Computational Network Analysis of the Anatomical and Genetic Organizations in the Mouse Brain

Shuiwang Ji

Old Dominion University

Follow this and additional works at: https://digitalcommons.odu.edu/computerscience_fac_pubs

Part of the Biochemistry Commons, Computational Biology Commons, Computer Sciences Commons, Molecular Biology Commons, and the Statistics and Probability Commons

Repository Citation

Ji, Shuiwang, "Computational Network Analysis of the Anatomical and Genetic Organizations in the Mouse Brain" (2011). Computer Science Faculty Publications. 89.
https://digitalcommons.odu.edu/computerscience_fac_pubs/89

Original Publication Citation

Gene expression

Computational network analysis of the anatomical and genetic organizations in the mouse brain

Shuiwang Ji
Department of Computer Science, Old Dominion University, Norfolk, VA 23529, USA
Associate Editor: John Quackenbush

ABSTRACT

Motivation: The mammalian central nervous system (CNS) generates high-level behavior and cognitive functions. Elucidating the anatomical and genetic organizations in the CNS is a key step toward understanding the functional brain circuitry. The CNS contains an enormous number of cell types, each with unique gene expression patterns. Therefore, it is of central importance to capture the spatial expression patterns in the brain. Currently, genome-wide atlas of spatial expression patterns in the mouse brain has been made available, and the data are in the form of aligned 3D data arrays. The sheer volume and complexity of these data pose significant challenges for efficient computational analysis.

Results: We employ data reduction and network modeling techniques to explore the anatomical and genetic organizations in the mouse brain. First, to reduce the volume of data, we propose to apply tensor factorization techniques to reduce the data volumes. This tensor formulation treats the stack of 3D volumes as a 4D data array, thereby preserving the mouse brain geometry. We then model the anatomical and genetic organizations as graphical models. To improve the robustness and efficiency of network modeling, we employ stable model selection and efficient sparsity-regularized formulation. Results on network modeling show that our efforts recover known interactions and predicts novel putative correlations.

Availability: The complete results are available at the project website: http://compbio.cs.odu.edu/mouse/
Contact: sj@cs.odu.edu
Supplementary Information: Supplementary data are available at Bioinformatics online.

Received on May 2, 2011; revised on October 1, 2011; accepted on October 3, 2011

1 INTRODUCTION

The mammalian central nervous system (CNS) generates high-level control functions, and knowledge on the anatomical and genetic organizations in this system can elucidate the functional brain circuitry. The enormity of complexity is reflected in the large number of cell types, each with unique gene expression patterns. Therefore, it is of central importance to capture the anatomical localization of gene expressions in the brain. Recent advances in bioimaging technologies, such as the high-throughput in situ hybridization (ISH) technique, have made it possible to capture the spatial expression patterns in the adult mouse brain (Lein et al., 2007). Consequently, genome-scale expression atlases in the form of digital images have been produced at increasing speed and resolution. The marriage of image processing tools and advanced computational methods opens the door for unraveling the functional brain circuitry and the generation of high-level cognitive functions on top of it.

The Allen Brain Atlas (ABA) (Lein et al., 2007) contains 3D atlas of gene expression in the adult mouse brain and is one of the most comprehensive datasets for spatial expression patterns in the mammalian CNS. It provides cellular resolution 3D expression patterns in the male, 56-day-old C57BL mouse brain. In this atlas, genome-wide coverage is available in sagitally oriented sections. In addition, coronal sections at a more refined scale are available for a set of about 4000 genes showing restricted expression patterns. The image data are generated by in situ hybridization using gene-specific probes, followed by slide scanning, 3D image registration to the Allen Reference Atlas (ARA) (Dong, 2009) and expression segmentation (Lein et al., 2007; Ng et al., 2007). This results in a set of spatially aligned 3D volumes of size 67 × 41 × 58, one for each gene, that document the spatial expression patterns of genes in the mouse brain. Efficient and effective analysis of these high-throughput data can shed light on the global function of mammalian CNS (Jones et al., 2009). On the other hand, the sheer volume and complexity of these data pose significant challenges for efficient computational analysis. Hence, computational understanding of these data is limited to unsupervised techniques, which cluster the brain regions into co-expressed groups (Bohland et al., 2010).

