Factor Analysis on a Graph

Masayuki Karasuyama
Department of Computer Science
Nagoya Institute of Technology
Aichi, Japan

Hiroshi Mamitsuka
Institute for Chemical Research
Kyoto University
Kyoto, Japan

Department of Computer Science
Aalto University
Espoo, Finland

Abstract

Graph is a common way to represent relationships among a set of objects in a variety of data analysis fields. We consider the case that the input data is not only a graph but also numerical features, each one of which corresponds to a node in the graph. In practice, the primary importance is often in understanding interactions on the graph nodes which effect on covariance structure of the numerical features. We propose a Gaussian based analysis which is a combination of graph constrained covariance matrix estimation and *factor analysis* (FA). We show that this approach, called graph FA, has desirable interpretability. In particular, we prove the connection between graph FA and a graph node clustering based on a perspective of *kernel method*. This connection indicates that graph FA is effective not only on the conventional noise-reduction explanation of the observation by FA but also on identifying important subgraphs. The experiments on synthetic and real-world datasets demonstrate the effectiveness of the approach.

1 Introduction

Analyzing relationships among objects is an essential task for a variety application areas of machine learning such as bioinformatics (Ideker et al., 2002), web mining (Kosala and Blockeel, 2000), and social network analysis (Boyd and Ellison, 2007). *Graph* is an established way to describe those relationships mathematically, in which a *node* represents an object and an *edge* connecting two nodes represents a relationship between the two nodes. For instance, *protein-protein interaction (PPI) networks* regard a protein as a node and an interaction as an edge in a graph. On the other hand, we often have numerical data as well, i.e., a set of feature vectors, and in particular we consider the case that each dimension of a feature vector corresponds to each node of the given graph. In PPI networks, a gene expression value corresponds to each protein (a node), and they are required to analyze simultaneously. For practical analysis , the primary importance is often in analyzing coordinated variations of the features *on* the graph to understand underlying mechanisms of how the graph is related to the observed numerical data.

Based on the motivation mentioned above, our problem setting in this paper can be stated as follows:

> Identifying covariance structure generated from interactions on given graph nodes, and creating an interpretable representation of those interactions.

When we consider the numerical input data only, *principal component analysis* (PCA) (Jolliffe, 2002) and *factor analysis* (Harman, 1960) are effective basic tools to achieve this goal, which provide an interpretable data reduction using a linear latent variable model. Obviously, these well-known methods are not optimal for our purpose because they do not consider the graph at all.

For better interpretability, we employ a Gaussian based approach, in which the graph connectivity is interpreted as conditional dependency. Our method, called *graph FA*, first maps numerical data onto a graph using a Gaussian model having the graph connectivity as the conditional dependency, by which we extract covariance explained by the graph. From the estimated Gaussian, we construct a lower dimensional linear model by FA. The extracted lower dimensional representation is easy to interpret because of the Gaussian assumption, and summarizes coordinated variations explained by the graph connectivity, while naive application of classical methods (like PCA) can not

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84.

deal with the graph structure. We emphasize that the procedure of graph FA is simple but has not been explored in depth so far.

An important contribution in this paper is to explore the connection between graph FA and a *graph node clustering*. We prove that graph FA is a continuous approximation of a graph node clustering based on a *graph path-based kernel*. This connection indicates that graph FA is effective not only on the conventional noise-reduction explanation of the observation by FA but also on identifying important subgraphs having strong interactions. In our experiments, we perform a clustering based empirical evaluation on synthetic datasets, and also show results on a significant subgraph identification problem in a biological network.

2 Factor Analysis on a Graph

Suppose that we have n feature vectors $\mathcal{X} := \{\boldsymbol{x}_i\}_{i=1}^n$, where $\boldsymbol{x}_i \in \mathbb{R}^d$. Without loss of generality, the mean of $\boldsymbol{x}_i$ is assumed to be zero. A graph $\mathcal{G} := \{\mathcal{V}, \mathcal{E}\}$ is defined by a set of nodes $\mathcal{V} := \{1, \ldots, d\}$ and a set of pairs of nodes $\mathcal{E} := \{(i,j) \mid i,j \in \mathcal{V}\}$. Let $\boldsymbol{\Sigma} \in \mathbb{R}_+^{d\times d}$ be a covariance matrix and $\boldsymbol{\Theta} := \boldsymbol{\Sigma}^{-1}$ be the corresponding inverse covariance matrix, where $\mathbb{R}_+^{d\times d}$ is a $d \times d$ positive definite matrix. We focus on a Gaussian distribution which has the conditional dependency pattern specified by the given graph $\mathcal{G}$:

$$\mathcal{T} := \{\boldsymbol{\Theta} \mid \boldsymbol{\Theta}_{ij} = 0 \text{ for } (i,j) \notin \tilde{\mathcal{E}},\ \boldsymbol{\Theta} \in \mathbb{R}_+^{d\times d}\},$$

where $\tilde{\mathcal{E}}$ includes the diagonal entries $\tilde{\mathcal{E}} := \mathcal{E} \cup \{(i,i)\}_{i=1}^d$. In this set of matrices, the (i,j)-element should be 0 if a pair of nodes (i,j) does not have an edge.

