Journal of the American Statistical Association
September 2017 ■ Volume 112 ■ Number 519

Taylor & Francis
Taylor & Francis Group

Journal of the American Statistical Association

ISSN: 0162-1459 (Print) 1537-274X (Online) Journal homepage: http://amstat.tandfonline.com/loi/uasa20

Finding Common Modules in a Time-Varying Network with Application to the Drosophila Melanogaster Gene Regulation Network

Jingfei Zhang & Jiguo Cao

To cite this article: Jingfei Zhang & Jiguo Cao (2017) Finding Common Modules in a Time-Varying Network with Application to the Drosophila Melanogaster Gene Regulation Network, Journal of the American Statistical Association, 112:519, 994-1008, DOI: 10.1080/01621459.2016.1260465

To link to this article: http://dx.doi.org/10.1080/01621459.2016.1260465

JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION
2017, VOL. 112, NO. 519, 994–1008, Applications and Case Studies
https://doi.org/10.1080/01621459.2016.1260465

Taylor & Francis
Taylor & Francis Group

Finding Common Modules in a Time-Varying Network with Application to the *Drosophila Melanogaster* Gene Regulation Network

Jingfei Zhang[a] and Jiguo Cao[b]

[a]Department of Management Science, School of Business Administration, University of Miami, Miami, FL; [b]Department of Statistics and Actuarial Science, Simon Fraser University, Burnaby, BC, Canada

ABSTRACT
Finding functional modules in gene regulation networks is an important task in systems biology. Many methods have been proposed for finding communities in static networks; however, the application of such methods is limited due to the dynamic nature of gene regulation networks. In this article, we first propose a statistical framework for detecting common modules in the *Drosophila melanogaster* time-varying gene regulation network. We then develop both a significance test and a robustness test for the identified modular structure. We apply an enrichment analysis to our community findings, which reveals interesting results. Moreover, we investigate the consistency property of our proposed method under a time-varying stochastic block model framework with a temporal correlation structure. Although we focus on gene regulation networks in our work, our method is general and can be applied to other time-varying networks. Supplementary materials for this article are available online.

ARTICLE HISTORY
Received October 2015
Accepted October 2016

KEYWORDS
Community structure; Consistency; Gene regulation network; Hypothesis testing; Markov chain Monte Carlo; Robustness

1. Introduction

Molecular genetic studies of *Drosophila melanogaster*, the common fruit fly, have facilitated our understanding of developmental processes in many organisms. For example, since approximately 75% of the diseases known to occur in humans have a recognizable match in the genome of the *Drosophila melanogaster* (Reiter et al. 2001), quantitative descriptions of the gene expressions in the development of the *Drosophila melanogaster* can have large impacts in developmental biology and biomedical studies for humans. Genes, genetic techniques, and other discoveries are often first elucidated in the *Drosophila melanogaster* and then translated to mammalian systems (Pandey and Nichols 2011).

A recent genome-wide microarray profiling of the *Drosophila melanogaster* revealed the gene expression patterns exhibited during the course of its development (Arbeitman et al. 2002). In this study, 4028 genes were examined at 66 distinct time points spanning the embryonic stage (time points 1–30), the larval stage (time points 31–40), the pupal stage (time points 41–58), and the adulthood stage (time points 59–66) of the organism. Based on that study, the time-varying gene regulation network among a set of 588 genes that are related to the developmental process is constructed in Song, Kolar, and Xing (2009). Figure 1 shows the time-varying gene regulation network at different time points.

The gene regulation network, which represents the regulatory relationships among a set of genes, is often organized as a set of interacting functional modules; each module in the gene regulation network consists of a group of co-regulated genes that collaborate in biological processes to efficiently perform a biological function (Segal et al. 2003).

Identifying and characterizing functional modules in the *Drosophila melanogaster* gene regulation network is an important task in developmental biology. First, it can help us understand how genes in multicellular organisms work synergistically to coordinate gene expressions. Second, finding functional modules can help predict the functionalities of previously uncharacterized genes since genes in the same modules share common functionalities. Currently, approximately 25% of the genes in the *Drosophila melanogaster* are computed genes, that is, genes that have no experimental data to support their roles in biological processes (Misra et al. 2002). Third, identifying functional modules also has important biotechnological applications. For example, when the deletion of a certain function is necessary, one can achieve this by removing the entire functional module (Gulbahce and Lehmann 2008).

One common way to study functional modules in a gene regulation network is to look at a static network, that is, one snapshot of the network or a summarized picture of all regulatory relationships in the study period. A static representation of the gene regulation network may lose important information, since as the *Drosophila melanogaster* progresses through its life cycle, its gene regulation network goes through extensive topological changes in response to the changing developmental requirements of the organism. For example, in the edge density plot in Figure 2, the maximum edge density occurs at $t = 41$, with 2061 edges connecting 588 genes, and the lowest edge density occurs at $t = 58$, with 1712 edges connecting 588 genes. The edge

CONTACT Jingfei Zhang ✉ ezhang@bus.miami.edu Department of Management Science, School of Business Administration, University of Miami, Miami, FL 33146, USA.

Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/r/JASA.
Supplementary materials for this article are available online. Please go to www.tandfonline.com/r/JASA.

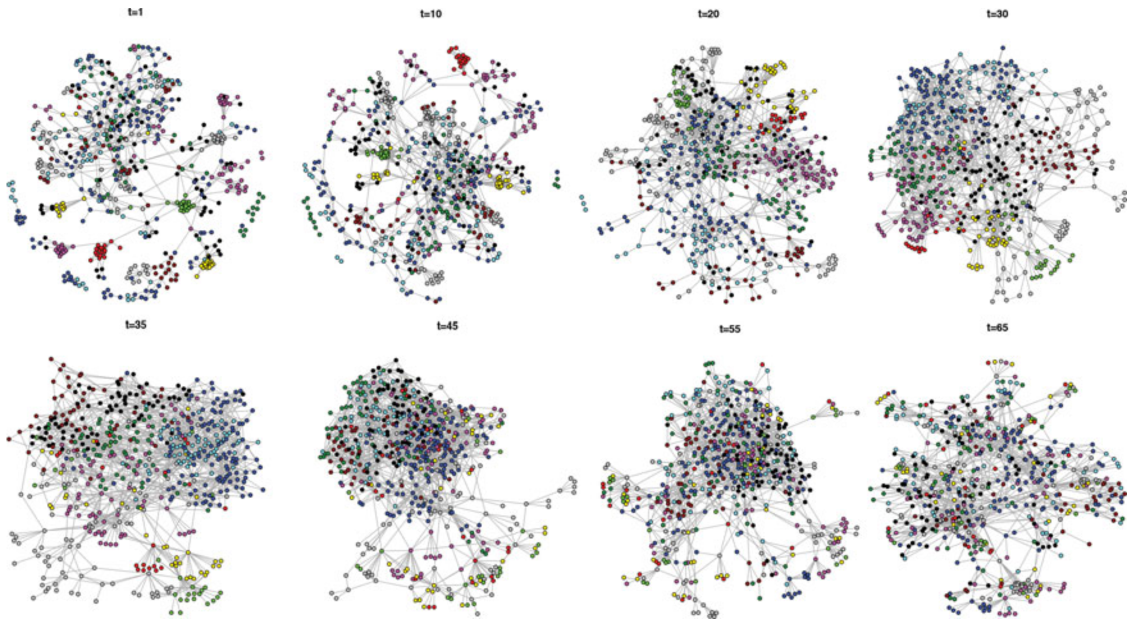

Figure 1. Plots of the *Drosophila melanogaster* gene regulation network at times $t = 1, t = 10, t = 20, t = 30, t = 35, t = 45, t = 55$, and $t = 65$. Nodes of the same color belong to the same group (see Section 4).

density over time shows a wave structure, with two peaks occurring during the mid-embryonic stage and the early pupal stage.

In addition to the network edge density, the network topology also varies notably over time. In Figure 1, we can see that most genes interact selectively with others in the early embryonic stage (time points 1, 10), resulting in a fragmented network with a few densely connected node clusters. During the late pupal stage (time point 55), the genes actively interact with each other, and the network is no longer fragmented. Such important information is not well preserved in a static representation of the time-varying network.

Furthermore, genes within different functional modules are found to be expressed at different times (Arbeitman et al. 2002). For example, cell-cycle genes tend to have high expression levels during the early embryonic stage when cell division is rapid, but very few cell-cycle genes are expressed at high levels after this period. On the other hand, most metabolic genes experience the highest levels of expression only shortly before and during the larval and adult stages. These findings suggest that a static representation of the network may overlook important gene interactions. A preferable approach to identify functional modules is to use information provided by the time-varying gene regulation network throughout its entire time course.

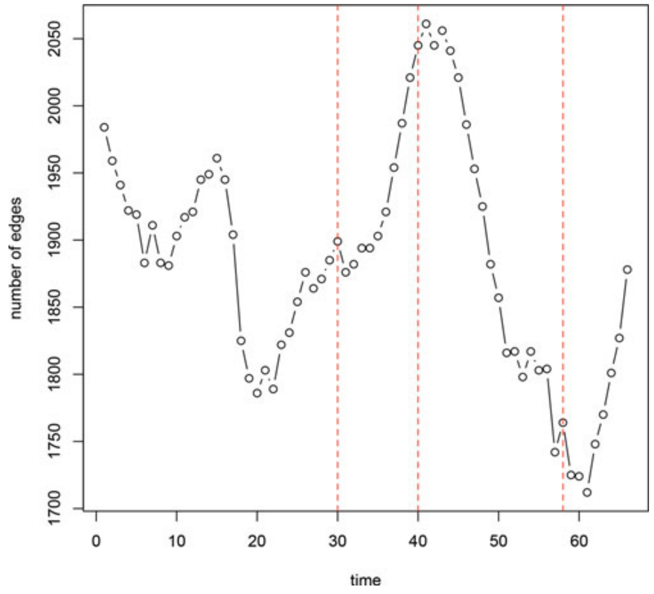

Figure 2. Plot of number of edges at 66 time points. The vertical lines mark the different stages in the development of *Drosophila melanogaster* (from left to right: embryonic, larval, pupal, adulthood).

To identify the functional modules from a time-varying gene regulation network, we consider the following network community structure. First, each node (or gene) is assigned to one community (or functional module), and the assignments do not vary with time; this assumption is consistent with the findings that most genes belong to only one functional module (De Bivort, Huang, and Bar-Yam 2007; Zhao, Schriefer, and Stormo 2007). Second, the nodes within a community have more interactions among themselves than with nodes in other communities; this is because the genes within a functional module are co-regulated and have highly coordinated interactions in response to the developmental requirements (Segal et al. 2003). Third, we assume that the interactions within and between communities are dynamic and vary with time; this corresponds to the findings that genes within different modules are expressed at different times (Arbeitman et al. 2002). We subsequently refer to this type of structure as the *common community structure* in a time-varying network.

For the *Drosophila melanogaster* gene regulation network, by considering the common community structure in a time-varying network, we can better identify and characterize the functional modules. First, as we can observe from Figure 1, the gene interactions are highly variable over time. When identifying functional modules, compared to using only one snapshot of the network, using networks at all time points yields more reliable results since it does not leave out important gene interactions. Second, in instances where a functional module is only active for a short period of time, the strong gene interactions in the short interval may be overlooked in a summarized network that aggregates all interactions in the whole time interval. By considering the time-varying gene interactions when inferring common functional modules, we can better identify the modules that are only active for a short period of time. Third, in addition to helping us understand how the genes in the modules coordinate gene expressions, the identified common functional modules can provide valuable insights into the time-varying expression of each module and its relation to the development of the *Drosophila melanogaster*.