In this article, we employ advanced computational techniques to model the anatomical and genetic organizations in the mouse brain as networks. First, to reduce the size of data and accelerate efficient analysis and storage, we propose to apply tensor factorization techniques to reduce the data volumes (Kolda and Bader, 2009; Wrede, 1972). This tensor formulation treats the stack of 3D volumes as a 4D data array, thereby preserving the mouse brain geometry. Based on the reduced data, we model the anatomical and genetic organizations as graphical models in which each vertex represents a spatial location or a gene, and the edges between vertices encode the correlations between locations and genes (Dempster, 1972; Edwards, 2000). To improve the efficiency of network modeling, we employ an approximate formulation for Gaussian graphical modeling, which involves a series of sparse regularized regression problems (Meinshausen and Bühlmann, 2006). The efficiency of this approximate formulation enables us to employ a robust estimation technique known as stability selection (Meinshausen and Bühlmann, 2010), which estimate and combine multiple models based on resampling.

We apply the data reduction and network modeling techniques to learn the anatomical and genetic networks underlying the mouse brain...
brain using the ABA expression volume data. Results show that the expression patterns of spatially adjacent voxels tend to correlate. We also observe that the expression patterns of certain brain structures are correlated to the patterns of a large number of other regions, some of which are spatially distant. In-depth analysis reveals that such correlation captures existing knowledge on the brain functionality. Our efforts on genetic network modeling identify functionally related genes that act in a concerted manner in the mouse brain.

2 HIGH-ORDER FEATURE EXTRACTION VIA TENSOR FACTORIZATION

In ABA, the ISH image series of each gene are aligned to the ARA. To faithfully capture the mouse brain geometry, a 3D grid is employed to divide the 3D ARA space into quadrats, and expression information within each quadrat is summarized. Specifically, an expression segmentation algorithm is employed to identify expressed cells, and then an expression energy value is computed from each voxel as a function of the intensity and density of expression within that voxel. These image processing steps convert each expression pattern into a 3D volume. To enable the application of matrix computation techniques such as the singular value decomposition (SVD), these volumes are usually converted to vectors and stacked into a data matrix (Bolholt et al., 2010). However, such conversion fails to retain the spatial locality and other high-order information in the expression volumes.

To overcome this limitation, we propose to treat the 3D volumes as 3D tensors and stack them together to form a 4D tensor. We then employ tensor factorization techniques to reduce the dimensionality of this 4D tensor along each mode, resulting in significant data compression.

A key advantage of this tensor representation is that the associated tensor computation techniques, such as high-order SVD and low-rank tensor approximation, can be employed to compress the data without flattening the internal structure of the high-order data array. These techniques approximate the original tensor by a core tensor multiplied by a basis matrix along each mode. Hence, the core tensor and the set of basis matrices give a compact representation of the original tensor, and the core tensor captures the major information in the original tensor.

2.1 Background on tensors

Tensors, also known as multidimensional matrices (Kolda and Bader, 2009; Wede, 1972), are higher order generalizations of vectors (first-order tensors) and matrices (second-order tensors). The order of a tensor is the number of indices, also known as modes or ways. In this article, tensors are denoted by boldface Euler script letters, e.g. $\mathbf{X} \in \mathbb{R}^{(1) \times (2) \times \cdots \times (N)}$, and its elements are denoted as $x_{i_1,\ldots,i_N}$, where $1 \leq i_k \leq d_k$ for $n = 1,\ldots, N$. As a generalization of matrix multiplication, the $n$-mode tensor-matrix product defines the multiplication of a tensor by a matrix in mode $n$ (Lathauwer et al., 2000a). The $n$-mode product of a tensor $\mathbf{X} \in \mathbb{R}^{(1) \times \cdots \times (N)}$ with a matrix $\mathbf{A} \in \mathbb{R}^{d_1 \times \cdots \times d_N}$ is denoted by $\mathbf{X} \times_n \mathbf{A}$. The result is a tensor of size $d_1 \times \cdots \times d_{n-1} \times d_n \times d_{n+1} \times \cdots \times d_N$ defined elementwise as:

