Principal Components Analysis of Discrete Datasets

Yifan Zhu

December 1, 2018

Abstract

We propose a Gaussian copula based method to perform principal component analysis for discrete data. By assuming the data are from a discrete distributions in the Gaussian copula family, we can consider the discrete random vectors are generated from a latent multivariate normal random vector. So we first obtain an estimate of the correlation matrix of latent multivariate normal distribution, then we use the estimated latent correlation matrix to get the estimates of principal components. We also focus on the case when we have categorical sequence data with multinomial marginal distribution. In this case the marginal distribution is not univariate and thus the usual Gaussian copula does not fit here. The optimal mapping method is proposed to convert the original data with multivariate discrete marginals to the mapped data with univariate marginals. Then the usual Gaussian copula can be used to model the mapped data, and we apply the discrete principal component analysis to the mapped data. The senators' voting data was used in the experiment as an example. Finally, we also propose a matrix Gaussian copula method to deal with data with multivariate marginals. It can be considered as an extension of Gaussian copula, and we use the latent correlation matrix in the matrix Gaussian copula to obtain the principal components.

1 Introduction

For high dimensional data of dimension p, a very large number p of variables is measured on each sample unit, and interpreting results of analyses might be difficult. So for high dimensional data, dimension reduction is usually performed prior to the analysis of the data. Principal component analysis (PCA) is one of the dimension reduction methods. It can be considered to find a linear subspace such that the distance between the samples and the linear subspace is minimized. The linear subspace should have a dimension that is significantly less than p, and then the dimension of the data can be reduced significantly by projecting every sample to the linear subspace. The basis of the subspace is known as principal components (PCs), and the projection of a sample projected to a PC is called the score for that PC. Therefore scores can be considered as a lower dimensional representation of the original data. So scores can be used as response variables to fit MANOVA or regression models, to cluster sample units or to build classification rules.

Since PCA is only a linear dimension reduction method, it may not produce a proper lower dimensional representation of the original data when the data are on a non-linear manifolds in the high dimensional space. In this case, principal manifolds (Gorban et al. (2008)) generalized the linear manifold of PCA by explicitly constructing an embedded manifold for data approximation. Another problem with PCA is that it is sensitive to outliers. In some cases, the outliers are difficult to identify. For example, in data mining algorithms like correlation clustering, the assignment of points to clusters and outliers is not known beforehand. To solve this problem, Kriegel et al. (2008) proposed generalization of PCA based on a weighted PCA, which increases robustness by assigning different weights to data objects based on their estimated relevancy. Markopoulos et al. (2014) also proposed an outlier-resistant version of PCA formulations. Another issue with PCA is that it is not scale-invariant, i.e. changing the measurement scale of variables makes the estimates different (Borgognone et al. (2001)). Also, data are usually assumed to be Gaussian or sub-Gaussian distributed such that a fast convergence rate can be obtained. In addition, PCA gives the consistent

estimator of principal components only when the dimension d is fixed (Anderson et al. (1958)). Under a double asymptotic framework in which both the sample size n and dimensionality d can increase (with possibly $d > n$), PCA does not achieve a consistent estimator of principal components. Johnstone and Lu (2009) showed that the leading eigenvector of sample covariance or correlation matrix cannot converge to the true leading eigenvector. Han and Liu (2014) proposed a high dimensional semi-parametric scale-invariant principal component analysis method copula PCA (COCA) to solve these problems. COCA is scale invariant and its estimating procedure is adaptive over the whole nonparanormal family, which contains and is much larger than the Gaussian. It is also robust to modeling and data contamination, and can be consistent even when the dimensionality is nearly exponentially large relative to the sample size. However, nonparanormal family only contains continuous distributions. So COCA does not work when the data are discrete or categorical.