Let $\boldsymbol{S}$ be a sample covariance matrix estimated from $\{\boldsymbol{x}_i\}_{i=1}^n$. We first project $\boldsymbol{S}$ onto the set $\mathcal{T}$ to extract covariance associated with graph connectivity. We consider the minimization of *Kullback-Leibler divergence* (KL divergence) (Kullback and Leibler, 1951):

$$KL(p(\boldsymbol{x}) \| q(\boldsymbol{x})) = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})} d\boldsymbol{x},$$

where p and q are d-dimensional probabilistic density functions. By substituting the Gaussian distribution $\mathcal{N}(\boldsymbol{0}, \boldsymbol{S})$ into p, and the Gaussian distribution $\mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$, which has the sparse inverse covariance $\boldsymbol{\Theta}$, into q respectively, the following minimization problem is derived:

$$\begin{aligned} \min_{\boldsymbol{\Theta} \in \mathbb{R}_+^{d\times d}} \quad & -\log\det\boldsymbol{\Theta} + \operatorname{trace}(\boldsymbol{\Theta}\boldsymbol{S}) \qquad (1) \\ \text{s.t.} \quad & \boldsymbol{\Theta}_{ij} = 0,\ (i,j) \notin \tilde{\mathcal{E}}. \end{aligned}$$

This problem is also known as the maximum likelihood estimation of *graphical Gaussian model* (GGM) (Whittaker, 1990). Since KL divergence can be interpreted as a pseudo-distance between two density functions, the solution of the minimization problem (1) is the best approximation to $\mathcal{N}(\boldsymbol{0}, \boldsymbol{S})$ by the Gaussian having the dependency structure specified by the graph $\mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$. Our intuition behind this process is twofold:

- Extracting covariance which can be explained by the graph
- Removing covariance which can not be explained by the graph

For the Gaussian estimated by (1), we consider *factor analysis* (Harman, 1960), which is a standard statistical model represented by the following linear model:

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{f} + \boldsymbol{\epsilon},\ \boldsymbol{f} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}),\ \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Psi}),$$

where $\boldsymbol{A} \in \mathbb{R}^{d\times k}$ is a factor loading matrix having a smaller number of columns $k < d$ and $\boldsymbol{f} \in \mathbb{R}^k$ is a k-dimensional latent vector, and $\boldsymbol{\epsilon} \in \mathbb{R}^d$ is an independently distributed error term having a diagonal matrix $\boldsymbol{\Psi} \in \mathbb{R}_+^{d\times d}$ as a covariance matrix. Since $\boldsymbol{f}$ and $\boldsymbol{\epsilon}$ are assumed to be independent, the covariance matrix of the above model is $\boldsymbol{A}\boldsymbol{A}^\top + \boldsymbol{\Psi}$.

Suppose that $\widehat{\boldsymbol{\Sigma}} := \widehat{\boldsymbol{\Theta}}^{-1}$ is the inverse of the optimal solution $\widehat{\boldsymbol{\Theta}}$ of the minimization of KL divergence (1), the maximum likelihood estimation of factor analysis for $\widehat{\boldsymbol{\Sigma}}$ is

$$\begin{aligned} \min_{\boldsymbol{A}\in\mathbb{R}^{d\times k}, \boldsymbol{\Psi}\in\mathcal{D}_+^d} & -\log\det(\boldsymbol{A}\boldsymbol{A}^\top + \boldsymbol{\Psi})^{-1} \\ & + \operatorname{trace}((\boldsymbol{A}\boldsymbol{A}^\top + \boldsymbol{\Psi})^{-1}\widehat{\boldsymbol{\Sigma}}), \qquad (2) \end{aligned}$$

where $\mathcal{D}_+^d$ is a space of $d\times d$ diagonal matrices in which all the diagonal elements are positive.

The resulting $\boldsymbol{A}$ is expected to explain the covariance structure on the graph. One of the important goals of the standard factor analysis is to interpret the factor loading matrix $\boldsymbol{A}$ describing relations between features and factors. Our interest is in a coordinated variation occurring on a *connected subgraph* (defined as subsets of the nodes and edges of the entire graph $\mathcal{G}$ and there must exist a path between every pair of nodes). Hereafter, we will show that the loading matrix $\boldsymbol{A}$ obtained by the above procedure has advantageous properties for this purpose mainly through a perspective of *kernel method*.

3 Kernel-based Interpretation

In this section, we show a connection between graph FA and a graph node clustering based on a *path-based*

graph kernel. This connection indicates usefulness of graph FA for simultaneously analyzing graph connectivity and Gaussian covariance structure, which is our main focus in this paper.

3.1 Sparse Inverse Covariance as a Kernel Matrix for a Graph

Since $\widehat{\boldsymbol{\Sigma}}$ is positive-definite, it can be regarded as a *kernel matrix*:

$$\boldsymbol{K}^{GGM} := \widehat{\boldsymbol{\Sigma}}.$$

We call this $\boldsymbol{K}^{GGM}$ *GGM kernel*. GGM kernel would be expected to reflect the graph connectivity structure since $\widehat{\boldsymbol{\Sigma}}(= \widehat{\boldsymbol{\Theta}}^{-1})$ is calculated based on the graph $\mathcal{G}$ constraint shown in (1).

A general way to define a kernel function between two nodes on a graph is to sum evaluation scores for a set of paths between the two nodes (S.-Taylor and Cristianini, 2004). Let $\mathcal{P}_{ij}$ be a set of paths between nodes i and j on the graph. A path $\mathcal{P} \in \mathcal{P}_{ij}$ is defined by a set of nodes ordered from i to j, i.e., $\mathcal{P} := \{(p_1, \ldots, p_m) | p_1 = i, p_m = j, m \leq d\}$. An evaluation $K_{\mathcal{P}}$ for a path $\mathcal{P} \in \mathcal{P}_{ij}$ is defined by a product of scores for neighboring nodes on the path:

$$K_{\mathcal{P}} := \prod_{\{k | p_k, p_{k+1} \in \mathcal{P}\}} \kappa_{p_k, p_{k+1}}, \tag{3}$$

where $\kappa_{p_k, p_{k+1}}$ is a base score for a pair of neighboring nodes. This score for a path $K_{\mathcal{P}}$ represents a connection between i and j through the path $\mathcal{P}$, and thus the simplest way to define a kernel for a node pair (i, j) is to sum up all possible paths $\sum_{\mathcal{P} \in \mathcal{P}_{ij}} K_{\mathcal{P}}$.

