>
</tr>
<tr>
<td>GOTERM_CC_FAT</td>
<td>GO:0031981--nuclear lumen</td>
<td>54</td>
<td>0.017591</td>
<td>0.208852</td>
</tr>
<tr>
<td>GOTERM_CC_FAT</td>
<td>GO:0031974--membrane-enclosed lumen</td>
<td>67</td>
<td>0.013715</td>
<td>0.21668</td>
</tr>
</tbody>
</table>

Table 8 – Gene ontology enrichment of the giant component of the network composed of nonlinear associations with $\Delta$ greater than 0.85. The background used is the mouse annotated genome.

Nonlinear interactions are part of the giant component, indicating that nonlinear associations often associate with each other.
It is tedious to visually analyze every interaction, so we use the kinases as a method to focus onto fewer associations. The nonlinear associations look to show rather consistent pattern of association coined ‘noncoexistence’, where the abundance of one gene transcript, the other is less abundant (Reshef et al. 2012). However, when plotting on a log scale, some of the associations, such as Mapk14 and Rac1, have clear quadratic relationships (Figure 13A). At low and high levels of Mapk14, Rac1 has relatively low expression, while at intermediate levels of Mapk14, expression is high. This pattern existed in some associations with incident nodes, including Mapk14 and Arfgap2, and

Figure 13 – Select gene expression plots for nonlinear associations. Gene expression is in tag count per million (tpm), log scale. Different tissue differentiation series are represented by different colors (see table 7).
Did1 and Rac1 (Figure 13B, 12C). Furthermore, the different differentiation series still appear to cluster quite clearly, with erythrocytic differentiation having high Mapk14 yet relatively low Arfgap2 expression, while bone-derived macrophage response to tuberculosis infection having low Mapk14 expression and relatively low Arfgap2 expression. All the other datasets have intermediate Mapk14 expression and high Arfgap2 expression (Figure 13C).

In another nonlinear association involving a protein kinase, Cdkl3, there appears to be strikingly clear clusters of unique ratios of gene expression differentiable by particular tissue differentiation series (Figure 13D).

Notably, the average level of expression appears to be lower with many of the kinases, suggesting a lower threshold of gene expression required for functionality.
We also investigate the second largest component of the network generated by the 0.85 threshold. With 24 nodes, it is several orders of magnitude smaller than the giant component. This component appears to be centered on one hub node, Ubl3 (ubiquitin-like 3) with a degree of 12 (compared to the network average of 3.08). The rest of the component consists of branches from three nodes incident to the central hub node, X99384, Ak4, and Ulk3 (Figure 14).

This cluster was enriched for peptidyl-prolyl cis-trans isomerase activity as a molecular function, due to the presence of Fkbp1a, Fkbp1b, and Fkbp2. All three of these genes are present in one of the aforementioned branches (Figure 14).

**Figure 14** – The second largest component of strong nonlinear associations ($\Delta \geq 0.85$). The spatial layout is done using force atlas layout.
The gene expression plots for the Ubl3 and its incidental nodes are all strikingly similar in not only the pattern of association, but also in the fact that the differentiation series that appear to be responsible are consistent between all 12 associations (Figure 15). Furthermore, the pattern is consistent even though the incident nodes have a varied range of expression from 0-70 tpm to 0-7000 tpm. In every instance, Ubl3 is expressed primarily in the epithelial differentiation time course, while genes represented by the incident nodes are expressed in the tuberculosis infection of bone marrow-derived macrophage. Furthermore, Ubl3 is only expressed when the level of expression of the incident genes is relatively low; the inverse is also true. This is another instance of nonlinear association exhibiting noncoexistence (Figure 15). In the other time courses, none of these genes show a considerable amount of expression.
Figure 15 – The gene expression association plots and corresponding $\Delta$ for the hub node Ubl3 (x-axis) and 8 of its 12 incident nonlinearly-related nodes on the y-axis (see figure 12). Gene expression is in tag count per million (tpm). Different tissue differentiation series are represented by different colors (see table 7). Note that the y-axis scales may vary graph to graph.
The nonlinear associations that do not involved the hub node Ubl3 are varied in their appearance. In the instance of Fkbp2 and Ulk3, the nonlinear relationship appears to be due to a potential bifurcation of the two genes (Figure 16B). Nearly all of the significant gene expression is accounted by one dataset, bone-derived macrophage response to tuberculosis. The relation between Ak4 (adenylate kinase) and Spp1 (secreted phosphoprotein 1) appears to be inverse – at low Ak4 expression levels, Spp1 is expressed at relatively high levels, while at high Ak4 expression levels, the Spp1 expression is low (Figure 16). Again, this relationship appears to be confined entirely to