$$\mathbf{X} \times_n \mathbf{A} = \sum_{j=1}^{d_j} x_{i_1,\ldots,i_{n-1},j,i_{n+1},\ldots,i_N} a_{j,i_{n+1}} \cdots a_{j,i_N} \mathbf{R}_n,$$

Let $\mathbf{A} \in \mathbb{R}^{d_1 \times \cdots \times d_N}$ be another tensor of the same size as $\mathbf{X}$. The scalar product of these two tensors is defined as:

$$\langle \mathbf{X}, \mathbf{A} \rangle = \sum_{i_1=1}^{d_1} \sum_{i_2=1}^{d_2} \cdots \sum_{i_N=1}^{d_N} x_{i_1,\ldots,i_N} a_{i_1,i_2,\ldots,i_N} \cdots a_{i_N,i_1} \cdots a_{i_N,i_2} \cdots a_{i_1,i_N},$$

where $1 \leq \sum_{k=n+1}^{N} i_k \leq d_n$.

2.2 Tensor factorization

High-order singular value decomposition (HOSVD) (Lathauwer et al., 2000a) is a generalization of the SVD for matrices. Given a tensor $\mathbf{X} \in \mathbb{R}^{(1) \times (2) \times \cdots \times (N)}$, its HOSVD can be expressed as:

$$\mathbf{X} = \mathbf{X}_1 \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3 \times_4 \mathbf{U}_4 \times_5 \mathbf{U}_5, \quad \mathbf{X} = \sum_{n=1}^{N} \mathbf{X}_n \times_n \mathbf{U}_n,$$

where $\mathbf{X} \in \mathbb{R}^{(1) \times \cdots \times (N)}$, $\mathbf{X}_n$, and $\mathbf{U}_n$ are fixed, orthogonal matrices. In HOSVD, the basis matrices $\mathbf{U}_n$ are computed as the left singular matrices of the mode-$n$ unfolding of $\mathbf{X}$, and the core tensor can then be computed as:

$$\mathbf{X} = \mathbf{X}_1 \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3 \times_4 \mathbf{U}_4 \times_5 \mathbf{U}_5,$$

where $\mathbf{C} = \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3 \times_4 \mathbf{U}_4 \times_5 \mathbf{U}_5$.

It follows from this definition that $\mathbf{X}$ can be expressed as:

$$\mathbf{X} = \mathbf{X} = \mathbf{U}_1 \times_1 \mathbf{X}_2 \times_2 \mathbf{U}_2 \times_3 \mathbf{X}_3 \times_3 \mathbf{U}_3 \times_4 \mathbf{X}_4 \times_4 \mathbf{U}_4 \times_5 \mathbf{X}_5 \times_5 \mathbf{U}_5,$$

where $\mathbf{C} = \mathbf{X}_1 \times_1 \mathbf{X}_2 \times_2 \mathbf{X}_3 \times_3 \mathbf{X}_4 \times_4 \mathbf{X}_5 \times_5 \mathbf{X}_5 \times_5 \mathbf{U}_5$.

Hence, the key to the low-rank tensor factorization problem is to compute the basis matrices. The factorization of a 3D tensor is illustrated in Figure 1.