In our case, we define the base score $\kappa_{p_i, p_{i+1}}$ by

$$\kappa_{p_i, p_{i+1}} = -\boldsymbol{\Theta}_{p_i, p_{i+1}}. \tag{4}$$

In the quadratic term of the Gaussian density function $-\boldsymbol{x}^\top \boldsymbol{\Theta} \boldsymbol{x}$, $-\boldsymbol{\Theta}_{ij}$ is a coefficient for the interaction term of i- and j-th dimensions. We thus regard $-\boldsymbol{\Theta}_{ij}$ as an evaluation score for the strength of the connection of neighboring nodes, and it is also known that the inverse covariance matrix $\boldsymbol{\Theta}$ represents conditional independency. In particular, when the diagonal entries of $\boldsymbol{\Theta}$ is scaled as 1, $-\boldsymbol{\Theta}_{ij}$ is identical to the conditional correlation between i and j.

Using the base score $\kappa_{p_i, p_{i+1}}$ (4), GGM kernel can be decomposed into the following weighted sum of possible paths:

Theorem 1 (Path-based decomposition of GGM kernel)**.** *The (i, j)-element of the kernel matrix $\boldsymbol{K}^{GGM}$ can be written as a weighted sum of all possible paths between i and j as follows:*

$$\boldsymbol{K}^{GGM}_{ij} = \sum_{\mathcal{P} \in \mathcal{P}_{ij}} \det(\boldsymbol{\Sigma}_{\mathcal{P}}) K_{\mathcal{P}}, \tag{5}$$

where $\boldsymbol{\Sigma}_{\mathcal{P}}$ is a sub-matrix of $\boldsymbol{\Sigma}$ constructed by features in the path $\mathcal{P}$, and $K_{\mathcal{P}}$ is defined by (3) with the base score (4).

Sketch of proof. The equation can be easily derived from a covariance decomposition theorem (Jones and West, 2005) with some simple algebric operations. See supplementary appendix A for detail. □

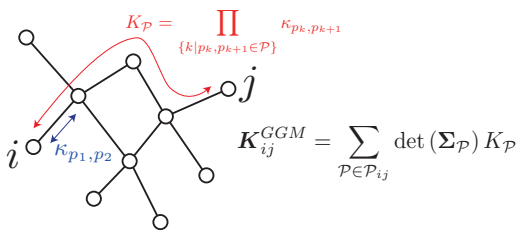


Figure 1: A schematic illustration of GGM kernel, which is a weighted sum of all possible paths.

Figure 1 shows an illustration of GGM kernel. According to (5), each path is weighted by $\det(\boldsymbol{\Sigma}_{\mathcal{P}})$ which evaluates amount of information of features in $\mathcal{P}$. The determinant of covariance is also called *generalized variance* as a measure of multi-dimensional dispersion.

Our discussion so far can be summarized by the following remark:

Remark 1. *GGM kernel can be written as a sum of evaluation scores of all possible paths weighted by a generalized variance for dimensions included in each path. Based on this analysis, we see that an intuitive interpretation of GGM kernel is an evaluation of a relationship between two nodes on a graph through the connectivity structure.*

Kernels for graph nodes have been widely studied such as diffusion kernel (Smola and Kondor, 2003). A noticeable characteristic of GGM kernel is its direct link to Gaussian distribution of the input space which provides highly interpretable results of graph FA.

3.2 GGM Kernel k-means

We show an equivalence between kernel k-means clustering with GGM kernel and graph FA, which indicates the subgraph identification property of graph FA. Suppose $\{\boldsymbol{\phi}_j\}_{j=1}^d$ is a set of d instances in some feature space $\boldsymbol{\phi}_j \in \mathcal{F}$ induced by a kernel matrix $\boldsymbol{K}^{GGM}$, i.e.,

$\boldsymbol{K}^{GGM} = \boldsymbol{\Phi}\boldsymbol{\Phi}^\top$ where $\boldsymbol{\Phi} := [\ \boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_d\ ]^\top$. We regard $\boldsymbol{\phi}_i$ as a feature representation for a graph node.

Assuming that we already have the maximum likelihood estimate $\widehat{\boldsymbol{\Psi}}$ for $\boldsymbol{\Psi}$. Since the diagonal elements of $\boldsymbol{\Psi}$ represent noise strength in each node, we consider defining instance importance for kernel k-means problem based on $\widehat{\boldsymbol{\Psi}}$. The objective function of our weighted kernel k-means is then defined by

$$\sum_{i=1}^{k} \sum_{j \in \mathcal{C}_i} \widehat{\psi}_j^{-1} \left\| \boldsymbol{\phi}_j - \boldsymbol{\mu}_i \right\|^2,$$

where $\mathcal{C}_i$ for $i = 1, \ldots, k$, is an index set of the i-th cluster, and $\boldsymbol{\mu}_i$ is a centroid of the i-th cluster. This weighting means that errors for features with larger independent components are less important. Suppose that $\boldsymbol{Z}$ is a $d \times k$ cluster indicator matrix, in which (i, j) element takes 1 if $i \in \mathcal{C}_j$ or takes 0 otherwise. We extend $\boldsymbol{Z}$ to have negative indicator, i.e., it takes 1 or -1 if $i \in \mathcal{C}_j$ or takes 0 otherwise. This means that each cluster can select $\boldsymbol{\phi}_i$ or $-\boldsymbol{\phi}_i$ adaptively to minimize the above objective (ignoring sign differences).