**Figure 16** – Select association plots a) Ak4 and Spp1, a nonlinear association not involving Ubl3 b) Fkbp2 and Ulk3 a nonlinear association not involving Ubl3 and with a distinctly different distribution c) Tcn2 and Ak4, which do not have a nonlinear association. Gene expression is in tag count per million (tpm). Different tissue differentiation series are represented by different colors (see table 7).
one dataset. Visual inspection of association plots between genes that are not nonlinearly associated is in agreement with our measurement of nonlinearity (Figure 16C).

There are is the similarity that most all these genes in this particular component have relatively high levels of expression in bone-derived macrophage response to tuberculosis.
4. Discussion

With the advent of the maximal information coefficient, a new way for detecting associations in large datasets had been found. Here, we apply this statistics and derivations of this statistic successfully to the unreleased, high-quality FANTOM HelicosCage gene expression dataset.

4.1 Implementation and Optimization

The first challenge we faced was to utilize MIC in the context of MatLab, and then proceed to apply it at a scale which had not yet been done. After coming up with several different versions of code, we manage to process datasets faster than the original code provided by the developers of the MINE statistical package. This is much due to the help of a C-based version, which speeds up the calculations due to the inherent qualities of the C coding language (Nikishkov et al. 2003). Furthermore, we are also able to avoid the memory limitations often associated with Java-based code by using the C-based code (Albanese et al. 2012). The end result is a MatLab wrapper that is up to twice as fast as the original Java-based code and not limited by memory. Because of MINE’s wide range of application to anything from detecting associations in gene expression datasets (as we have done here), to looking for associations amongst performance statistics of a professional sports league (as done by Reshef et al. 2011), the wrapper produced here is a product that will be beneficial to other members of the Ravasi group and the Computational Bioscience Research Center.
4.2 Visual Analysis of Networks with Linear and Nonlinear association

Our efforts to create networks for visual analysis were mixed. Initially, we showed that the resolution of modularity has a significant effect on how well modules represent clusters of linearity. Even looking at specific modules and then reapplying modularity did not seem to generate the clusters of linearity we envisioned. This might reflect issues with the modularity algorithm applied.

Indeed, there are many different approaches to calculating modularity. We use an approach that optimizes a measure of modularity. Other approaches search for interedges and remove them in order to split and define communities (Newman & Girvan 2004). Others merge adjacent nodes based on a measurement of similarity using random walking. This is based off the assumption that each node is connected to itself, and random walking can determine how interconnected the nodes surrounding a particular node are (Pau & Latapy, 2006). All these methods claim some advantage, whether it is the speed of computation or quality of the modules. In our analysis, we only use one without exploring other approaches. It is also important to mention that our edge weight is rather unique in that it is actually two types of measurement, one for linearity and one for nonlinearity, ‘fused’ together. This allows us to able have spectrum of linearity where as a value approaches 1, it is linear; as it approaches 0, it is nonlinear. However, because of our artificial combination of two different statistics, the values at the end of the spectrums are not actually related to each other, which is a deviation from the normal scenario. This probably has unforeseeable consequences on our calculation of modularity.