One of the commonly used algorithms to compute the basis matrices is the alternating least squares (ALS) method (Lathauwer et al., 2000b). In each iteration of this method, one of the basis matrices is optimized while all others are fixed. Specifically, when $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(N)}$ are fixed, we first compute $\mathbf{X}^{(1)} \times_1 \mathbf{X}^{(2)} \times_2 \mathbf{X}^{(3)} \times_3 \mathbf{X}^{(4)} \times_4 \mathbf{X}^{(5)} \times_5 \mathbf{X}^{(6)}$.

When the columns of $\mathbf{X}^{(1)}$ are obtained as the first $\mathbf{R}_1$ columns of the left singular matrix of $\mathbf{X}^{(1)}$, which is the mode-$n$ unfolding of $\mathbf{X}$. In ALS, the basis matrices are usually initialized as the truncated basis.
The 4D tensor of gene expression obtained from the ABA is factorized as

\[ \mathbf{X} = \mathbf{U} \mathbf{V}^T \]

where \( \mathbf{X} \) is the original tensor while its size is significantly reduced. This data reduction step is critical for the subsequent efficient analysis. Based on the reduced data, we employ sparse graphical modeling approaches to construct the anatomical and genetic networks underlying the mouse brain.

### 3.1 A sparsity regularization formulation

Gaussian graphical models are a class of methods for modeling the relationships among a set of variables (Edwards, 2000; Whittaker, 1990). In this formulation, the d-dimensional variable \( \mathbf{x} = (x_1, x_2, \ldots, x_d)^T \) is assumed to follow a multivariate Gaussian distribution \( \mathbf{x} \sim \mathcal{N}(\mu, \Sigma) \), where \( \mu \in \mathbb{R}^d \) and \( \Sigma \in \mathbb{R}^{d \times d} \) are the mean and covariance, respectively. The conditional dependency between pairs of variables can be encoded into a graphical model in which vertices represent variables and edges characterize the conditional dependency between variables. In particular, there is an edge between nodes corresponding to \( x_i \) and \( x_j \) if and only if these two variables are conditionally dependent given all other variables. This is equivalent to the saying that there exists an edge between nodes corresponding to \( x_i \) and \( x_j \) if and only if the \((i,j)\)th entry of the inverse covariance matrix (also known as concentration matrix) \( \mathbf{Q} = \mathbf{\Sigma}^{-1} \) is non-zero (Dempster, 1972; Edwards, 2000).

This correspondence is illustrated in Figure 2.

Given a set of \( n \) observations \( y_1, y_2, \ldots, y_n \), the concentration matrix can be estimated by maximizing the penalized log-likelihood as follows (Banerjee et al., 2008; Friedman et al., 2008; Yuan and Lin, 2007):

\[ \hat{\mathbf{Q}} = \arg \max_{\mathbf{Q} \in \mathbb{R}^{d \times d}} \log \det \mathbf{Q} - \frac{1}{2} \mathbf{y}^T \mathbf{Q}^{-1} \mathbf{y} \]  

where \( \det \mathbf{Q} \) is the determinant of \( \mathbf{Q} \), \( \mathbf{Q} \succeq 0 \) represents that \( \mathbf{Q} \) is positive definite, \( \mathbf{y} \) denotes the empirical covariance matrix computed from data, and \( \|\mathbf{Q}\|_1 \) is the 1-norm of \( \mathbf{Q} \), which is the sum of the absolute values of the entries of \( \mathbf{Q} \). The first two terms in Equation (8) are the log-likelihood, and the last term is used to enforce that many entries of \( \mathbf{Q} \) are set to zero, yielding a sparsely connected graph. This formulation has been used to model the gene networks in Arabidopsis thaliana (Wille et al., 2004). The optimization problem in Equation (8) is convex and can be solved by several algorithms such as the interior point method (Banerjee et al., 2008) and the graphical lasso algorithm (Friedman et al., 2008). However, all these algorithms are computationally expensive and can only be applied to small-scale problems. For the modeling of mouse brain networks, we have thousands of genes and tens of thousands of voxels; hence, this formulation is not applicable.