Spectral relaxation is often used to approximate clustering methods, by which the weighted kernel k-means can be reformulated as follows (see supplementary appendix C for derivation):

Definition 1 (Spectral relaxation of weighted kernel k-means)**.** *Defining* $\boldsymbol{V}_k := \widehat{\boldsymbol{\Psi}}^{-1/2}\boldsymbol{Z}\boldsymbol{C}^{1/2}$*, which lead* $\boldsymbol{V}_k^\top \boldsymbol{V}_k = \boldsymbol{C}^{1/2}\boldsymbol{Z}\widehat{\boldsymbol{\Psi}}^{-1}\boldsymbol{Z}\boldsymbol{C}^{1/2} = \boldsymbol{C}^{1/2}\boldsymbol{C}^{-1}\boldsymbol{C}^{1/2} = \boldsymbol{I}$*, we derive the spectral relaxation of the weighted kernel k-means as follows:*

$$\begin{aligned} \max_{\boldsymbol{V}_k \in \mathbb{R}^{d \times k}} \quad & \operatorname{trace}\left(\boldsymbol{V}_k^\top \widehat{\boldsymbol{\Psi}}^{-1/2} \widehat{\boldsymbol{\Sigma}} \widehat{\boldsymbol{\Psi}}^{-1/2} \boldsymbol{V}_k \right), \quad (6) \\ \textit{subject to} \quad & \boldsymbol{V}_k^\top \boldsymbol{V}_k = \boldsymbol{I}. \end{aligned}$$

From the definition of $\boldsymbol{V}_k$, the indicator matrix $\boldsymbol{Z}$ is estimated as

$$\widehat{\boldsymbol{Z}} = \widehat{\boldsymbol{\Psi}}^{1/2} \boldsymbol{V}_k \boldsymbol{C}^{-1/2}. \quad (7)$$

Based on the above definition, the following equivalence between FA and weighted kernel k-means can be derived:

Theorem 2. *Given* $\widehat{\boldsymbol{\Psi}}$ *as a weight, an optimal solution of the spectral relaxation of weighted kernel k-means* $\widehat{\boldsymbol{Z}}$ *(7) is identical to* $\widehat{\boldsymbol{A}}$ *of (2) up to a constant in each column vector.*

Sketch of proof. Given $\widehat{\boldsymbol{\Psi}}$, the first order optimality condition for $\boldsymbol{A}$ in FA can be formulated in a similar form of the eigenvalue decomposition to (6), which results in the theorem above. See supplementary appendix D for more detail. □

We would like to stress the following important remark to clarify the advantage of graph FA.

Remark 2. *Graph FA is equivalent to the spectral relaxation of weighted k-means in the GGM kernel induced feature space, which reflects the graph connectivity through the path-based evaluation. This means that the matrix* $\widehat{\boldsymbol{A}}$ *approximately indicates strongly connected subgraphs, while summarizing covariance structure on the graph, simultaneously.*

3.3 Post-processing to Improve Interpretability

In the context of factor analysis (Harman, 1960), a post-processing called *rotation* is applied to improve the interpretability of $\boldsymbol{A}$. In the context of the spectral relaxation of k-means, Zha et al. (2001) performed a similar post-processing to obtain better cluster indicator. We derive a rotation for graph FA which considers optimalities of both of FA and kernel k-means. Let $\boldsymbol{Q} \in \mathbb{R}^{k \times k}$ be an orthogonal matrix. For example, *varimax* rotation (Kaiser, 1958) estimates $\boldsymbol{Q}$ by maximizing the variance of squared elements of the rotated matrix $\widehat{\boldsymbol{A}}\boldsymbol{Q}$, which improves interpretability because the elements of $\widehat{\boldsymbol{A}}\boldsymbol{Q}$ tend to be either large magnitude or close to 0. In this paper, we use the following form of transformation:

$$\widehat{\boldsymbol{A}}_{\text{rot}} := \widehat{\boldsymbol{A}}(\boldsymbol{\Lambda}_k - \boldsymbol{I})^{-1/2}\boldsymbol{Q}. \quad (8)$$

This transformation keeps optimality, shown by the following theorem:

Theorem 3. *The matrix* $\widehat{\boldsymbol{A}}_{\text{rot}}$ *in (8) is identical to an optimal solution of the spectral relaxation of weighted kernel k-means (6) up to a constant of each column vector, and it also keeps the optimality of the likelihood by modifying the factor distribution as* $\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}^\top(\boldsymbol{\Lambda}_k - \boldsymbol{I})\boldsymbol{Q})$.

See supplementary appendix E for the proof. The rotation (8) thus improves interpretability while keeping the both optimalities of kernel k-means and the likelihood by allowing the non-identity covariance of the factors.

4 Relations to Other Approaches

In this section, we describe relationships between our approach and other existing methods.

4.1 Graph based Regularization

A standard approach to incorporating a graph into machine learning algorithms is to use an additional penalty which makes parameters smooth on the given

graph (Li and Li, 2008; Sandler et al., 2008; Tibshirani and Taylor, 2011; Yang et al., 2012). Let $\boldsymbol{W}$ be an adjacency matrix of the graph $\mathcal{G}$ in which the (i,j) element is $W_{ij}=1$ if $(i,j)\in\mathcal{E}$, and $W_{ij}=0$ otherwise. Let $\boldsymbol{L}\in\mathbb{R}^{d\times d}$ be the *graph Laplacian matrix* defined as $\boldsymbol{L}:=\boldsymbol{D}_W-\boldsymbol{W}$, where $\boldsymbol{D}_W$ is a diagonal matrix in which i-th diagonal entry $\boldsymbol{D}_W$ is equal to $\sum_j W_{ij}$. The simplest way is to add trace$\boldsymbol{A}^\top\boldsymbol{L}\boldsymbol{A}$ as an additional penalty, by which rows of $\boldsymbol{A}$ are forced to have similar values if they are closely connected in the graph. However, this approach has the following two problems: 1) A regularization parameter to balance the effect of the graph Laplacian is usually necessary, but selecting an appropriate value is difficult. 2) Negative correlation is difficult to incorporate [1].

Note that the same idea is often used to control smoothness in terms of instances $\{1,\ldots,n\}$. For example, in graph-based semi-supervised learning methods (Zhu et al., 2003; Zhou et al., 2004; Belkin et al., 2006), each data instance $\boldsymbol{x}_i$ is regarded as a graph node, and the prediction is regularized to be smooth on the graph. On the other hand, our focus in this paper is on the graph which represents relationships between different dimensions of the feature vector $\{1,\ldots,d\}$, and thus these approaches including (Zheng et al., 2011; Jiang et al., 2013) are not directly applicable to our problem setting.