Many methods for community structure detection have been proposed in recent years. See Fortunato (2010) for a review. One major class of methods involves maximizing some partition quality function over all possible partitions of the network (Flake, Lawrence, and Giles 2000; Shi and Malik 2000; Newman and Girvan 2004). Another major class of techniques is model-based approaches, that is, fitting probabilistic models to networks with K communities (Nowicki and Snijders 2001; Handcock, Raftery, and Tantrum 2007; Airoldi et al. 2008; Bickel and Chen 2009; Jin 2015). Recently, Bickel and Sarkar (2016) proposed a hypothesis testing approach that determines the number of communities K. However, these approaches can only be applied to a single static network. Several recent works investigated the problem of community detection in dynamic networks (Mucha et al. 2010; Bansal, Showmich, and Paymal 2011; Bassett et al. 2013; Nguyen et al. 2014). These methods focus on community structures that vary with time and cannot be applied to infer common functional modules that are expressed at different times in the time course of the gene regulation network.

In this work, we address an understudied but important issue in network community study: inferring common functional modules in a time-varying network. The contributions of our work are threefold. First, we propose a statistical framework for modularity-based common module detection in a time-varying network. Second, we describe both a significance test and a robustness test for an identified community structure. Third, we show the consistency property of the proposed method under a temporal stochastic block model framework. Although we focus on the *Drosophila melanogaster* gene regulation network in our work, our method is general and can be applied to other time-varying networks as well.

The remainder of the article is organized as follows. Section 2 proposes the modularity-based method for finding common communities in a time-varying network. A new significance test is introduced to test the statistical significance of the identified community structure. Moreover, we describe a robustness test for testing the robustness of the community detection results under small perturbations of the network. Section 3 demonstrates the effectiveness of our proposed method through simulation studies. Section 4 discusses our community detection findings and the enrichment analysis. Section 5 discusses the consistency property of the proposed method under a temporal stochastic block model framework. Section 6 provides discussion and some concluding remarks.

2. Statistical Methods for Finding Common Modules

2.1. Notation

Consider a time-varying network $\mathcal{G} = (G(t), t \in \mathcal{T})$, where $G(t) = (V, \mathcal{E}(t))$ is the network at time t. The node set V contains n nodes, that is, $V = \{v_1, \ldots, v_n\}$, and the edge set $\mathcal{E}(t)$ contains the list of edges at time t. The network $G(t)$ can be uniquely represented by its adjacency matrix $A(t)$, where $A_{ij}(t) = 1$ if there is an edge between node i and node j at time t, and $A_{ij}(t) = 0$ otherwise. In this article, we focus on simple graphs (undirected graphs with no self-loops or multiple edges). However, our method can be generalized to directed networks (see discussion in Section 6). Throughout this article, we will use the terms graph and network interchangeably.

For simple graphs, the adjacency matrix $A(t)$ is a symmetric 0-1 matrix with a zero diagonal. Let $\mathcal{D} = (\boldsymbol{d}(t), t \in \mathcal{T})$, where $\boldsymbol{d}(t) = (d_1(t), \ldots, d_n(t))$ and $d_i(t)$ is the number of connections incident to node i at time t. At time t, the vector of the column (row) sums of $A(t)$ equals the degree sequence $\boldsymbol{d}(t)$. Finally, let $m(t) = \sum_{i<j} A_{ij}(t)$ denote the total number of edges in $G(t)$.

2.2. Modularity Function for Time-Varying Networks

Newman and Girvan (2004) proposed the modularity function to measure the relative strength of the division of a graph into communities. For a given graph $G(V, E)$ with n nodes, let $\boldsymbol{e} = (e_1, \ldots, e_n)$ denote the community assignment, where $e_i \in \{1, \ldots, K\}$ specifies which community node i belongs to. The modularity function $Q(\boldsymbol{e}, G)$ is defined as

$$Q(\boldsymbol{e}, G) = \frac{1}{2m} \sum_{i,j} [A_{ij} - E(A_{ij})] \delta(e_i, e_j), \tag{1}$$

where $m = \sum_{i<j} A_{ij}$, and $\delta(e_i, e_j) = 1$ if $e_i = e_j$, and $\delta(e_i, e_j) = 0$ otherwise. Here, the expectation $E(A_{ij})$ is calculated under some null model that describes static networks with no community structure.

The modularity function measures the difference between the observed number of intracommunity edges and the expected number of intracommunity edges under the null model. We use the term intracommunity edges to denote edges whose two end nodes are within the same community. If the observed number of intracommunity edges in the network is close to the expected number, the modularity function Q is close to 0. When Q approaches 1, the observed number of intracommunity edges is much higher than the expected number, indicating a strong community structure.

In the Newman–Girvan modularity, the null model is specified to be the *configuration model* (Bender and Canfield 1978; Bollobás 1980). In a configuration model with degree sequence $(d_1, \ldots, d_n)$, each node i is assigned with d_i half-edges, $i = 1, \ldots, n$. To generate a graph from the configuration model, a random matching is conducted on the $2m$ half-edges. Under a configuration model, it can be shown that

$$P(A_{ij} = 1) \approx \frac{d_i d_j}{2m}.$$

In the Newman–Girvan modularity, the degree sequence $(d_1, \dots, d_n)$ in the configuration model is fixed at the observed degree sequence from G. It is worth mentioning that the configuration model generates multigraphs (graphs that allow self-loops and multiple edges) and does not have a uniform distribution over all multigraphs with the given degree sequence (Kranakis 2013).

Zhang and Chen (2016) proposed a null model that assumes a uniform distribution over all simple graphs with a fixed degree sequence, that is,

$$p(G) = \frac{1}{|\Sigma_{\boldsymbol{d}}|}, \quad \text{for } G \in \Sigma_{\boldsymbol{d}},$$

where $\Sigma_{\boldsymbol{d}}$ is the set of all simple graphs with degree sequence $\boldsymbol{d}$. Under such a null model, $E(A_{ij})$ is uniformly

$$\frac{d_i d_j}{2m} + o(1), \tag{2}$$

when $\max_i d_i = o(m^{1/4})$ as $m \to \infty$.

To define the modularity function for a time-varying network $\mathcal{G} = (G(t), t \in \mathcal{T})$, we need to formulate the null model. First, the null model should describe random networks with no community structure. Second, the networks in the null space should share basic structural properties with $\mathcal{G}$ (Newman 2006; Zhang and Chen 2016). For the null model in the modularity function for a time-varying network, we propose to preserve the observed time-varying degree sequence. Often, the edge distribution among the nodes in real-world networks displays high global inhomogeneity, a few nodes with high degrees and many nodes with low degrees, and local inhomogeneity, a high concentration of edges within certain groups of nodes and a low concentration of edges between these groups (Fortunato 2010). To study the local inhomogeneity (or community structure), it is therefore desirable to preserve the observed degree sequence (Zhang and Chen 2016). We fix the degree sequence of the time-varying graphs from the null model at the observed degree sequence $(\boldsymbol{d}(t), t \in \mathcal{T})$.

The sample space in our null model is defined as

$$\boldsymbol{\Sigma}_{\mathcal{D}} = \{(g(t), t \in \mathcal{T}) : g(t) \text{ is a simple graph with} \\ \text{degree sequence } \boldsymbol{d}(t),\ t \in \mathcal{T}\}.$$

For a time-varying network $\boldsymbol{g} = (g(t), t \in \mathcal{T})$ from the sample space $\boldsymbol{\Sigma}_{\mathcal{D}}$, the null distribution is defined as

$$p(\boldsymbol{g}) = \prod_{t \in \mathcal{T}} \frac{1}{|\Sigma_{\boldsymbol{d}(t)}|}, \tag{3}$$

where $\Sigma_{\boldsymbol{d}(t)}$ is the set of all simple graphs with degree sequence $\boldsymbol{d}(t)$. Under the null model, there is no preference for any particular graph configuration, and every time-varying network in the null space $\boldsymbol{\Sigma}_{\mathcal{D}}$ is equally likely to occur.

Under the proposed null model, we have

$$P(A_{ij}(t) = 1) = \frac{|\Sigma_{\boldsymbol{d}(t)|A_{ij}(t)=1}|}{|\Sigma_{\boldsymbol{d}(t)}|},$$

where $|\Sigma_{\boldsymbol{d}(t)|A_{ij}(t)=1}|$ is the total number of simple graphs with degree sequence $\boldsymbol{d}(t)$ and $A_{ij}(t) = 1$. Following the results in (2), we have

$$E(A_{ij}(t)) \approx \frac{d_i(t) d_j(t)}{2m(t)},$$

and $m(t) = \sum_{i<j} A_{ij}(t)$. Note that the proposed null model is time-varying and it captures the dynamic nature of the observed network.

At any time $t \in \mathcal{T}$, we define the modularity matrix $M(t)$ for $G(t)$ as

$$M(t) = A(t) - E(A(t)).$$

The modularity matrix $M(t)$ measures the "distance" between the observed network $G(t)$ and the expected network under the null model at time t. For the time-varying network $\mathcal{G} = (G(t), t \in \mathcal{T})$, the modularity matrix $\mathcal{M}$ is defined as

$$\mathcal{M} = \int_{t \in \mathcal{T}} M(t).$$

Let $\mathbf{e} = (e_1, \dots, e_n)$ denote the community assignment, where $e_i \in \{1, \dots, K\}$ is the community that node i belongs to. The modularity function $\mathcal{Q}(\boldsymbol{e}, \mathcal{G})$ for a time-varying network is defined as

$$\mathcal{Q}(\boldsymbol{e}, \mathcal{G}) = \frac{1}{2\bar{m}} \sum_{i,j} \mathcal{M}_{ij} \delta(e_i, e_j), \tag{4}$$

where $\bar{m}$ is defined as

$$\bar{m} = \int_{t \in \mathcal{T}} m(t).$$

It is easy to see that the modularity function $\mathcal{Q}(\boldsymbol{e}, \mathcal{G})$ assumes values in [−1,1]. When the observed number of intracommunity edges is much greater than the expected number of intracommunity edges, $\mathcal{Q}$ approaches 1, indicating a strong community structure in $\mathcal{G}$. Hence, the communities in $\mathcal{G}$ are identified by finding the $\boldsymbol{e}$ that maximizes the modularity function $\mathcal{Q}(\boldsymbol{e}, \mathcal{G})$. In Section 5, we show that the $\boldsymbol{e}$ that maximizes the proposed modularity in (4) is consistent under a temporal stochastic block model framework.