\[ \hat{\mathbf{Q}}_{\lambda} = \hat{\mathbf{Q}}_{\lambda} \]  

### 3.2 Robust estimation via stability selection

The regularization parameter \( \lambda \) in Equation (9) controls the trade-off between the sparsity of solution and data fit. Specifically, when \( \lambda \) is set to a very large value, most of the entries of \( \mathbf{w} \) are set to zero. Hence, a challenge in practice is how to select the value for \( \lambda \). Stability selection (Meinshausen and Bühlmann, 2006) addresses this problem by ideas similar to the ensemble learning methods widely used in machine learning (Bühlmann, 2004). In stability selection, we choose a set of \( \lambda \) values denoted by \( \Lambda \), instead of a single \( \lambda \) value. For each \( \lambda \in \Lambda \), we compute the selection probability for each variable, which is defined as the probability of each variable being selected when randomly subsampling from the data. Formally, let \( I \) be a random subsample of \( y_1, y_2, \ldots, y_n \) of size \( \lfloor n/2 \rfloor \) drawn without replacement. The selection probability for variable \( x_i \) is defined as

\[ \hat{p}_{\lambda}(i) = \mathbb{P}(x_i \in \mathcal{A}^*(I)) \]  

**Fig. 2.** Illustration of the concentration matrix (A) and the corresponding graphical model (B). The zero entries in the concentration matrix are unfilled, while the non-zero entries are filled with green. In this example, \( x_1 \) and \( x_5 \) are conditionally independent given all other variables.

In Meinshausen and Bühlmann (2006), an approximate formulation is proposed to learn Gaussian graphical models by solving a series of sparse regression problems. Specifically, the conditional dependencies between \( x_i \) and all other variables are learned by solving the following 1-norm penalized regression problem known as lasso (Tibshirani, 1996):

\[ \hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^d} \|y - \mathbf{X}^T \mathbf{w}\|_2^2 + \lambda \|\mathbf{w}\|_1, \]  

where \( y = (y_1, y_2, \ldots, y_n)^T \in \mathbb{R}^{d \times (n-1)} \) is the data matrix obtained by removing the \( i \)-th data item. The conditional dependencies between \( x_i \) and all other variables are obtained from the corresponding components in the weight vector \( \mathbf{w} \). Note that the regression of \( x_i \) onto \( x_j \) and that of \( x_j \) onto \( x_i \) may not give the same result. Hence, two simple schemes, based on logic operations or and and, are proposed to interpret the results. In the first scheme, two variables are considered to be conditionally dependent if either of them yields non-zero weight (Meinshausen and Bühlmann, 2006). In the second scheme, they are considered as conditionally dependent if both of them give non-zero weights. The first scheme is employed in this work (Meinshausen and Bühlmann, 2006). The pairwise relationships between all pairs of variables can be obtained by running the sparse regression problem in Equation (9) for each variable. A critical observation that leads to the efficiency of the formulation in Equation (9) is that it involves solving \( d \) independent lasso problems, one for each variable. The lasso problem can be solved very efficiently by many algorithms such as the accelerated gradient method (Liu et al., 2009). It has been shown that this sparse regression formulation of Gaussian graphical modeling maximizes the pseudo likelihood (Friedman et al., 2010) and is an approximation to the maximum likelihood scheme in Equation (8) (Banerjee et al., 2008; Friedman et al., 2008). In particular, the exact maximization of log likelihood involves solving the lasso problems iteratively as in the graphical lasso algorithm (Friedman et al., 2008), and the formulation in Equation (9) can be considered as a one-step approximation to the maximum likelihood scheme. We employ this approximate formulation to learn the mouse brain networks due to its efficiency.
where $A^t(I)$ denotes the set of variables that have been selected when $I$ is used as the sample and the regularization parameter is set to $\lambda$. Note that this definition of $A^t(I)$ is independent of the specific method used for variable selection. The probability in Equation (10) is with respect to both the random sampling and other sources of randomness such as that induced by the algorithm as we discuss below. For every variable $x_i$, the stability path is given by the selection probabilities $\hat{\pi}_k^I, k \in \Lambda$. It has been shown in Meinshausen and Bühlmann (2010) that 100% random resampling is sufficient to obtain accurate estimates.