4.2 Graph Clustering

Another direction of research having potential relationships with our approach are *graph clustering* (Schaeffer, 2007) and *community/module detection in graph* (Newman, 2006). Given a (weighted) graph, these methods find subgraphs in which nodes are strongly connected to each other. *Spectral clustering* (Shi and Malik, 2000; Ng et al., 2001; Meila and Shi, 2001) can be considered as a method in this category, which is also based on graph Laplacian (Chung, 1997). One of the standard formulations of spectral clustering is the following spectral relaxation of *minimum cut* (see e.g., von Luxburg, 2007, for detail):

$$\min_{\boldsymbol{F}\in\mathbb{R}^{d\times k}} \quad \mathrm{trace}\left(\boldsymbol{F}\boldsymbol{L}\boldsymbol{F}\right) \qquad (9)$$
$$\text{s.t.} \quad \boldsymbol{F}^\top\boldsymbol{F}=\boldsymbol{I}.$$

This can also be interpreted as a dimensionality reduction of the graph into the k-dimensional space $\boldsymbol{F}$. To obtain the cluster assignment, for example the standard k-means clustering is applied to $\boldsymbol{F}$. Saerens et al. (2004) shows that the dimensionality reduction by spectral clustering, called *Laplacian eigenmap* (Belkin and Niyogi, 2003), can be interpreted as PCA in the space defined by a commute distance in the graph. Furthermore, if we regard the inverse matrix of the graph Laplacian matrix as a kernel matrix, spectral clustering can also be interpreted as weighted kernel k-means (Dhillon et al., 2004). Thus, spectral clustering is closely related to our approach. However, spectral clustering has the following difficulties for our purpose: 1) Negative correlation is again difficult to deal with. 2) Unlike the standard PCA under the Gaussian assumption, resulting eigenvectors are difficult to interpret in a sense of the original input space $\mathcal{X}$.

Graph FA avoids these difficulties based on the Gaussian model. *Stochastic block model* is also widely used for network analysis (e.g., Karrer and Newman, 2011), but it is not for providing interpretable representation of the input distribution.

5 Experiments

We evaluate graph FA using synthetic datasets and a gene expression dataset originally used by the research of breast cancer. We also use PCA instead of factor analysis (i.e., $\boldsymbol{\Psi}=\sigma^2\boldsymbol{I}$), which we call *graph PCA*. For comparison, we used the standard principal component analysis (PCA), factor analysis (FA), PCA regularized by graph Laplacian (Lap-PCA), and spectral clustering (SC). In this paper, Lap-PCA indicates a PCA regularized by the graph Laplacian matrix which is described in Section 4. To define the objective function of Lap-PCA, we used a similar technique to Jiang et al. (2013) (see supplementary appendix F for detail). All methods can produce k ($<d$) vectors (e.g., principal direction in PCA, and eigenvectors of the graph Laplacian matrix in SC) which are parsimonious representation of the data. We refer to those vectors in general as *basis vectors*. For rotation, we used varimax rotation (Kaiser, 1958). In Lap-PCA, we used the unnormalized graph Laplacian matrix, and gave weights to each edge by the standard Gaussian kernel for which the width parameter was determined by the median heuristics (Gretton et al., 2007). The standard SC often uses Gaussian kernel, but here we used the absolute value of covariances as the edge weights for SC.

For the estimation process of graph FA (and graph PCA) we can utilize existing optimization methods for graphical Gaussian model and factor analysis (For the GGM step, we can also use graphical lasso when the graph is not available, and the properties that we discussed still hold in that case as well). For the optimization of FA, we need an initial value of $\boldsymbol{\Psi}$ (once we

[1] Although some methods use the signed graph (Goldberg et al., 2007; Wu et al., 2011), then negative weights represent dissimilarity between two nodes. We would like to regard negative correlation as a possible interaction in a graph (e.g., inhibition).

fix $\boldsymbol{\Psi}$, we can obtain $\boldsymbol{A}$). We first scaled a covariance matrix into a correlation matrix for numerical stability (then ψ_i is in $(0, 1]$), which can be recovered after the optimization, and simply set $\boldsymbol{\Psi} = 0.1\boldsymbol{I}$ as an initial value. A simple heuristics here is that we chose a relatively small value for $\boldsymbol{\Psi}$ because we would like $\boldsymbol{A}$ to explain covariance as far as possible.

5.1 Synthetic Dataset

We first use synthetic datasets. The graph is generated by the three network models called *lattice*, *Watts-Strogatz*, and *Barabási-Albert* (Cohen and Havlin, 2010). In the lattice model, a set of nodes is located on grid points (we used 2 dimensional grid), and each node simply has links to its nearest neighbors (4 neighbors in the case of the 2 dimensional grid). The Watts-Strogatz model is generated by randomly rewiring a 1 dimensional lattice graph with a certain probability (we used 0.1). The Barabási-Albert model generates a graph having the so-called *scale free* property which often appears in real world networks (we set both of the number of initial nodes and the number of edges of new nodes as 3 in the generative process of this network). We set the number of nodes of each graph as $d = 100$.

We generated $n = 100$ instances of the input $\{\boldsymbol{x}_i\}_{i=1}^{100}$ using a Gaussian distribution in which large covariances concentrating on randomly chosen connected subgraphs. To choose the subgraphs, we first divided d features into 5 disjoint groups having the same number of features, and then defined the maximum connected component in each group as the subgraph. Defining the 5 subsets of nodes as $\{\mathcal{S}_i\}_{i=1}^5$, where $\mathcal{S}_i \subset \{1, \ldots, d\}$ is an index set of each connected component. Based on this grouping, we first generated a covariance matrix $\boldsymbol{\Sigma}_0 = \boldsymbol{A}\boldsymbol{A}^\top + \boldsymbol{\Psi} + \boldsymbol{E}$, where $\boldsymbol{A} \in \mathbb{R}^{d\times 5}$ is a loading matrix, $\boldsymbol{\Psi}$ is diagonal matrix, and $\boldsymbol{E} \in \mathbb{R}^{n\times n}$ is a noise term. These three variables are defined as follows: $\boldsymbol{A}_{ij} \sim \mathcal{N}(0, 1)$ for $i \in \mathcal{S}_j$, and 0 otherwise, $\boldsymbol{\Psi}_{ii} \sim \Gamma(5, 0.1)$, and $\boldsymbol{E} = 0.1\tilde{\boldsymbol{E}}\tilde{\boldsymbol{E}}^\top/d$, where each element of $\tilde{\boldsymbol{E}} \in \mathbb{R}^{d\times d}$ is generated by $\mathcal{N}(0, 1)$. A true covariance matrix for synthetic data is then defined by the solution of (1) with $\boldsymbol{S} = \boldsymbol{\Sigma}_0$, by which the graph connectivity is embedded as the underlying dependency. We used $k = 5$ for all methods, which is the true value for the number of the activated subgraphs. Results are the average of the 30 trials.