Nonetheless, we are able to carefully search through the 20 modules that are present when modularity is applied at resolution of 0.5 and identify a few instances where one
cluster of linear associations was nonlinearly associated with another cluster. In the blue module of Figure 7, several gene ontology terms were enriched. Notably, the genes responsible for the enrichment for DNA repair and cancer susceptibility are all part of the 11 gene Fanconi anemia pathway, which is responsible for stabilizing the genome (Levitus et al. 2005). Interestingly, only 9 of the 11 Fanconi anemia genes are identified as having linear relationship with each other in our analysis, suggesting that perhaps the other 2 Fanconi anemia genes have a different functionality. Because the only known function of the Fanconi anemia genes is to stabilize the genome, then one could hypothesize that in the tissues that these genes are most expressed are going through some period of genome instability that needs to be counteracted by upregulation of the Fanconi Anemia pathway

The other genes that are linearly correlated with the Fanc genes and exist in the red module include Fat1, Fat2, and Fat4, which are all transmembrane cadherin proteins involved in cellular proliferation and planar polarity (Valletta et al. 2012). The means by which the Fat proteins regulate these processes is not yet clear, but some have implicated the nuclear factor kappa B pathway as a potential downstream target (Shang et al. 2009). In our analysis, the Fat genes have numerous nonlinear associations, including with Kremen1, which is a transmembrane receptor that negatively modulates WNT/beta-catenin signaling to prevent proliferation (Nakamura et al. 2001). Together, this information might suggest that these represent alternative methods of transmitting signals across the membrane to regulate cell proliferation. With Kremen1, which is expressed predominantly in the mesenchymal stem cell differentiation to adipocytes and osteocytes dataset might be necessary for dynamic control of proliferation. Conversely, the Fat
genes, which are present at greatest levels the neuron and T-cell differentiation course, might be regulating proliferation in these tissues via the nuclear factor kappa b pathway (Saburi et al. 2012). Notably, in the one dataset that does not actually involve proliferation, macrophage response to tuberculosis does not have the any of the aforementioned genes expressed to a high degree, supporting the hypothesis that proliferation is a somehow being regulated by these genes.

Alternatively, this interaction could be stranger than one could anticipate which is still likely considering that Fat proteins can have a positive or negative affect on cell proliferation and cell polarity depending on the tissue it is expressed in and with what proteins it is interacting with (Saburi et al. 2012).

Mrpl21, a mitochondrial ribosomal protein, has nonlinear interactions with several of Fat and Fanc gene (Figure 7C). Interestingly, it was expressed predominantly in the adipocyte and osteopocyte differentiation series, suggesting that these cells might be undergoing an enrichment of mitochondria. Indeed, adipocytes are known to be mitochondria-rich: the high mitochondria content of brown-adipose tissue in infants is actually responsible for producing warmth to regulate temperature homeostasis (Pauw et al. 2009). In the other tissue sets, the expression is lower but still present, suggesting that mitochondria are dividing at a rate to accommodate cell proliferation. Furthermore, the one time series that does not represent true proliferation (and thus would not have upregulation of mitochondrial related genes), the macrophage response to tuberculosis, lacks Mrp21 expression entirely.
4.3 Nonlinear modules: Ubl3 and incident nodes

Ubl3, ubiquitin-like 3, the hub node of the second largest component of the nonlinear associations in the in vitro mouse differentiation series, is 117a.a. long protein that is poorly characterized. While its structure has been resolved and it has been successfully cloned and expressed (Chadwick et al. 1999), little is known about its functionality. In our analysis, we identify it as a hub node with 12 distinct nonlinear associations (Figure 14). Other ubiquitin-like proteins (UBLs) are known to modify proteins in a manner similar to ubiquitin, targeting them for degradation in a pathway distinct from traditional ubiquitin degradation pathway. The incident nodes have a wide range of functions, but all must be biologically relevant most in the macrophage response to tuberculosis, less so in the epithelial differentiation series, and completely irrelevant in the other tissue differentiation series.