In the simple case when $\mathcal{T} = \{t_1, t_2, \dots, t_S\}$, where S is the total number of observations or snapshots, we have

$$\mathcal{M} = \sum_{l=1}^{S} M(t_l),$$

and

$$\bar{m} = \sum_{l=1}^{S} m(t_l).$$

Here, the modularity function $\mathcal{Q}(\boldsymbol{e}, \mathcal{G})$ can be considered as an averaged version of the modularity in each graph $G(t)$, $t = t_1, \dots, t_S$, that is,

$$\mathcal{Q}(\boldsymbol{e}, \mathcal{G}) = \frac{\sum_{l=1}^{S} m(t_l) Q(\boldsymbol{e}, G(t_l))}{\sum_{t=1}^{S} m(t_l)}.$$

2.3. Modularity Maximization

Maximizing the modularity function in (4) is a very difficult problem, since the number of communities K is generally unknown. Brandes et al. (2008) showed that finding a partition that maximizes the modularity function for a given graph is NP-complete. Existing heuristic approaches for maximizing the modularity function come from various fields, including computer science, physics, sociology, and statistics. Some are fast techniques that can be efficiently applied to large graphs (Clauset, Newman, and Moore 2004; Wakita and Tsurumi 2007; Blondel et al. 2008), while some are more accurate but limited to graphs of moderate size (Guimera, Sales-Pardo, and Amaral 2004; Massen and Doye 2005).

The two objective functions in (4) and (1) are in similar forms. Once we have calculated $\bar{m}$ and $\mathcal{M}$, existing modularity maximization approaches can be used to find the $\mathbf{e} = (e_1, \dots, e_n)$ that maximizes the modularity function in (4).

In this article, we adopt the Louvain maximization method proposed by Blondel et al. (2008). In the Louvain method, the optimization has two phases that are repeated iteratively. First, we assign each node in the network to its own community. Next, for each node i, we calculate the change in modularity when we assign node i to the community of its neighbor j. The change in modularity that results from moving an isolated node i into community k can be easily computed by

$$\Delta Q = \left[\frac{O_{kk} + 2w_{ik}}{2m} - \left(\frac{O_k + w_i}{2m}\right)^2\right] - \left[\frac{O_{kk}}{2m} - \left(\frac{O_k}{2m}\right)^2 - \frac{w_i}{2m}\right],$$

where O_{kk} is the sum of weighted edges in community k, O_k is the sum of the weighted edges that are linked to nodes in community k, w_{ik} is the sum of the weighted edges that are linked from node i to community k, and w_i is the sum of weighted edges linked to node i. Note that a similar expression can be formulated to calculate the change in the modularity when we remove i from its current community. Therefore, we can calculate the change in the modularity by removing i from its current community and adding it to community k. Once this value has been obtained for every community that node i is linked to, i is moved to the community that results in the largest increase in modularity. If no increase is possible, then i remains in its original community.

In the second phase, the algorithm aggregates nodes in the same community and builds a new network whose nodes are the communities from the previous step. The weights of the edges between the new nodes are given by the summing of the weights of the edges connecting the two corresponding communities. Edges connecting nodes within the same community lead to self-loops in the new network. These steps are repeated iteratively until the modularity reaches its maximum. The algorithm can be summarized as

Algorithm. Take the modularity matrix M as input:

1. Assign each node to its own community.
2. For each node i, place i into the neighboring community (possibly its own) that leads to the highest modularity gain.
3. Repeat Step 2 until no nodes can be moved.
4. If the new modularity is higher than the initial modularity, then build a new network whose nodes are the current communities and return to Step 1. If not, output the community assignment and the modularity value.

The Louvain method has been used with success for networks of many different types and for sizes of up to 100 million nodes and billions of links. The analysis of a typical network with 2 million nodes takes 2 min on a standard PC (Blondel 2011). Fortunato (2010) noted that the modularity maxima found by the Louvain method often compare favorably with those found by Clauset, Newman, and Moore (2004) and Wakita and Tsurumi (2007).

It is important to note that our approach is not tied to the Louvain method. In fact, most existing modularity maximization methods can be used, without modification, under our framework. However, in practice, we find that the Louvain method yields better modularity maxima than the other methods and is computationally more efficient.

It is worth mentioning that the Louvain method may not arrive at the same result in successive runs since the nodes in the network are ordered randomly in the algorithm. Moreover, in the aggregation step, each node is assigned to the community that leads to the largest modularity increase; if there are several communities that all lead to the largest increase, one community is randomly selected. Hence, it is possible the community assignment obtained from a single run of the Louvain method is a local maximum. In our analysis, we apply the Louvain method κ times to find the maxima of the modularity function. In general, κ should increase with the size of the network. In our analysis of the *Drosophila melanogaster* gene regulation network, the number of applications κ is set as $\kappa = 1000$. We do not observe notable improvements in the maximized modularity function for $\kappa > 1000$. However, other networks of comparable size may benefit if larger values of κ are selected.

Despite the large volume of works that have been proposed for detecting communities in networks, few have been proposed to assess the statistical significance or to quantify the uncertainty associated with identified community structures. In Section 2.4 and Section 2.5, we describe, respectively, a significance test and a robustness test to assess our community findings.

2.4. Significance Test

Newman and Girvan (2004) suggested that networks with a strong community structure typically have a maximum modularity value $\max_{\boldsymbol{e}} Q_{\text{NG}}(\boldsymbol{e}, G) \in [0.3, 0.7]$. This criterion has been widely used as a general rule of thumb in subsequent works on modularity-based community detection. However, a large Newman–Girvan modularity value does not necessarily indicate a strong community structure.

For instance, random graphs from the Erdös-Rényi model (every pair of nodes is equally likely to have an edge) can have partitions with large modularity values (Guimera, Sales-Pardo, and Amaral 2004; Reichardt and Bornholdt 2006). Such graphs should not have a community structure, since the probability of having an edge between any pair of nodes is the same and every node is treated equally. Based on the definition of modularity, a network should only be considered to have a community structure if its maximized modularity value is significantly larger than the maximized modularity value of the graphs from the null model. In this section, we propose a hypothesis testing procedure to test whether an identified common community structure is significant.

Given the time-varying graph $\mathcal{G}$ and a community assignment $\boldsymbol{e}^* = (e_1^*, \dots, e_n^*)$, the statistical significance of the partition is

$$P[\mathcal{Q}(\boldsymbol{e}^*, \mathcal{G}) \leq \max_{\boldsymbol{e}} \mathcal{Q}(\boldsymbol{e}, \boldsymbol{g})], \tag{5}$$

where $\boldsymbol{g}$ follows the null distribution in (3). A small p-value indicates that the partition $\boldsymbol{e}^*$ obtained from $\mathcal{G}$ is significant, since graphs under the null model are unlikely to have a maximized modularity as large as $\mathcal{Q}(\boldsymbol{e}^*, \mathcal{G})$.

Sampling $\boldsymbol{g} = (g(t), t \in \mathcal{T})$ from $\boldsymbol{\Sigma}_{\mathcal{D}}$ is achieved by sampling $g(t)$ from $\Sigma_{\boldsymbol{d}(t)}, t \in \mathcal{T}$. One way to sample from $\Sigma_{\boldsymbol{d}(t)}$ is to use the configuration model. However, the graphs generated from the configuration model may contain loops and multiple edges. Discarding all multigraphs could lead to a high invalid sample ratio, since the probability of having loops and multiple edges quickly increases when the degrees increase in a configuration model (Chung and Lu 2002).

Alternatively, we may efficiently sample $g(t)$ from $\Sigma_{\boldsymbol{d}(t)}$ using the following Markov chain Monte Carlo (MCMC) algorithm. Denote an edge between node i and node j by $\{i, j\}$. At each step of the MCMC algorithm, randomly choose two edges $\{x, y\}$ and $\{u, v\}$ from the current graph g such that x, y, u, and v are four distinct nodes. If there are no edges between x and u or y and v, the Markov chain moves to a new state g' by replacing the edges $\{x, y\}$ and $\{u, v\}$ in the current graph g by two new edges $\{x, u\}$ and$\{y, v\}$; otherwise, the Markov chain stays at the current graph g. We may take a sample after every N_0 MCMC steps to reduce the dependence between samples.

To estimate the p-value in (5), we generate samples $\boldsymbol{g}_1, \dots, \boldsymbol{g}_S$ from the null model in (3) using the aforementioned MCMC algorithm. We then approximate (5) by

$$\frac{1}{S}\sum_{i=1}^{S} I\big(\mathcal{Q}(\boldsymbol{e}^*, \mathcal{G}) \leq \max_{\boldsymbol{e}} \mathcal{Q}(\boldsymbol{e}, \boldsymbol{g}_i)\big).$$

2.5. Robustness Test

In addition to assessing the significance, it is also important that we consider the robustness of an identified community structure against small perturbations of the network. If a small change in the topology of the network can completely change the outcome of our community findings, we cannot fully trust the identified community structure. In this section, we describe a robustness test for an identified community structure.

We consider two types of network perturbations: adding/removing edges and rewiring edges. Note that adding and removing edges change the degree distribution, whereas rewiring edges preserves the degree distribution. In both types of perturbation, the total number of edges remains unchanged. To perform a perturbation at level γ for a time-varying network $\boldsymbol{g} = (g(t), t \in \{t_1, \dots, t_S\})$, we use the following perturbation scheme.

```
Algorithm (Perturbation at level $\gamma$).
Start from time $l = 1$.
  1. Set $s = 0$.
  2. Randomly pick an edge $\{x, y\}$ from the current graph $g(t_l)$,
     (a) with probability 0.5, we randomly choose an edge $\{u, v\}$ from $g(t_l)$, such that $x, y, u$, and $v$ are four distinct nodes and there are no edges between $x$ and $u$ or $y$ and $v$. We then replace $\{x, y\}$ and $\{u, v\}$ in $g(t_l)$ by two new edges, $\{x, u\}$ and $\{y, v\}$. Set $s = s + 2$.
     (b) with probability 0.5, we randomly choose an unconnected pair of nodes $u, v$ from $g(t_l)$. We then delete the edge $\{x, y\}$ and add the edge $\{u, v\}$. Set $s = s + 1$.
  3. If $s/m(t_l) \leq \gamma$, go to step 2. If $s/m(t_l) > \gamma$, set $l = l + 1$ and go to Step 1.
```

Before we can describe the robustness test, we first introduce a numerical measure to quantify the difference between two partitions. In this work, we adopt the normalized mutual information measure (Danon et al. 2005). Consider the community assignment $\{x_i\}$ and $\{y_i\}$, where x_i and y_i indicate the cluster labels of vertex i in partitions $\mathcal{X}$ and $\mathcal{Y}$, respectively. Assume that labels x and y are the observed values of two random variables X and Y. The normalized mutual information (nmi) is measured by

$$\text{nmi}(\mathcal{X}, \mathcal{Y}) = \frac{2I(X, Y)}{H(X) + H(Y)}.$$

Here, $I(X, Y) = H(X) - H(X|Y)$ is the mutual information and $H(X) = -\sum_x P(x)\log P(x)$ is the Shannon entropy of X. The normalized mutual information equals 1 if the two partitions are identical, and its expected value is 0 if the two partitions are independent.

For a graph $\mathcal{G}$ with an identified community structure $\hat{\boldsymbol{c}}$, to see how the community detection output changes as a function of the perturbation parameter γ, we perform the following steps:

Step 1: Use the proposed perturbation scheme to perform a perturbation of $\mathcal{G}$ at level γ. Let $\mathcal{G}_\gamma$ denote the perturbed time-varying network.

Step 2: Find $\hat{\boldsymbol{c}}_{\boldsymbol{\gamma}} = \arg\max \mathcal{Q}(\boldsymbol{e}, \tilde{\boldsymbol{g}}_{\boldsymbol{\gamma}})$.

Step 3: Calculate the normalized mutual information between $\hat{\boldsymbol{c}}$ and $\hat{\boldsymbol{c}}_{\boldsymbol{\gamma}}$.