We first compared KL divergence between estimated covariance by each method and $\boldsymbol{\Sigma}_{\mathcal{G}}$. Table 1 shows the results. Graph FA has the best values for all the three networks. Graph FA and graph PCA were more accurate compared with FA and PCA, respectively. This confirms that giving dependency structure improves estimation accuracy. On the other hand, graph FA and FA have better results compared with graph PCA and PCA, respectively. We see that absorbing differences of individual variances also improves accuracy. Overall, these results are not surprising because graph FA and graph PCA have true dependency as the graph, but by combining subsequent results on activated subgraph identification, we see that our basis vectors is accurate in both senses of low dimensional representations of Gaussian and activated subgraph indicators, which can not be realized simultaneously by other methods.

Table 1: Comparisons of KL divergence with the true covariance for the synthetic datasets. We use the function (1) which does not contain constant terms. The best method and comparable methods according to the t-test at the significance level of 5% against the best method are specified by boldface. The left-most column shows network models (L: Lattice, WS: Watts-Strogatz, and BA: Barabási-Albert).

	Graph PCA	Graph FA	PCA	FA	Lap-PCA
L	120.64	**102.43**	154.29	108.01	125.58
	(6.88)	**(5.94)**	(9.20)	(6.19)	(7.12)
WS	119.70	**102.11**	153.32	107.53	124.79
	(8.96)	**(7.49)**	(11.94)	(7.55)	(9.01)
BA	80.73	**68.77**	106.30	73.73	85.50
	(7.50)	**(5.46)**	(12.23)	(5.40)	(7.55)

Next, we compared how accurately each method captures the coordinated variations embedded in the subgraphs. In this case, ideally, the basis vectors obtained by each method should have large absolute values for the indices included in connected component $\{S_i\}_{i=1}^5$. Regarding the absolute values of each one of basis vectors as indicators of one of $\{S_i\}_{i=1}^5$, we calculated *area under the curve* (AUC). Since there exist 5! possible assignment patterns between the basis vectors and $\{S_i\}_{i=1}^5$, we chose the best AUC values among those possible patterns because the assignment between the basis vectors and $\{S_i\}_{i=1}^5$ is arbitrary. Only for SC, instead of rotation, we applied k-means to the basis vector, because the basis vector of SC is not suitable to the above procedure. We calculated AUC for SC by sorting nodes based on distances from the closest centroid. We here further added noise term to the ground truth covariance $\boldsymbol{\Sigma}_{noisy} := \boldsymbol{\Sigma}_{\mathcal{G}}/\|\boldsymbol{\Sigma}_{\mathcal{G}}\|_F + 0.1\boldsymbol{E}/\|\boldsymbol{E}\|_F$. Our purpose is to see whether our approach still identifies activated subgraphs in this contaminated situation.

Table 2 shows the results. For all methods except for SC, we provide the results of (a) after rotation and (b) before rotation. First we focus on the rotated cases. For all graph types, graph FA achieved the best AUC values being followed by Graph PCA. For the Barabási-Albert model, the AUC of graph FA was

Table 2: The average best AUCs and their standard deviations for identifying activated subgraphs in the synthetic datasets. The best method and comparable methods according to the t-test at the significance level of 5% against the best method are specified by boldface. The left-most column shows network models (L: Lattice, WS: Watts-Strogatz, and BA: Barabási-Albert).

	Graph PCA		Graph FA		PCA		FA		Lap-PCA		SC
	(a)	(b)	(a)	(b)	(a)	(b)	(a)	(b)	(a)	(b)	
L	.80 (.06)	.76 (.07)	**.85 (.05)**	.83 (.05)	.70 (.05)	.65 (.04)	.72 (.04)	.67 (.04)	.69 (.05)	.65 (.04)	.77 (.03)
WS	.80 (.06)	.75 (.07)	**.85 (.06)**	.81 (.05)	.70 (.05)	.66 (.04)	.72 (.05)	.67 (.03)	.70 (.05)	.66 (.04)	.77 (.04)
BA	**.85 (.07)**	.83 (.06)	**.85 (.07)**	.83 (.07)	.80 (.06)	.79 (.05)	.79 (.06)	.79 (.06)	.79 (.06)	.79 (.05)	.56 (.06)

still the best, and that of graph PCA was comparable to graph FA. The other methods, i.e., PCA, FA, Lap-PCA, and SC, could not capture the activated subgraphs compared to our approaches. In this experiments, the rotated results has higher or the same AUC values before rotation.

Figure 2 shows illustrative examples of graph FA (rotated), FA (rotated), and SC using a toy network with $k = 3$. The covariance $\mathbf{\Sigma}_{noisy}$ is used. Graph FA clearly indicates covariance and connectivity simultaneously compared to FA. It is difficult to interpret covariance structure from SC because of its difficulty in distributional interpretations (note that all the elements of the first eigenvector is 1).