The genes that make up the incident nodes might be putative targets of Ubl3. Indeed, the noncoexistence that we observe between these nodes suggests that the function of Ubl3 could be much like that of other ubiquitin-like proteins (UBLs): to provide an alternative but specific marker for proteasomal degradation. While it is important to note that ubiquitin-related degradation does occur at the protein level, and our measurements are looking at the transcript level, the expression of Ubl3 could be resulting with a degradation of existing targets while also corresponding to a down-regulation in gene expression. Until experimental confirmation, the conclusions about Ubl3 are speculative; nonetheless, it highlights the investigative power of the approach.
4.4 General conclusions of the nonlinear network

Of the strongest nonlinear associations existing in the in vitro mouse tissue differentiation series, protein kinases, molecules that are critical in a wide range of cellular signaling and regulation, are significantly enriched (Table 8). This drives and interesting hypothesis that signaling molecules, like protein kinases, could be interacting with a variety of targets in a nonlinear manner, and that these interactions are also represented at the transcript level. In other words, the cellular processes associated with dynamic response to the environment, such as differentiation, are carefully controlled by a variety of posttranslational modifications, such as protein kinases. These posttranslational modifications are occurring most often in a nonlinear manner mediated by other proteins, like protein kinases, and that this nonlinear control is due to nonlinear gene expression.

Indeed, often times signal propagation is nonlinear, and while this is most often observed at a protein level, prolonged processes such as tissue differentiation, it is reasonable to expect that some genes are going to be nonlinearly expressed (Purvis & Lahav, 2013). For instance, it might be that a signal is transduced via a protein kinase. Kinases, being enzymes, are capable phosphorylating multiple proteins; consequently, a lower expression of kinase relative to its targets would be expected. A similar scenario could play out with membrane proteins responsible for transducing signals across a membrane.

Additionally, nonlinear associations are enriched for genes located in the lumens of organelles, including the nucleus (Table 8). This might be indicative that, because of the relatively large space of the cytoplasm, a gene involved in signaling still needs to be expressed a level nearer the level of its downstream targets order to transduce its signal.
effectively. In the closer confines of an organelle, a gene involved in amplifying signal transduction, such as the aforementioned kinases, would not have to be expressed at an equal level as its targets. In the context of a dynamic process, this could lead to nonlinear associations.

4.5 Further work

During our analyses, we did notice some phenomenon in our data that might be indicative of a less than ideal analysis. For one, there seems to be some erroneous linear associations, though the cause of it is not clear. We did notice some instances where an association was deemed linear, but had a rather clear bifurcation (Figure 9E), suggesting a limitation of the Pearson correlation coefficient. This very well might be due to the high density of low values, which cause the Pearson correlation coefficient to treat interesting data, like that of the bifurcation, as outliers. This could be addressed by removing low gene expression values, especially if the gene expression in another differentiation series or at another time point is several orders of magnitude larger.

Notably, we do not factor in gene expression over time in any of our analysis, despite the data be time course data. This provides an obvious follow up step where the gene expression in each differentiation series can be analyzed with respect for time. The MIC statistic would still be valuable in these analyses to identify associations present over time.

One of the inherent limitations of a computational approach, particularly with in network science, is that most of the conclusions that can be drawn from analyses are merely Indeed, that is a limitation we would like to stress here. While we can hypothesize about potential interactions and functions, the truth is that most of these genes are not very well
characterized. Even more limiting is that we are dealing with many different tissue types, so while a particular set of genes might have a known function in one tissue, their functions are not defined in another tissue. In addition, all these genes might have altered function in a dynamic cellular state like differentiation. With that said, most statements made about single genes would need to be verified experimentally.

The core formulae we use for defining both linear and nonlinear associations could be further developed as well. Firstly, we only consider positive Pearson values, which completely neglect negative associations, no matter how significant they are. A viable alternative would to use the absolute value of $\rho$, which would still restrict the range of $\rho$ from zero to one and keep it comparable to MIC, which also has a range from zero to one.

The data we work with in this project is very rich, and we only develop and explore one approach of inquiry. Other approaches, such as cluster analysis, are viable exploratory data mining techniques that would provide insights different from those made here (Flynn et al. 1999).
REFERENCES


Valletta, D., Czech, B., Thasler, W. E., Müller, M., Bosserhoff, A.-K., & Hellerbrand, C. (2012). Expression and function of the atypical cadherin FAT1 in chronic liver