Based on the selection probabilities, stable variables can be defined. For a cutoff $\pi_{th}$ with $0 < \pi_{th} < 1$ and a set of parameters $\Lambda$, the set of stable variables are defined as

$$\hat{S}_{\text{stable}} = \{x_i : \max_{k \in \Lambda} (\hat{\pi}_k^I) > \pi_{th}\}.$$  

(11)

By choosing the set of stable variables under the control of the cutoff $\pi_{th}$, we keep variables with a high selection probability and discard those with low selection probabilities. It has been shown that the results of stability selection vary little for sensible choices of the cutoff $\pi_{th}$ and the parameter set $\Lambda$.

It has also been shown that performance can be further improved if additional randomness is introduced into the lasso problem in Equation (9). In particular, we can randomize the amount of regularization for each variable by solving the following problem:

$$\hat{w} = \arg \min_{w \in \mathbb{R}^d} ||y - Xw||^2 + \lambda \sum_{k \in \Lambda} |w_k|^2,$$  

(12)

where $D_k^I = \{1, \ldots, d_k^I\}$, $\epsilon_k$ are IID random variables in $[0,1]$ and $\alpha \in (0,1]$ is a user-specified weakness factor.

4 RESULTS AND DISCUSSION

4.1 Experimental setup

In this article, we use a set of expression volumes for 3012 genes documented in the coronal sections as in Bohland et al. (2010). This set of genes exhibit restricted expression patterns and thus are of high neurobiological interest. For anatomical network modeling, we only use the left hemisphere voxels, since only this part of the brain is annotated in ARA. This gives rise to a 4D tensor of size $3012 \times 67 \times 41 \times 33$ in which the first index corresponds to genes, and the other three indices represent the rostral–caudal, dorsal–ventral and left–right spatial directions, respectively. In tensor factorization, we keep the dimensionality of the last three modes while reduce the dimensionality of the first mode, since we are interested in modeling the relationships among brain voxels. For genetic network modeling, we use the full volumes, and the size of our 4D tensor is $3012 \times 67 \times 41 \times 58$. In this case, we keep the dimensionality of the first mode while reducing the dimensionality of the other three modes.

The computational experiments were performed on a cluster consisting of 256 cores and 512 GB RAM. The lasso formulation was solved using the SLEP package (Liu et al., 2009). We can determine the $\lambda$ value that enforces $w$ to be a zero vector in Equation (9) (Liu et al., 2009), and this $\lambda$ value is denoted as $\lambda_{max}$. Then we set $\Lambda = \{0, \lambda_{max}, 0.2\lambda_{max}, \ldots, 0.9\lambda_{max}\}$. The selection probabilities were estimated on 100 random resampling, and the weakness factor $\alpha$ was set to 0.8. The sizes of reduced data were set to retain 90 and 80% of the original information for anatomical and genetic network modeling, respectively, based on the computational resource requirements. Specifically, the size of the reduced tensor is $179 \times 67 \times 41 \times 33$ in anatomical network modeling and is $3012 \times 22 \times 13 \times 19$ in genetic network modeling.

4.2 Results on anatomical network modeling

Computational modeling of the anatomical organization in the mouse brain yields a graph in 3D space in which the vertices represent brain regions, and the edges characterize the expression correlations between regions. The correlation patterns can be visualized by showing slices of the 3D brain network on 2D planes. Figure 3 shows one slice of the brain network along the coronal section. We can observe that most of the edges connect adjacent regions, showing that spatially adjacent regions tend to exhibit correlated expression patterns. Note that these correlation patterns are learned without knowing the spatial locations of voxels.