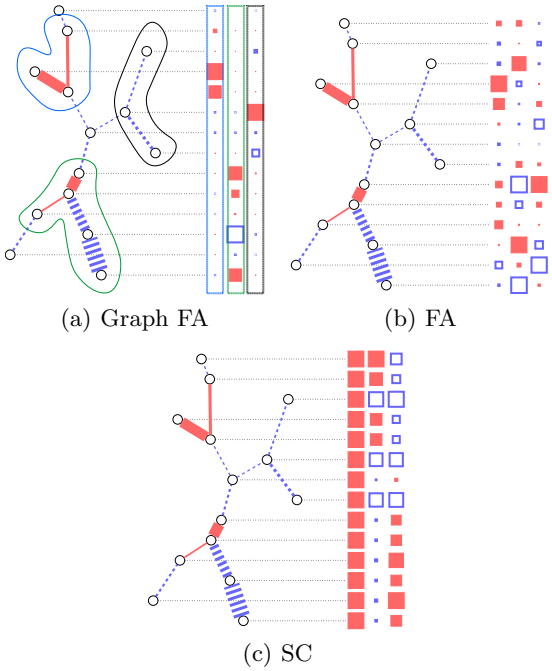
5.2 Analyzing Protein Network and Gene Expression

Next we show the effectiveness of our approach by using protein-protein interaction network (PPI), pathway networks and gene expression data. For better understanding of the role of genes (and corresponding proteins), investigating their behavior with a network is quite important for recent biological data analysis. We used a gene expression data from the study of breast cancer (van de Vijver and et al., 2002). The data contains $n = 295$ breast cancer patients in which 78 patients have distant metastasis within five years. We retrieved the PPI network from *Pathway Commons* database (Cerami and et al., 2011) regarding *Homo Sapiens* proteins, and used only the pairs annotated as "`INTERACTS_WITH`", and we further combined another graph created by *KEGG* pathway (Kanehisa and Goto, 2000) through R package *graphite* (Sales et al., 2012) for which we used the edges annotated as "`activation`" or "`inhibition`". We first chose 2000 genes most correlated with metastasis, and then extracted the largest connected components in the corresponding PPI networks. As a result, the PPI network has $d = 1829$ features and 25187 edges. The experiments were run 10 times using randomly sampled 90% of instances.

Our aim here is to evaluate how biologically meaningful subgraphs can be identified by each method. In particular we evaluate the case that each method outputs subgraphs with similar sizes because it is difficult to compare the significance between subgraphs with different sizes. To extract the subgraph with a similar size, for each basis vector, we first sorted the elements of the basis vector in the descending order of the absolute values, and then, from the top of the sorted elements, we found the smallest subset of features in which the maximum connected component contains at least 50 nodes. Only for SC, we used k-means algorithm as we did in the previous synthetic experiment, and sorted the indices based on the distance from the closest centroid. The detected maximum connected component is defined as the subgraph for each basis vector. To compare different sets of genes, we used *gene ontology* (GO) (Ashburner, 2000) term enrichment analysis. GO is a current standard for annotating genes (or gene products, i.e. proteins), providing terms which specify gene products' molecular function, biological process, and localization to cellular components. For each gene, we can search annotated terms in GO, and we count how often each term appears in the given set of genes. The statistical significance of the terms can be evaluated by probabilities that the terms are counted by chance using *hypergeometric test*.

The results are shown in Figure 3, in which the horizontal axis is the negative log p-values (adjusted by Bonferroni correction) and the vertical axis is the number of enriched (significant) GO terms. We only counted GO terms sequentially appearing on the connected nodes in the graph. Even if the same GO term appears at two nodes, they are not counted twice when that node pair is not directly connected. This means that nodes in a connected subgraph have to be related to each other through the connection of the edges. We here used $k \in \{5, 10, 15, 20\}$ for all compared methods.

We first focus on the results obtained after rotation (the solid lines). Graph FA found the largest number of GO terms in all results in Figure 3 (except only for $-\log_{10}(p)$ of around 3 of $k = 10$, where graph PCA had the largest number). This suggests that a subgraph found by graph FA can realize statistically significant biological functions through the connections on the graph because the larger number of significant



(a) Graph FA (b) FA

(c) SC

Figure 2: Illustrative examples using a toy graph. The input data was randomly generated by a Gaussian distribution in which the covariance is defined through the graph and additional noise (see main text for detail). The boxes in the right side represent basis vectors, in which size of boxes indicates the absolute values of elements (filled: positive, and unfilled: negative). The width of edges represent the amount of covariances and the types of lines represent sign (solid: positive, and dashed: negative). The basis vectors of graph FA provides highly interpretable results (nodes surrounded by the lines indicate a set of dimensions having higher absolute values in the basis vector).

GO terms are shared by neighbouring nodes compared to the other methods. The differences between graph FA and graph PCA indicate that differences of single variances in the models can bring largely different results.

The rotation process increased the number of terms for graph FA. Since the identification condition of the original $\boldsymbol{A}$ is just for the computational reason, these results are also reasonable.

6 Conclusion

We proposed *Graph FA: factor analysis on graph* for analyzing coordinated variations of numerical features on the graph. We showed that this approach has the desirable interpretability due to the following two links:

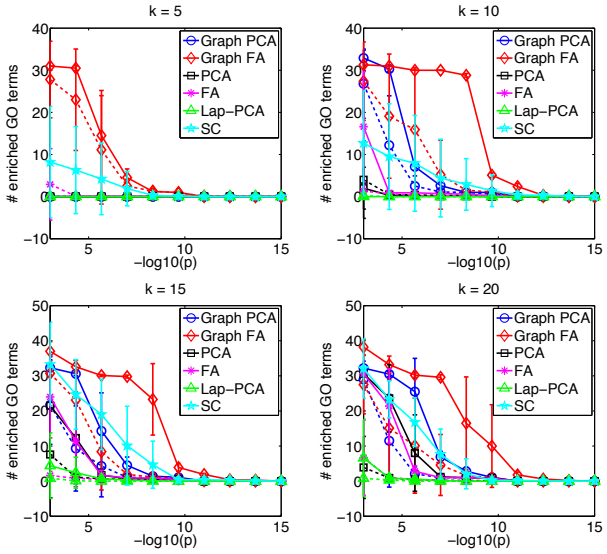


Figure 3: The number of enriched GO terms (appearing sequentially on neighboring nodes) as a function of cutoff values of p-values. The bar of each point represents the standard deviation. The solid lines are with rotation, and the dashed lines are without rotation.