For each γ, we repeat this procedure N times and report the average of the normalized mutual information measures.

3. Simulation Study

Consider a dynamic network $\mathcal{G} = (G(t), t = t_1, \dots, t_S)$ with node set $V = \{v_1, \dots, v_n\}$ and community label $\boldsymbol{c} = (c_1, \dots, c_n)$, where $c_i \in [1, \dots, K]$. Denote the adjacency matrix of $G(t)$ by $A(t)_{n\times n}$, where $A_{ij}(t) = 1$ if node i and node j are connected in $G(t)$ and $A_{ij}(t) = 0$ otherwise.

We evaluate the performance of the proposed method through simulated time-varying networks and compare it to the performances of the following methods:

- Method 1: infer $\boldsymbol{c}$ from $G(t^*)$, $t^* \in [t_1, \dots, t_S]$, that is, community detection based on a single snapshot of the time-varying network,

- Method 2: infer $\boldsymbol{c}$ from A^* where $A^*_{ij} = \sum_t A_{ij}(t)$, that is, community detection based on a summary multigraph.

To create the multigraph in Method 2, for each pair of nodes i and j, one simply counts the total number of interactions that i and j have in the observation window. To obtain $\hat{\boldsymbol{c}}$ in Method 1, we maximize the modularity function calculated from a single simple (no multiple edges, no self-loops) network (Newman and Girvan 2004). To obtain $\hat{\boldsymbol{c}}$ in Method 2, we maximize the modularity function calculated from a single multigraph (Newman 2004).

Another way to summarize a time-varying network would be to construct $\tilde{A}$ where $\tilde{A}_{ij} = \max_t A_{ij}(t)$, that is, $\tilde{A}_{ij} = 1$ if node i and node j interact at least once in the observation window. We find that this method gives poor community detection results and can occasionally yield fully connected simple graphs as summary graphs. Therefore, we do not include this method in our comparison.

We consider three simulation settings in this section. In Simulation 1.1, we consider networks independently sampled from a stochastic block model with a time-homogenous probability matrix. In Simulation 1.2, we consider networks independently sampled from a stochastic block model with a time-varying probability matrix. In Simulation 2, we consider temporally correlated samples from a stochastic block model with a time-varying probability matrix.

Simulation Setting 1

- At time t, the edges $A_{ij}(t)$ are independent Bernoulli random variables with

$$A_{ij}(t) \sim \text{Bernoulli}(\theta_{c_i c_j}(t)),$$

where $$\Theta(t) = \begin{pmatrix} \theta_{11}(t) & \cdots & \theta_{1K}(t) \\ \vdots & \ddots & \vdots \\ \theta_{K1}(t) & \cdots & \theta_{KK}(t) \end{pmatrix}$$

is the probability matrix of the stochastic block model at time t.

- Our simulated networks have four communities, and each community has 50 nodes. We have S observations within $t \in [0, 1]$, and all the observations are equally spaced.

Simulation 1.1: $$\Theta(t) = \begin{pmatrix} 0.2 & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.2 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.3 & 0.2 \\ 0.1 & 0.1 & 0.2 & 0.3 \end{pmatrix}.$$

In Simulation 1.1, the probability matrix $\Theta(t)$ remains constant for $t \in [0, 1]$. For each S, we simulate 100 time-varying networks $\mathcal{G}_1, \ldots, \mathcal{G}_{100}$, where $\mathcal{G}_i = (G_i(t), t = t_1, \ldots, t_S)$. For each time-varying network $\mathcal{G}_i$, we find $\hat{\boldsymbol{c}}_i$ and calculate the normalized mutual information between the true community membership and $\hat{\boldsymbol{c}}_i$, $i = 1, \ldots, 100$. This procedure is performed using the proposed method, Method 1 (single snapshot taken at $t = 0.5$), and Method 2. The results are summarized in the left plot in Figure 3.

From Figure 3, we can see that Method 1, which only looks at one snapshot of the network, has poor performance, with an average NMI of 0.282. This is likely due to the small differences between the intracommunity connection probabilities and the intercommunity connection probabilities. Our proposed method outperforms Method 2 for all plotted values of S. Method 2 does not have satisfactory performance, as the communities are difficult to distinguish in a multigraph created by adding all networks together. Note that our proposed method has good performance even with as few as five samples ($S = 5$) from the stochastic block model. It is worth mentioning that the proposed method, Method 1 and Method 2 become equivalent when only one network is sampled from the model.

Simulation 1.2: $$\Theta(t) = \begin{pmatrix} \theta_{11}(t) & 0.1 & 0.1 & 0.1 \\ 0.1 & \theta_{22}(t) & 0.1 & 0.1 \\ 0.1 & 0.1 & \theta_{33}(t) & 0.2 \\ 0.1 & 0.1 & 0.2 & \theta_{44}(t) \end{pmatrix},$$

with $\theta_{11}(t)$, $\theta_{22}(t)$, $\theta_{33}(t)$, and $\theta_{44}(t)$ plotted in Figure 4. In this simulation setting, the probability matrix of the stochastic block model is time-varying. At time $t = 0$, community 1 and community 3 are active, while community 2 and community 4 are inactive. At time $t = 0.5$, communities 2, 3, and 4 are equally active, while community 1 becomes inactive.

For each S, we simulate 100 time-varying networks $\mathcal{G}_1, \ldots, \mathcal{G}_{100}$, where $\mathcal{G}_i = (G_i(t), t = t_1, \ldots, t_S)$. For each time-varying network $\mathcal{G}_i$, we find $\hat{\boldsymbol{c}}_i$ and calculate the normalized mutual information between the true community membership and $\hat{\boldsymbol{c}}_i$, $i = 1, \ldots, 100$. This procedure is performed using the proposed method, Method 1 (single snapshot taken at $t = 0.5$)

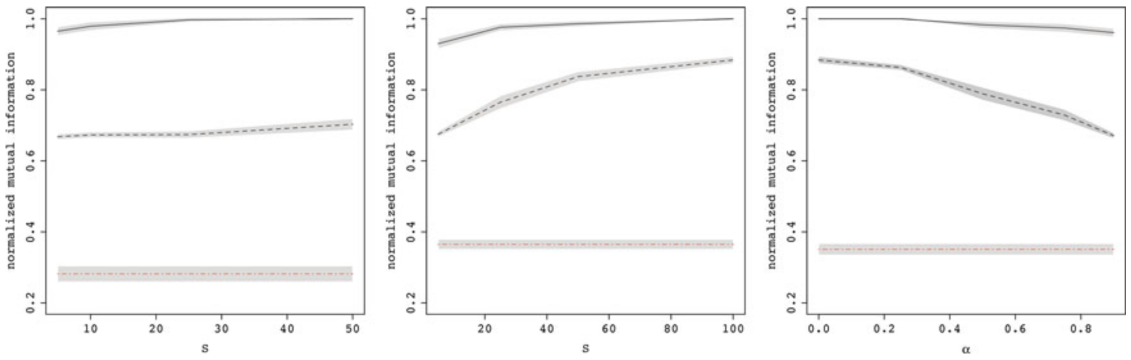

Figure 3. Normalized mutual information between the true community membership and the community membership calculated from the proposed method (black solid line), Method 1(red dashed line) and Method 2 (black dashed line). The gray bands are the 95% confidence intervals for the mean normalized mutual information. Left plot: results from Simulation 1.1; Middle plot: results from Simulation 1.2; Right plot: results from Simulation 2.

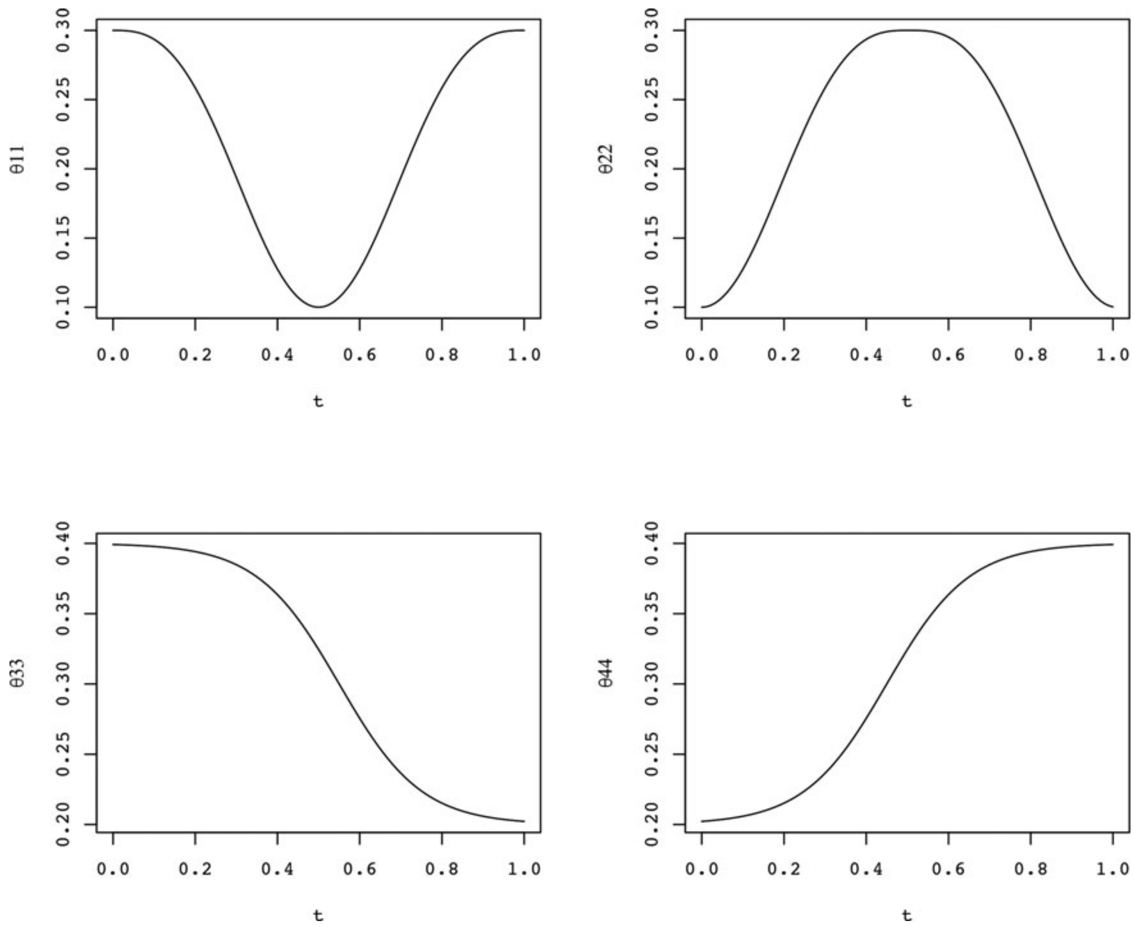

Figure 4. The four time-varying probability functions $\theta_{11}(t)$, $\theta_{22}(t)$, $\theta_{33}(t)$, and $\theta_{44}(t)$.

and Method 2. The results are summarized in the middle plot in Figure 3.

We can see that the proposed method has the best performance out of the three methods. When we only have observations at $t = 0, 0.25, 0.5, 0.75$, and 1, that is, $S = 5$, the proposed method has an average NMI of 0.930. Table 1 summarizes the performance of Method 1 when the snapshot is taken at different times. From the low NMI measures from Method 1 in Table 1, it is clear that one snapshot cannot fully capture the community structure of the underlying model. Method 2 has improved performance when S increases. However, the NMI measures for Method 2 are still below 0.9. This is because for communities that are active for a short period of time, the strong connections are ablated in an aggregated picture.