Although most of the edges connect spatially adjacent regions, there are apparent exceptions. A slice-by-slice examination of the entire anatomical networks at multiple cutoffs reveal that the voxels annotated as dentate gyrus (DG) in the ARA are highly correlated to many voxels in distant regions as shown in Figure 4. According to classical neuroanatomy, the DG plays an important role in learning and memory by processing and representing spatial information, and it has always been a topic of intense interest (Scharfman, 2007). It has been shown that the DG receives multiple sensory inputs including vestibular, olfactory, visual, auditory and somatosensory from its upstream perirhinal cortex and entorhinal cortex. It plays the role of a gate or filter, blocking or filtering excitatory activity from the inputs and controlling the amount of excitation that is propagated to the downstream hippocampus (Scharfman, 2007). A close examination of Figure 4 shows that the correlation patterns are largely consistent with those classical results. A more quantitative analysis of the results show that the correlation patterns obtained solely based on gene expressions match well with the known functions of DG. In particular, the expression patterns of the DG is highly correlated to those of the cerebral cortex and the main olfactory bulb, which provide sensory inputs to DG. In addition, DG is highly correlated to the hippocampal region and the retrohippocampal region, propagating the filtered signals to its downstream regions. We also observe that the intra-DG correlations dominate, demonstrating again that most of the edges connect spatially adjacent regions. Besides the correlations with known...
functions, our modeling of the anatomical networks also identifies many new relationships with DG that are not known from classical anatomical studies.

Based on the obtained networks in 3D space, a variety of network analysis and visualization techniques can be employed to analyze the anatomical organization in the mouse CNS. In Bohland et al. (2010), the $K$-means algorithm is used to cluster the brain voxels into groups based on dimensionality reduced expression data, and a metric known as the $S$ index was employed to quantitatively characterize the correspondence of the clustering results with the classical anatomy reflected in the ARA annotations. Specifically, let $R = \{r_1, \ldots, r_N\}$ be a partition of the set of brain voxels in which each $r_i$ comprises the set of indices of the voxels that map to that cluster (or anatomical label). The spatial overlap between a region from the ARA and the clustering result is defined as:

$$P_{ij} = |r_i \cap r_j| / |r_i|$$

From the $P_{ij}$ values that are computed over all pairs of ARA regions and cluster result, we can then derive a global scalar index of similarity between the two partitions. Since $P_{ij} \neq P_{ji}$, $X_{ij}$ is defined as $X_{ij} = \max\{P_{ij}, P_{ji}\}$ along with $W_{ij} = U_{ij} / \sum U_{ij}$, where $U_{ij} = \min(|r_i|, |r_j|)$ if $X_{ij} > 0$ and 0 otherwise. Finally, the $S$ index is defined as $S = 1 - 4 \sum W_{ij} X_{ij} (1 - X_{ij})$.

To compare our network modeling method with the $K$-means clustering, we apply the leading eigenvector community detection algorithm proposed by Newman (2006) and treat each detected community as a cluster. Since different cutoff values $\pi_{thr}$ in the stability selection yield different graphs, we vary $\pi_{thr}$ from 0.5 to 0.85 and detect communities from each of the resulting graphs. We then run $K$-means with $K$ equal to the number of communities so that the results are comparable. Since the results of $K$-means depend on the initialization, we run this algorithm 10 times and choose the one with the best result. We compute the $S$ index for each case and report the results in Figure 5. We can observe that the community detection results consistently give higher $S$ index values, indicating that the structures of our anatomical networks are in higher accordance with the classical anatomy. We also plot the number of detected communities as the cutoff changes in Figure 5. We can see that the number of communities lies approximately between 100 and 250, which is largely in correspondence with the number of structures in classical anatomy. Detailed results on community identification are provided in the Supplementary Material.