(1) Gaussian graphical model and a path-based kernel

(2) Factor analysis and weighted kernel k-means

Although the graph FA procedure itself is quite simple, to the best of our knowledge, this approach has not been considered in depth and the above relation has not been recognized. The experiments on synthetic and gene expression data demonstrated the effectiveness of the approach.

7 Acknowledgements

This work has been supported in part by MEXT KAKENHI 16H02868 and 17H04694, JST ACCEL, Collaborative Research Program of ICR, Kyoto University (grant #2017-27). Tekes: FiDiPro, Academy of Finland: AIPSE programme.

References

M. Ashburner. Gene ontology: Tool for the unification of biology. *Nature Genetics*, 25:25–29, 2000.

M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.

M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

D. M. Boyd and N. B. Ellison. Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 13(1):210–230, 2007.

E. G. Cerami and et al. Pathway commons, a web resource for biological pathway data. *Nucleic Acids Research*, 39:685–690, 2011.

F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.

R. Cohen and S. Havlin. *Complex networks: Structure, Robustness and Function*. Cambridge University Press, 2010.

I. S. Dhillon, Y. Guan, and B. Kulis. Kernel k-means: Spectral clustering and normalized cuts. In *Proc. of the 10th ACM SIGKDD*, pages 551–556. ACM, 2004.

A. B. Goldberg, X. Zhu, and S. J. Wright. Dissimilarity in graph-based semi-supervised classification. In M. Meila and X. Shen, editors, *Proceedings of the 11th International Conference on Artificial Intelligence and Statistics*, volume 2, pages 155–162. JMLR.org, 2007.

A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. J. Smola. A kernel method for the two-sample-problem. In *Advances in NIPS 19*, pages 513–520. MIT Press, 2007.

H. H. Harman. *Modern Factor Analysis*. The university of chicago press, 1960.

T. Ideker, O. Ozier, B. Schwikowski, and A. F. Siegel. Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics*, 18 (suppl 1):S233–S240, 2002.

B. Jiang, C. Ding, B. Luo, and J. Tang. Graph-Laplacian PCA: Closed-form solution and robustness. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3492–3498, 2013.

I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, 2002.

B. Jones and M. West. Covariance decomposition in undirected gaussian graphical models. *Biometrika*, 92(4):779–786, 2005.

H. F. Kaiser. The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, 23(3):187–200, 1958.

M. Kanehisa and S. Goto. KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 28 (1):27–30, 2000.

B. Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. *PHYSICAL REVIEW E*, 83:016107, Jan 2011.

R. Kosala and H. Blockeel. Web mining research: A survey. *SIGKDD Explorations Newsletter*, 2(1):1–15, June 2000.

S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

C. Li and H. Li. Network-constrained regularization and variable selection for analysis of genomic data. *Bioinformatics*, 24(9):1175–1182, 2008.

M. Meila and J. Shi. A random walks view of spectral segmentation. In *Proceedings of the 8th International Workshop on Artifical Intelligence and Statistics*. Morgan Kaufmann, 2001.

M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006.

A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in NIPS 14*, pages 849–856. MIT Press, 2001.

J. S.-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge university press, 2004.

M. Saerens, F. Fouss, L. Yen, and P. Dupont. The principal components analysis of a graph, and its relationships to spectral clustering. In *Proc. of the 15th ECML*, volume 3201 of *LNCS*, pages 371–383. Springer Berlin Heidelberg, 2004.

G. Sales, E. Calura, D. Cavalieri, and C. Romualdi. graphite - a bioconductor package to convert pathway topology to gene network. *BMC Bioinformatics*, 13(1):20, 2012.

T. Sandler, J. Blitzer, P. P. Talukdar, and L. H. Ungar. Regularized learning with networks of features. In *Advances in NIPS 21*, pages 1401–1408, 2008.

S. E. Schaeffer. Survey: Graph clustering. *Computer Science Review*, 1(1):27–64, Aug. 2007.

J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

A. J. Smola and I. R. Kondor. Kernels and regularization on graphs. In *Proceedings of the Annual Conference on Computational Learning Theory*, pages 144–158, 2003.

R. Tibshirani and J. Taylor. The solution path of the generalized lasso. *The Annals of Statistics*, 39(3): 1335–1371, 2011.

M. J. van de Vijver and et al. A gene-expression signature as a predictor of survival in breast cancer. *New England Journal of Medicine*, 347(25):1999–2009, 2002.

U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

J. Whittaker. *Graphical Models in Applied Multivariate Statistics*. Wiley Publishing, 1990.

L. Wu, X. Ying, X. Wu, A. Lu, and Z.-H. Zhou. Spectral analysis of k-balanced signed graphs. In *Advances in Knowledge Discovery and Data Mining - 15th Pacific-Asia Conference (PAKDD)*, volume 6635 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2011.

S. Yang, L. Yuan, Y.-C. Lai, X. Shen, P. Wonka, and J. Ye. Feature grouping and selection over an undirected graph. In *Proc. of the 18th ACM SIGKDD*, pages 922–930. ACM, 2012.

H. Zha, X. He, C. Ding, H. Simon, and M. Gu. Spectral relaxation for k-means clustering. In *Advances in NIPS 14*, pages 1057–1064. MIT Press, 2001.

M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, and D. Cai. Graph regularized sparse coding for image representation. *Image Processing, IEEE Transactions on*, 20(5):1327–1336, 2011.

D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems (NIPS) 16*. MIT Press, 2004.

X. Zhu, Z. Ghahramani, and J. D. Lafferty. Semi-supervised learning using Gaussian fields and harmonic functions. In T. Fawcett and N. Mishra, editors, *Proceedings of the 20th International Conference on Machine Learning (ICML)*, pages 912–919. AAAI Press, 2003.