Simulation Setting 2

In Simulation 2, we simulate networks using the following model:

- At time t_l, edge $A_{ij}(t_l)$ is a Bernoulli random variable with

$$A_{ij}(t_l) = uA_{ij}(t_{l-1}) + (1-u)\upsilon,$$

where $u \sim^{\text{iid}}$ Bernoulli(α) and

$$\upsilon \sim^{\text{ind}} \text{Bernoulli}\left(\frac{\theta_{c_ic_j}(t_l) - \alpha\theta_{c_ic_j}(t_{l-1})}{1-\alpha}\right).$$

- $\Theta(t) = \begin{pmatrix} \theta_{11}(t) & \cdots & \theta_{1K}(t) \\ \vdots & \ddots & \vdots \\ \theta_{K1}(t) & \cdots & \theta_{KK}(t) \end{pmatrix}$ is the time-varying probability matrix.
- Our simulated networks have four communities, and each community has 50 nodes. We have $S = 100$ equally spaced observations within $t \in [0, 1]$.

Table 1. The normalized mutual information between the true community membership and community memberships calculated from Method 1.

Snapshot at time t	$t=0$	$t=0.25$	$t=0.5$	$t=0.75$	$t=1$
NMI	0.357	0.423	0.365	0.418	0.373

It can be shown that $A_{ij}(t)$ is a Bernoulli random variable with $P(A_{ij}(t) = 1) = \theta_{c_ic_j}(t)$. Hence, the marginal distribution of $A_{ij}(t)$ is the same as that in Simulation 1.2. However, $A_{ij}(t_l)$ and $A_{ij}(t_{l-1})$ are correlated with

$$\text{corr}(A_{ij}(t_l), A_{ij}(t_{l-1})) = \alpha\sqrt{\frac{\theta_{c_ic_j}(t_{l-1})(1-\theta_{c_ic_j}(t_{l-1}))}{\theta_{c_ic_j}(t_l)(1-\theta_{c_ic_j}(t_l))}}.$$

More discussion on this model and its theoretical properties can be found in Section 5.

In this simulation, we use the time-varying $\Theta(t)$ from Simulation 1.2. Given $\Theta(t)$, larger values of α result in a higher correlation between two adjacent time points. For each α, we simulate 100 time-varying networks $\mathcal{G}_1, \ldots, \mathcal{G}_{100}$, where $\mathcal{G}_i = (G_i(t), t = t_1, \ldots, t_S)$. For each time-varying network $\mathcal{G}_i$, we find $\hat{\boldsymbol{c}}_i$ and calculate the normalized mutual information between the true community membership and $\hat{\boldsymbol{c}}_i$, $i = 1, \ldots, 100$. This procedure is performed using the proposed method, Method 1 (single snapshot taken at $t = 0$) and Method 2. The results are summarized in the right plot in Figure 3.

The proposed method outperforms Method 1 and Method 2. When $\alpha = 0$, $A_{ij}(t)$ is uncorrelated with the past observations, and the model is equivalent to the model used in Simulation 1.2.

4. Data Analysis

We implement the proposed method to identify common functional modules in the time-varying *Drosophila melanogaster* gene regulation network described in Section 1. Using the proposed method, we identify 10 groups among the 588 nodes in $\mathcal{G}$, with a maximized modularity value $\mathcal{Q} = 0.4318$. A significance test and a robustness test of this community finding are performed in Sections 4.1 and 4.2, respectively.

Figure 1 is a graphical representation of our community findings. To better view the common community structure, we plot the adjacency matrices of the 10 networks in Figure 5. In the adjacency matrix plot, cell (i, j) is nonempty if there is an edge between node i and node j. The nonzero cells in the adjacency matrices have a block diagonal structure, which indicates frequent interactions within blocks and sparse interactions between blocks. Moreover, we can see that the number of interactions within each cluster waxes and wanes over time, and different clusters dissolve and reappear according to different schedules.

For example, the genes from Group 1 reveal increased activity during the larval stage (time points 31–40) and experience a peak near the end of the larval stage. The genes from Group 5 peak in their activity at the beginning of the embryonic stage (time points 1–10) and remain active throughout the embryonic stage. These genes become relatively inactive during the rest of the life cycle. These findings suggest that the activity cycles of different gene functional modules can follow distinct temporal patterns. Since the genes we investigated are all related to the developmental process, the modular structure weakens during the adult stage (time points 59–66). Refer to the supplementary file for the gene membership of each group.

To identify significantly related biological processes for each of the 10 groups, we perform a gene ontology enrichment analysis using the Web-based Gene Set Analysis Toolkit (WebGestalt) developed by Wang et al. (2013). The gene ontology enrichment analysis identifies significantly enriched annotated biological processes within a given set of genes. Such an analysis links the gene set to known phenotypes and helps elucidate the roles of these genes in states of health and disease (Subramanian et al. 2005). Moreover, the analysis can suggest potential phenotypes for newly identified gene variants and proteins associated with the gene set.

Figure 6 displays the heat map for the biological processes that are significantly enriched within the 10 groups of genes. A dark-colored cell at the ith row and the jth column in the heat map indicates that the ith biological process is strongly enriched in the jth gene group. Refer to the supplementary file for more details on the enrichment analysis. We can see that each group is enriched with several biological processes; the enrichment is especially strong for Group 4. Some processes are enriched in all groups (processes at the upper part of the heat map), such as *anatomical structure morphogenesis*, *system development*, *organ development*, and *cell differentiation*. These are general processes that include most of the biological processes involved in the developmental process as subtypes (or child processes). The enrichment of these processes is expected, since the genes we studied are all involved in the development of the *Drosophila melanogaster*.

We observe that more specific processes are enriched in each group. For example, Group 3 is enriched with genes involved in the *circulatory system development*; this biological process contains child processes such as *heart development* and *cardiovascular system development*. Notably, two highly important genes in the heart development, “robo” (i.e., roundabout) and

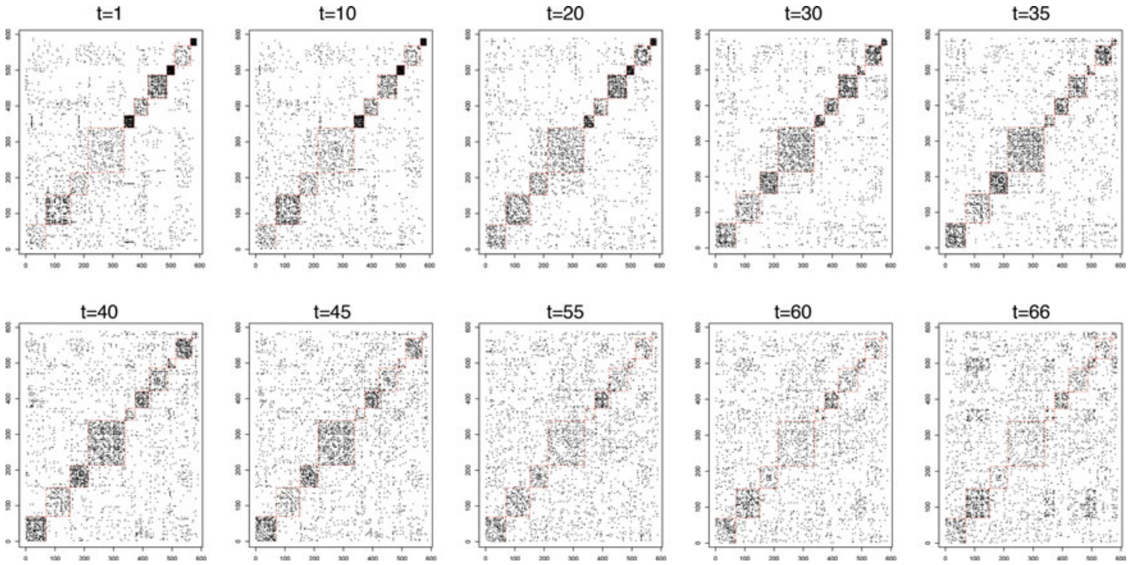

Figure 5. Adjacency matrix of the *Drosophila melanogaster* gene regulation network at times $t = 1$, $t = 10$, $t = 20$, $t = 30$, $t = 35$, $t = 40$, $t = 45$, $t = 55$, $t = 60$, and $t = 66$, with genes ordered according to the common community structure. The red dashed lines mark the boundaries of the 10 groups. The 10 groups are organized from the bottom left to the top right (Group 1 on bottom left; Group 10 on top right).

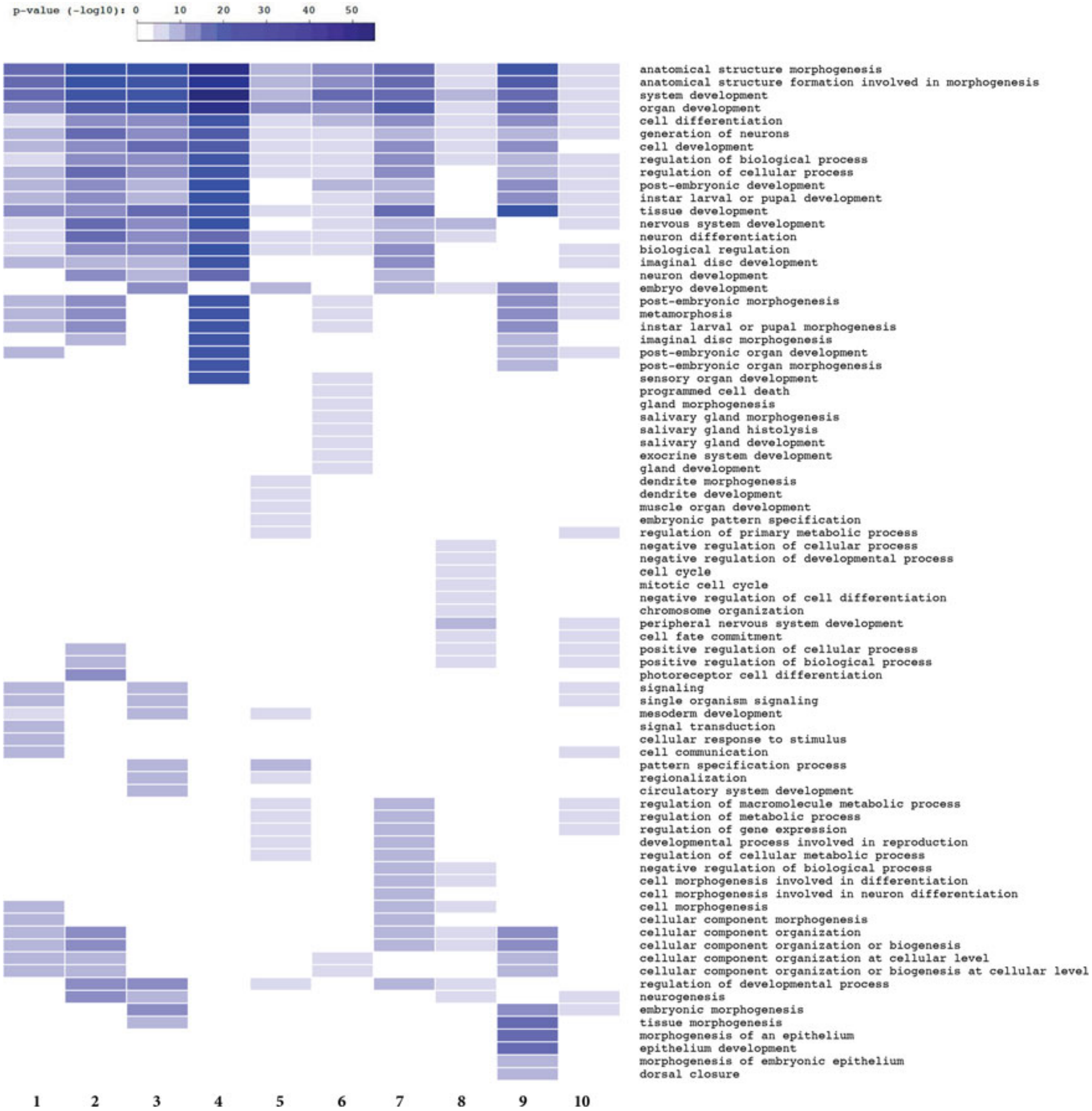

Figure 6. Heat map of enrichment in biological processes for Groups 1–10. The upper left inset shows the color key for the enriched processes (color intensity is proportional to the $-\log_{10}$ of *p*-values).