The classical anatomy was created mainly based on brain functions. Since functions are mainly determined by gene expression, the expression patterns within anatomical structures should be more correlated than those across structures. To validate this hypothesis, we show the distribution of the edges within and across the anatomical structures when $\pi_{thr} = 0.5$ in Figure 6. We also show the number of edges within and across structures when the cutoff varies from 0.2 to 0.9. We can observe that the edges within structures dominate in all cases, indicating that the expression patterns within classical anatomy are highly correlated. We can also observe from Figure 6 that the proportion of edges within anatomical structures increases as the cutoff increases. This indicates that most of the cross-structure edges have relatively small selection probabilities, and they are removed as the cutoff increases. The ranked lists of regions in terms of the number of connections are provided in the Supplementary Material.

### 4.3 Results on genetic network modeling

Modeling of the gene interactions using the techniques described in Section 3 yields a network consisting of 3012 vertices in which vertices represent genes, and edges characterize the correlations...
Fig. 5. Comparison of the communities detected in the anatomical networks and the K-means clustering results. (A) Shows the S index comparison between the anatomical structures in ARA and the results of community detection and K-means. (B) Shows the number of communities as the cutoff changes.

Fig. 6. Visualization of the edge distribution within and between different brain regions. (A) Shows the distribution of the edges within and across the anatomical structures when $\pi_{th} = 0.5$. The rows and columns correspond to the structures annotated in the ARA. This matrix is normalized to the interval [0, 1] row by row, and hence it is not symmetric. Each row indicates the proportion of a particular structure’s edges that connect to other structures. In particular, each entry $(i,j)$ in the matrix represents the proportion of structure $i$’s edges that connect to structure $j$. (B) Shows the number of edges within and between anatomical structures in ARA as the cutoff changes.

between genes. Since genes involved in the same pathway usually exhibit similar expression patterns, correlated expression patterns may imply similar biological functions. We hence use Gene Ontology (GO) (Ashburner et al., 2000) to evaluate the functional relationships among tightly connected genes in the network. In particular, we consider a gene and its direct neighbors as a group (Gustafsson et al., 2005) and evaluate the functional enrichment of each group using the hypergeometric distribution (Boyle et al., 2004). We apply Bonferroni correction for multiple hypothesis testing and consider GO terms with corrected $P < 0.05$ as statistically significant (Boyle et al., 2004). We vary the cutoff and observe that the set of highly connected genes are largely consistent (details provided in the Supplementary Material). We report the top 10 genes with the largest number of connections in Table 1 when $\pi_{th} = 0.8$ and show slices of their expression patterns in the Supplementary Material. We can observe that all these groups are highly enriched with the biological function binding or protein binding, implicating that they are likely to encode transcription factors. Among these 10 genes, the APP encodes an integral membrane protein expressed in many tissues and concentrated in the synapses of neurons. Homologous proteins have been identified in other organisms such as Drosophila, C.elegans and all mammals. APP is best known for its association with the Alzheimer’s disease, and mutations in critical regions of APP cause familial susceptibility to Alzheimer’s disease. It would be interesting to investigate how the ‘hubness’ of APP is related to CNS disease.
5 CONCLUSIONS

We model the anatomical and genetic organizations in the mouse brain as networks. To enable robust and efficient network construction, we employ tensor factorization techniques to reduce the data volumes. The resulting networks recover known relations and predict novel correlations not known from the literature. The employed network modeling formulation is an approximate scheme. It would be interesting to compare this approximate formulation with the exact one on small datasets, where exact optimization can be applied. The proposed methods can be applied to model other biological systems, such as the Drosophila transcriptional networks. We will explore the network modeling of other biological systems in the future.

Funding: This work was supported by Old Dominion University.

Conflict of Interest: none declared.

REFERENCES


Liu, J. et al. (2009) SLEP: Sparse Learning with Efficient Projections. Arizona State University, Tempe, Arizona, USA.