"eve" (i.e., even skipped), are members of Group 3. Group 4 is significantly enriched with genes that participate in the *sensory organ development*; this biological process is parent to child processes such as *sensory organ morphogenesis*, *eye development,* and *sex comb development*. Two critical genes in the eye development, "eya" (i.e., eyes absent) and "rg" (i.e., rugose), are members of Group 4. From Figure 5, we observe that both Groups 3 and 4 are highly active during the late embryonic stage, the larval stage, and the early pupal stage (time points 20–45). Moreover, Group 3 and Group 4 have more interactions during the late embryonic stage and the early larval stage (time points 20–35). These observations suggest that the genes involved in the *circulatory system development* and the *sensory organ development* are most active during the late embryonic stage, the larval stage, and the early pupal stage, and these genes have increased interactions during the late embryonic stage and the early larval stage.

We also find that processes related to *signaling* and *response to stimulus* are enriched in Group 1, while processes related to *neurogenesis* such as the *photoreceptor cell differentiation* are strongly enriched in Group 2. Group 5 is enriched with genes from *muscle organ development* and *dendrite development*. Group 6 is related to *gland development*, especially *salivary gland development*. Several processes related to *regulation of metabolic processes* are enriched in Group 7, such as *regulation of macromolecule metabolic processes* and *regulation of gene expression*. We can see that Group 8 is strongly enriched with *peripheral nervous system development*, and Group 9 is enriched with the *epithelium development*, which includes several child processes such as *morphogenesis of embryonic epithelium* and *dorsal closure*. It is worth mentioning that many *biological regulation* related processes are enriched in Group 10, despite its relatively small size.

Identifying these functional modules can provide valuable insights into their temporal patterns of activation, appearance, and disappearance at different time points during the life cycle of the *Drosophila melanogaster*. Due to the transient nature of gene interactions, certain temporal patterns may only be present

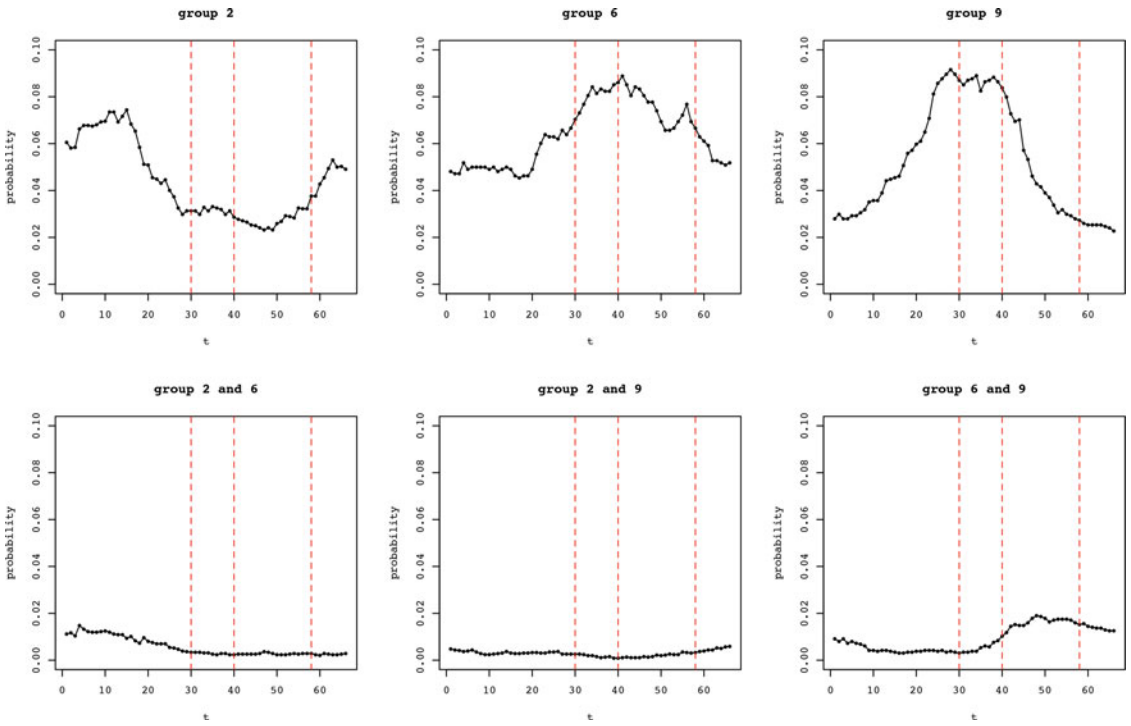

Figure 7. The time-varying intracommunity and intercommunity edge probabilities between Group 2, Group 6, and Group 9. The vertical lines mark the different stages in the development of the *Drosophila melanogaster* (from left to right: embryonic, larval, pupal, adulthood). The top three plots are the intracommunity edge probabilities, and the bottom three plots are the intercommunity edge probabilities.

for short periods of time. These types of patterns can be overlooked if we rely only on a single snapshot of the network for our analysis. In Figure 7, we illustrate the time-varying intracommunity and intercommunity edge probabilities for three randomly selected groups (Group 2, Group 6, and Group 9). The intracommunity edge probability for community k at time t is calculated using

$$\frac{2O_{kk}(t)}{n_k(n_k-1)},$$

where $O_{kk}(t)$ is the number of edges within community k at time t and n_k is the size of community k, $k \in \{1, \dots, K\}$. The intercommunity edge probability between community k and community h at time t is calculated using

$$\frac{O_{kh}(t)}{n_k n_h},$$

where $O_{kh}(t)$ is the number of edges between community k and community h at time t, $k, h \in \{1, \dots, K\}$.

From Figure 7, we see that the activeness of Group 2 has two peaks: one in the mid-embryonic stage and one near the beginning of the adulthood stage. Group 6 displays an increase in activity in the larval and early pupal stages, and Group 9 is highly active during the late embryonic and larval stages. Clearly, these three different groups follow three distinct temporal patterns; this shows that the activeness of the functional modules is highly dynamic in nature. Genes in functional modules act in tandem to achieve a relatively autonomous functionality, and the interactions between functional modules are weak. This can be observed from the bottom three plots of Figure 7, which show that the interactions between the three groups remain at a low level throughout the life cycle of *Drosophila melanogaster*. Previous experimental genetic studies have revealed similar findings. For example, empirical studies have revealed that the cell growth process can be decoupled from the cell cycle process in yeast, suggesting that independent modules control these two processes (Jorgensen et al. 2002).

4.1. Significance Test

In this section, we perform the significance test described in Section 2.4 on the identified community structure. From the null model for the *Drosophila melanogaster* gene regulation network, we generate 100 time-varying networks $\boldsymbol{g}_1, \dots, \boldsymbol{g}_{100}$, where $\boldsymbol{g}_i = (g_i^1, \dots, g_i^{66})$, $i = 1, \dots, 100$. We set the rewiring step parameter N_0 to 2000, which is approximately the total number of edges at each time point. With the thinning in the MCMC algorithm, the 100 samples we obtain are approximately iid. For each time-varying network, we find the maximum modularity $\max_{\boldsymbol{e}} \mathcal{Q}(\boldsymbol{e}, \boldsymbol{g}_i)$, $i = 1, \dots, 100$, and calculate the empirical p-value,

$$\frac{1}{100}\sum_{i=1}^{100} I\big(\mathcal{Q}(\boldsymbol{e}^*, \mathcal{G}) \leq \max_{\boldsymbol{e}} \mathcal{Q}(\boldsymbol{e}, \boldsymbol{g}_i)\big).$$

See Figure 8 for the histogram of $\max_{\boldsymbol{e}} \mathcal{Q}(\boldsymbol{e}, \boldsymbol{g}_i)$, $i = 1, \dots, 100$.

The maximized modularity of the observed network is $\mathcal{Q} = 0.4318$, which is significantly larger than the modularity maxima of the 100 networks sampled from the null model. The p-value is estimated to be zero, and this indicates that the community structure we identified is highly significant. We repeat this procedure 100 times. For all of the 100 runs, the p-values are estimated to be zero.

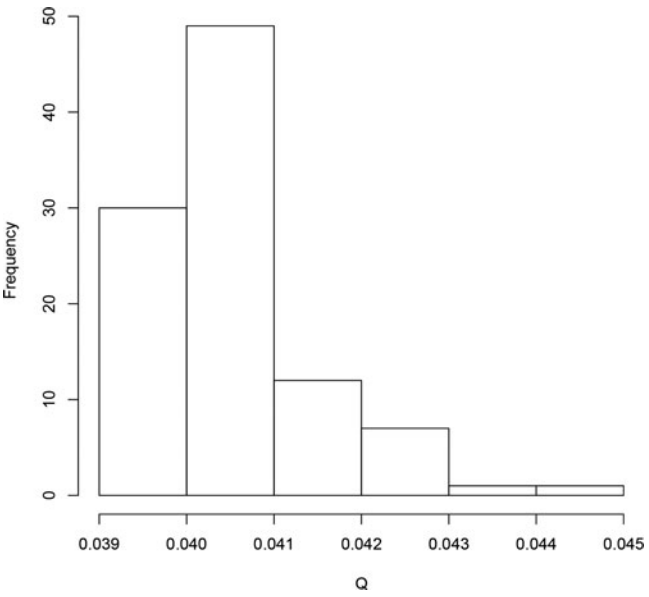

Figure 8. Histogram of the maximized modularity of the 100 randomly sampled time-varying networks from the null model.

4.2. Robustness Test

In this section, we perform the robustness test described in Section 2.5 on the identified community structure. Figure 9 is a plot of the normalized mutual information as a function of the perturbation parameter γ. For each γ, we repeat the perturbation procedure $N = 10$ times and report the average of the normalized mutual information measures. When $\gamma = 0$, the network is unperturbed, and the normalized mutual information between $\boldsymbol{c}$ and $\hat{\boldsymbol{c}}$ is 1, indicating that the two community findings are identical.

In Figure 9, we see that the community structure detected through our method is robust against small perturbations of the network. For example, the average normalized mutual information is close to 0.8 when $\gamma = 0.2$. Roughly speaking, we can say that approximately 20% of the edges must be rewired before 20% of the vertices move to different communities. On the other hand, for a random graph, rewiring 20% of the edges will change approximately 90% of the community assignments.

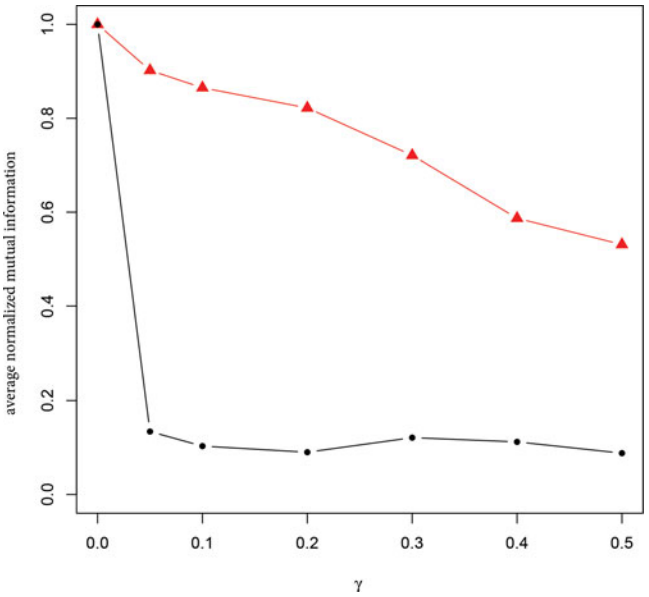

Figure 9. The normalized mutual information as a function of the perturbation parameter γ for the *Drosophila melanogaster* gene regulation network (red triangles), along with results for the corresponding random networks (black circles).

To determine how the normalized mutual information changes as a function of the perturbation parameter γ for random networks, we go through the following steps:

Step 1: Generate a time-varying network $\tilde{\boldsymbol{g}} = \{\tilde{g}^1, \ldots, \tilde{g}^{66}\}$, where $\tilde{g}^t$ is a randomly generated simple graph with degree sequence $\boldsymbol{d}(t)$, $t = 1 \ldots, 66$, and $\boldsymbol{d}(t)$ is the degree sequence of the *Drosophila melanogaster* network at time t.

Step 2: Calculate $\hat{\boldsymbol{c}} = \arg\max \mathcal{Q}(\boldsymbol{e}, \tilde{\boldsymbol{g}})$.

Step 3: Use the proposed perturbation scheme in Section 2.5 to perform a perturbation of $\tilde{\boldsymbol{g}}$ at level γ. Denote the perturbed time-varying network as $\tilde{\boldsymbol{g}}_{\boldsymbol{\gamma}}$.

Step 4: Find $\hat{\boldsymbol{c}}_{\boldsymbol{\gamma}} = \arg\max \mathcal{Q}(\boldsymbol{e}, \tilde{\boldsymbol{g}}_{\boldsymbol{\gamma}})$.

Step 5: Calculate the normalized mutual information between $\hat{\boldsymbol{c}}$ and $\hat{\boldsymbol{c}}_{\boldsymbol{\gamma}}$.

For each γ, the above procedure is repeated 10 times and the average is reported.

In randomly generated time-varying networks, the time-varying degree sequence $\mathcal{D}$ observed in the *Drosophila melanogaster* network is preserved. That is, the number of connections to gene i at time t in the randomly generated network is the same as that in the observed network. However, in the random network, edges are randomly placed between nodes with no community configurations. For random time-varying networks, there should be no community structure. This implies that for any partition, the number of intracommunity edges should be close to the number of intercommunity edges. In this case, since $\hat{\boldsymbol{c}}$ is obtained by maximizing $\mathcal{Q}(\boldsymbol{e}, \tilde{\boldsymbol{g}})$, $\hat{\boldsymbol{c}}$ is expected to be very sensitive to structure perturbations. Even a small perturbation can cause a significant change in the output of the algorithm.

It is worth mentioning that, in the real data analysis step, we also considered Method 1 and Method 2 described in Section 3. Since the topology of the network changes remarkably over time (as observed in Figure 1), Method 1 yields highly variable community detection results when different snapshots are used. Comparing the community detection results that we obtained using networks from two adjacency time points, $G(t_l)$ and $G(t_{l+1})$, we find that the normalized mutual information is on average 0.66. Using Method 2, nine communities are identified. However, the community findings are difficult to interpret (see supplementary file for a summary of the results). This could be because the fact that in Method 2, the networks at all time points are aggregated, and the temporal aspect of the network is completely ignored. For example, using Method 2, Group 3 consists of genes that are related to sensory organ development, cardiovascular development, and epithelium development, and Group 6 consists of genes that are related to gland development and epithelium development.

5. Consistency

The consistency property of community detection methods for static networks has been well studied in the literature (Bickel and Chen 2009; Rohe, Chatterjee, and Yu 2011; Zhao, Levina, and Zhu 2012; Jin 2015). However, the theoretical properties of community-finding methods on time-varying networks remain largely unaddressed. In this section, we investigate the theoretical properties of our proposed method for finding common modules in a time-varying network. To do so, we propose a time-varying stochastic block model framework with a temporal correlation structure.

Temporal Stochastic Block Model (TSBM)

1. Dynamic network $\mathcal{G} = (G(t), t = t_1, \ldots, t_S)$, has node set $V = \{v_1, \ldots, v_n\}$ and a latent community label $\boldsymbol{c} = (c_1, \ldots, c_n)$, where $c_i \in [1, \ldots, K]$ is the community that node i belongs to.
2. The marginal distribution of $\boldsymbol{c} = (c_1, \ldots, c_n)$ is multinomial, with parameters $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$, that is, nodes independently belong to community j with probability π_j, $1 \le j \le K$.
3. Define the time-varying probability matrix $\Theta(t) = \begin{pmatrix} \theta_{11}(t) & \cdots & \theta_{1K}(t) \\ \vdots & \ddots & \vdots \\ \theta_{K1}(t) & \cdots & \theta_{KK}(t), \end{pmatrix}$ where $\theta_{kl}(t)$ is a function of t.
4. Given $\boldsymbol{c}$, the edges $A_{ij}(t_l)$ are independent Bernoulli random variables with
$$A_{ij}(t_l) = uA_{ij}(t_{l-1}) + (1-u)v,$$
where $u \sim^{\text{iid}}$ Bernoulli(α) and
$$v \sim^{\text{ind}} \text{Bernoulli}\left(\frac{\theta_{c_ic_j}(t_l) - \alpha\theta_{c_ic_j}(t_{l-1})}{1-\alpha}\right),$$
where $0 \le \alpha < 1$, $\theta_{kl}(t_l) \ge \alpha\theta_{kl}(t_{l-1})$ and $1 - \theta_{kl}(t_l) \ge \alpha(1 - \theta_{kl}(t_{l-1}))$.

To have $\frac{\theta_{c_ic_j}(t_l) - \alpha\theta_{c_ic_j}(t_{l-1})}{1-\alpha} \in [0, 1]$, we have $\theta_{kl}(t_l) \ge \alpha\theta_{kl}(t_{l-1})$ and $1 - \theta_{kl}(t_l) \ge \alpha(1 - \theta_{kl}(t_{l-1}))$. The random variable v is defined so that $A_{ij}(t)$ is a Bernoulli random variable with $P(A_{ij}(t) = 1) = \theta_{c_ic_j}(t)$. Hence, the marginal distribution of $A_{ij}(t)$ is Bernoulli$(\theta_{c_ic_j}(t))$ and marginally, $A(t)$ follows a stochastic block model with probability matrix $\Theta(t)$. However, under this model, $A_{ij}(t_l)$ and $A_{ij}(t_{l-1})$ are correlated with

$$\text{corr}(A_{ij}(t_l), A_{ij}(t_{l-1})) = \alpha\sqrt{\frac{\theta_{c_ic_j}(t_{l-1})(1 - \theta_{c_ic_j}(t_{l-1}))}{\theta_{c_ic_j}(t_l)(1 - \theta_{c_ic_j}(t_l))}}.$$

When $\alpha = 0$, $A_{ij}(t_l)$, $l = 1, \ldots, S$, become independent samples from a stochastic block model with a time-varying probability matrix. If the probability matrix $\Theta(t)$ remains homogenous over time, we have

$$\text{corr}(A_{ij}(t_l), A_{ij}(t_{l-k})) = \alpha^k, k = 1, 2, \ldots.$$

Given $\Theta(t)$, the greater α is, the stronger the correlation is between $A_{ij}(t_l)$ and $A_{ij}(t_{l-k})$, $k = 1, 2, \ldots$.

Under the proposed framework in Section 2, we identify common modules by finding the maximizer of the modularity function in (4), that is,

$$\hat{\boldsymbol{c}} = \arg \max_{\substack{\boldsymbol{e}=(e_1,\ldots,e_n)\\ e_i\in\{1,\ldots,K\}}} \mathcal{Q}(\boldsymbol{e}, \mathcal{G}), \tag{6}$$

where $\mathcal{G}$ is a time-varying network. Under the temporal stochastic block model framework, we can consider two possible asymptotic regimes for the time-varying network:

1. Number of snapshots S is fixed and the graph size $n \to \infty$.
2. Number of snapshots $S \to \infty$ and the graph size n is fixed.

Under the first regime, the consistency property of (6) directly follows the results in Bickel and Chen (2009) and Zhao, Levina, and Zhu (2012). The second regime has not been considered in the literature, and we will focus on this setting in our work. We show that the estimator $\hat{\boldsymbol{c}}$ has the following consistency property.

Theorem 1. Consider $\mathcal{G}$ from a temporal stochastic block model with $\Theta(t)$. Define W^t to be a $K \times K$ matrix, with $W^t_{ab} = \pi_a\pi_b\theta_{ab}(t)/W^t_0$, $W^t_0 = \sum_{ab}\pi_a\pi_b\theta_{ab}(t)$ and define $C^t = W^t - (W^t\mathbf{1})(W^t\mathbf{1})^T$, $t = t_1, \ldots, t_S$. If the time-varying stochastic block model satisfies $\sum_t\sum_a C^t_{aa} > 0$ and $\sum_t\sum_{ab} C^t_{ab} < 0$ for $a \ne b$, then we have

$$\forall\epsilon > 0, P\left[\left(\frac{1}{n}\sum_{i=1}^{n} I(\hat{c}_i \ne c_i)\right) < \epsilon\right] \to 1 \text{ as } S \to \infty, \tag{7}$$

where $\hat{\boldsymbol{c}} = \arg\max_{\boldsymbol{e}} \mathcal{Q}(\boldsymbol{e}, \mathcal{G})$.

See supplementary file for proof. The consistency in Theorem 1 and the consistency in Bickel and Chen (2009) and Zhao, Levina, and Zhu (2012) are defined in two different asymptotic regimes. Specifically, in our setting, the number of graphs from the TSBM goes to infinity, while in each graph, the number of nodes n remains fixed. This corresponds to a setting where a large number of snapshots are taken of a time-varying network with a fixed number of nodes.

The constraints on the parameters in Theorem 1 essentially require that, on average, links are more likely to be established within blocks than between blocks, even though communities are not "active" throughout the entire observation window. One example is the $\Theta(t)$ in Simulation 1.2 and Simulation 2.

In the simplest case, when $K = 2$ and $\Theta(t)$ is time-homogenous, the conditions $C_{aa} > 0$, $C_{ab} < 0$ simplify to

$$\theta_{11}\theta_{22} > (\theta_{12})^2. \tag{8}$$

We can see that condition (8) is satisfied by TSBMs with the following parameter constraints:

$$P(A_{ij} = 1|c_i = 1, c_j = 1) > P(A_{ij} = 1|c_i = 1, c_j = 2), \tag{9}$$

$$P(A_{ij} = 1|c_i = 2, c_j = 2) > P(A_{ij} = 1|c_i = 1, c_j = 2). \tag{10}$$

Constraints (9) and (10) correspond to the motivation that links are more likely to be established within communities than between communities.

6. Discussion

In this article, we provide a statistical framework for finding common modules in the time-varying *Drosophila melanogaster* gene regulation network. We also propose a significance test and a robustness test for our community findings. Using the proposed method, we identify interesting functional modules in the gene regulation network. We also show that under a time-varying stochastic block model framework, the proposed modularity function is consistent.

Our method can be extended to directed networks. A few methods have been proposed for finding communities in static directed networks using modified modularity functions (see Fortunato 2010 for a review). The incorporation of directed edges will require a modification of our null model. One possibility would be to use the configuration model for directed networks.

Consider the proposed temporal stochastic block model framework. Another approach to find $\hat{\boldsymbol{c}}$ is to estimate $\Theta(t), \alpha, \boldsymbol{\pi}$, and $\boldsymbol{c}$ through maximizing the likelihood function. Estimating $\Theta(t)$ and $\boldsymbol{c}$ in the model can be very challenging, since the forms of $\theta_{kh}(t)$, $k, h = 1, \ldots, K$ are unknown and the space containing all possible community assignments $\boldsymbol{c}$ is of size K^n. Moreover, it is unclear whether $\hat{\boldsymbol{c}}$ will be consistent as the number of snapshots $S \to \infty$. With the proposed temporal stochastic block model, the time-varying probability matrix specification, parameter estimation, and the theoretical properties of $\hat{\boldsymbol{c}}$ is an interesting topic to investigate next.

Supplementary Materials

The supplementary materials contain proof of Theorem 1, details of the genes contained in the identified communities, further information on the gene ontology enrichment analysis and community findings when applying Method 2 to the Drosophila melanogaster gene regulation network.

Acknowledgments

The authors are grateful to Professor Eric P. Xing and Professor Le Song for providing the gene regulation network data. The authors thank Professor Bingyun Sun for her valuable input on our enrichment analysis and Michael Grosskopf, Christina Nieuwoudt, and Yunlong Nie for their proofreading. The authors are very grateful for the constructive comments from the editor, associate editor, and two anonymous reviewers. These comments have helped greatly in improving this work.

Funding

This research is partially supported by a discovery grant from the Natural Sciences and Engineering Research Council of Canada (NSERC) to the second author.

References

Airoldi, E. M., Blei, D. M., Fienberg, S. E., and Xing, E. P. (2008), "Mixed-Membership Stochastic Blockmodels," *Journal of Machine Learning Research*, 9, 1981–2014. [996]

Arbeitman, M. S., Furlong, E. E., Imam, F., Johnson, E., Null, B. H., Baker, B. S., Krasnow, M. A., Scott, M. P., Davis, R. W., and White, K. P. (2002), "Gene Expression During the Life Cycle of Drosophila melanogaster," *Science*, 297, 2270–2275. [994,995]

Bansal, S., Showmich, S., and Paymal, P. (2011), "Fast Community Detection for Dynamic Complex Networks," *Communications in Computer and Information Science*, 116, 196–207. [996]

Bassett, D. S., Porter, M. A., Wymbs, S. F., Grafton, S. T., Carlson, J. M., and Mucha, P. J. (2013), "Robust Detection of Dynamic Community Structure in Networks," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 23, 013142. [996]

Bender, E., and Canfield, R. (1978), "The Asymptotic Number of Labeled Graphs With Given Degree Sequences," *Journal of Combinatorial Theory A*, 24, 296–307. [996]

Bickel, P., and Chen, A. (2009), "A Non-Parametric View of Network Models and Newman-Girvan and Other Modularities," *Proceedings of the National Academy of Sciences*, 106, 21068–21073. [996,1005,1006]

Bickel, P. J., and Sarkar, P. (2016), "Hypothesis Testing for Automated Community Detection in Networks," *Journal of the Royal Statistical Society*, Series B, 78, 253–273. [996]

Blondel, V. D. (2011), "The Louvain Method for Community Detection in Large Networks," available at *https://perso.uclouvain.be/vincent.blondel/research/louvain.html*. [998]

Blondel, V. D., Guillaume, J. L., Lambiotte, R., and Lefebvre, E. (2008), "Fast Unfolding of Communities in Large Networks," *Journal of Statistical Mechanics: Theory and Experiment*, 10, P10008. [998]

Bollobás, B. (1980), "A Probabilistic Proof of an Asymptotic Formula for the Number of Labelled Regular Graphs," *European Journal of Combinatorics*, 1, 311–316. [996]

Brandes, U., Delling, D., Gaertler, M., Gorke, R., Hoefer, M., Nikoloski, Z., and Wagner, D. (2008), "On Modularity Clustering," *IEEE Transactions on Knowledge and Data Engineering*, 20, 172–188. [998]

Chung, F., and Lu, L. (2002), "Connected Components in Random Graphs With Given Expected Degree Sequences," *Annals of Combinatorics*, 6, 125–145. [999]

Clauset, A., Newman, M. E. J., and Moore, C. (2004), "Finding Community Structure in Very Large Networks," *Physical Review E*, 70, 066111. [998]

Danon, L., Diaz-Guilera, A., Duch, J., and Arenas, A. (2005), "Comparing Community Structure Identification," *Journal of Statistical Mechanics: Theory and Experiment*, 2005, P09008. [999]

De Bivort, B., Huang, S., and Bar-Yam, Y. (2007), "Empirical Multiscale Networks of Cellular Regulation," *PLoS Computational Biology*, 3, e207. [995]

Flake, G. W., Lawrence, S., and Giles, C. L. (2000), "Efficient Identification of Web Communities," in *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 150–160. [996]

Fortunato, S. (2010), "Community Detection in Graphs," *Physics Reports*, 428, 75–174. [996,997,998,1007]

Guimera, R., Sales-Pardo, M., and Amaral, L. A. S. (2004), "Modularity From Fluctuations in Random Graphs and Complex Networks," *Physical Review E*, 70, 025101. [998]

Gulbahce, S., and Lehmann, S. (2008), "The Art of Community Detection," *BioEssays*, 30, 934–938. [994]

Handcock, M. S., Raftery, A. E., and Tantrum, J. M. (2007), "Model-Based Clustering for Social Networks," *Journal of the Royal Statistical Society*, Series A, 170, 301–354. [996]

Jin, J. (2015), "Fast Community Detection by SCORE," *The Annals of Statistics*, 43, 57–89. [996,1006]

Jorgensen, P., Nishikawa, J. L., Breitkreutz, B. J., and Tyers, M. (2002), "Systematic Identification of Pathways that Couple Cell Growth and Division in Yeast," *Science*, 297, 395–400. [1004]

Kranakis, E. (2013), *Advances in Network Analysis and Its Application*, Berlin Heidelberg: Springer. [997]

Massen, C., and Doye, J. (2005), "Identifying Communities Within Energy Landscapes," *Physical Review E*, 71, 046101. [998]

Misra, S., Crosby, M. A., Mungall, C. J., Matthews, B. B., Campbell, K. S., Hradecky, P., Huang, Y., Kaminker, J. S., Millburn, G. H., Prochnik, S. E., and Smith, C. D. (2002), "Annotation of the Drosophila Melanogaster Euchromatic Genome: A Systematic Review," *Genome Biology*, 3, 1. [994]

Mucha, P. J., Richardson, T., Macon, K., Porter, M. A., and Onnela, J. P. (2010), "Community Structure in Time-Dependent, Multiscale, and Multiplex Networks," *Science*, 328, 876–878. [996]

Newman, M. E. J. (2004), "Analysis of Weighted Networks," *Physical Review E*, 70, 056131. [1000]

——— (2006), "Finding Community Structure in Networks Using the Eigenvectors of Matrices," *Physical Review E*, 74, 035104. [997]

Newman, M. E. J., and Girvan, M. (2004), "Finding and Evaluating Community Structure in Networks," *Physical Review E*, 69, 026113. [996,998,1000]

Nguyen, S. P., Dinh, T. S., Shen, Y., and Thai, M. T. (2014), "Dynamic Social Community Detection and Its Applications," *PloS One*, 9, e91431. [996]

Nowicki, K., and Snijders, T. A. B. (2001), "Estimation and Prediction for Stochastic Block Structures," *Journal of the American Statistical Association*, 96, 1077–1087. [996]

Pandey, U. B., and Nichols, C. D. (2011), "Human Disease Models in Drosophila Melanogaster and the Role of the Fly in Therapeutic Drug Discovery," *Pharmacological Reviews*, 63, 411–436. [994]

Reichardt, J., and Bornholdt, S. (2006), "Statistical Mechanics of Community Detection," *Physical Review E*, 74, 016110. [998]

Reiter, L. T., Potocki, L., Chien, S., Gribskov, M., and Bier, E. (2001), "A Systematic Analysis of Human Disease-Associated Gene Sequences in Drosophila Melanogaster," *Genome Research*, 11, 1114–1125. [994]

Rohe, K., Chatterjee, S., and Yu, B. (2011), "Spectral Clustering and the High-Dimensional Stochastic Blockmodel," *The Annals of Statistics*, 39, 1878–1915. [1005]

Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D., and Friedman, N. (2003), "Module Networks: Identifying Regulatory Modules and Their Condition-Specific Regulators From Gene Expression Data," *Nature Genetics*, 34, 166–176. [994,995]

Shi, J., and Malik, J. (2000), "Normalized Cuts and Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22, 888–905. [996]

Song, L., Kolar, M., and Xing, E. P. (2009), "KELLER: Estimating Time-Varying Interactions Between Genes," *Bioinformatics*, 25, 128–136. [994]

Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S. L., Golub, T. R., Lander, E. S., and Mesirov, J. P. (2005), "Gene Set Enrichment Analysis: A Knowledge-Based Approach for Interpreting Genome-Wide Expression Profiles," *Proceedings of the National Academy of Sciences*, 102, 15545–15550. [1002]

Wakita, K., and Tsurumi, T. (2007), "Finding Community Structure in Mega-Scale Social Networks," in *Proceedings of the 16th International Conference on World Wide Web*, pp. 1275–1276. [998]

Wang, J., Duncan, D., Shi, Z., and Zhang, B. (2013), "Web-Based Gene Set Analysis Toolkit (WebGestalt): Update 2013," *Nucleic Acids Research*, 41, W77–W83. [1002]

Zhang, J., and Chen, Y. (2016), "A Hypothesis Testing Framework for Modularity Based Network Community Detection," *Statistica Sinica*, 27, 437–456. [997]

Zhao, G., Schriefer, L. A., and Stormo, G. D. (2007), "Identification of Muscle-Specific Regulatory Modules in Caenorhabditis Elegans," *Genome Research*, 17, 348–357. [995]

Zhao, Y., Levina, E., and Zhu, J. (2012), "Consistency of Community Detection in Networks Under Degree-Corrected Stochastic Block Models," *The Annals of Statistics*, 40, 2266–2292. [1006]