In the practical, many data sets are collection of vectors of integers, non-negative counts , binary values or categorical values. For example, the count of a particular word appearing in a document is always a positive integer; The vote of a senator can only take three values: "Yes", "No" or "Absent"; The response of a survey for people's emotion might take ordinal categorical values, like "Very Unhappy", "Unhappy", "OK", "Happy" and "Very Happy"; The DNA sequence only takes 4 values: A, T, C, G. Recall that PCA's assumption is that the samples are in a Euclidean space, and the dimension reduced data are on the linear manifold in the high dimensional space. So apparently for discrete data these assumptions are not reasonable. Therefore the general PCA does not apply in finding the principal components of discrete data. Many methods has been proposed for dimension reduction of discrete data by finding adequate principal components : PCA for exponential family (Collins et al. (2002)), grade of membership (GOM) (Woodbury and Manton (1982)), probabilistic latent semantic indexing (PLSI) (Hofmann (2017)), non-negative matrix factorization (NMF) (Lee and Seung (1999)), genotype inference using admixtures (Pritchard et al. (2000)), latent Dirichlet allocation (LDA) (Blei et al. (2003)) , Gamma-Poisson models (GaP) (Canny (2004)), multinomial PCA (MPCA) (Buntine (2002)) and discrete component analysis (DPCA) (Buntine and Jakulin (2006)).

In this paper, our goal is to reduce the dimension of discrete or categorical data by some underlying principal components. However, we cannot apply PCA directly on the this kind of data. As is discussed above, the sample space only has finite points, so we cannot find linear manifold expanded by a proper set of basis (or principal components, PC) for this space like what we did for continuous distribution. Hence we consider that these data are generated from some latent variables with continuous distribution. Then instead of performing PCA on the observed data, we perform PCA on the latent variables. We also want to transform or connect the latent variables to multivariate normal random vectors. Since the normal distribution is completely determined by its first and second moments, and thus for dimension reduction, PCA is an optimal method for multivariate normal data.

The rest of the paper is organized as follows. In section 2, we review the coupla PCA method proposed by Han and Liu (2014); in section 3, we extend the copula PCA method to the discrete case. The data recovery with principal components is also discussed; in section 4, we focus on data with multinomial marginals, and introduce optimal mapping method to obtain a mapped data with univariate discrete marginals; in section 5, we apply the optimal mapping method and discrete copula PCA to analyze the data set; in section 6, we introduce an extension of Gaussian copula, called matrix Gaussian copula.And extend the discrete Gaussian copula PCA to discrete matrix Gaussian copula PCA to handle the data where each sample is a matrix.

2 The copula PCA

Han and Liu (2014) proposed a method called copula PCA for data from a nonparanormal distribution. This family of distribution is continuous and is actually the same as the continuous Gaussian copula family. By assuming nonparanormal distribution, the data can be seen to be generated

through a multivariate normal distribution, and we can do PCA with the correlation matrix of that latent multivariate normal random variable.

2.1 The nonparanormal distribution

Definition 2.1 (Han and Liu (2014)). Let $f^0 = \{f_j^0\}_{j=1}^d$ be a set of strictly increasing univariate functions. We say that a d dimensional random vector $\boldsymbol{X} = (X_1, \ldots, X_d)^T$ follows a nonparanormal distribution $\mathrm{NPN}_d(\boldsymbol{\Sigma}_0, f^0)$, if

$$f^0(\boldsymbol{X}) := (f_1^0(X_1), \ldots, f_d^0(X_d))^T \sim \mathrm{N}_d(\mathbf{0}, \boldsymbol{\Sigma}^0)$$

where $(\boldsymbol{\Sigma}^0)$ is a correlation matrix.

From the definition, the nonparanormal distribution must be a continuous distribution. And it is actually a special case of Gaussian copula family for continuous distributions.

Suppose we have $\boldsymbol{X} \sim \mathrm{NPN}_d(\boldsymbol{\Sigma}^0, f^0)$, then the marginal cumulative distribution functions (cdf) are

$$F_j(x_j) = P(X_j \le x_j) = P(f_j^0(X_j) \le f_j^0(x_j)) = \Phi(f_j^0(x_j))$$

And the joint cdf is

$$\begin{aligned}
F(x_1, x_2, \ldots, x_d) &= P(X_1 \le x_1, \ldots, X_d \le x_d) \\
&= P(f_1^0(X_1) \le f_1^0(x_1), \ldots, f_d^0(X_d) \le f_d^0(x_d)) \\
&= \Phi_{\boldsymbol{\Sigma}^0}(f_1^0(x_1), \ldots, f_d^0(x_d))) \\
&= \Phi_{\boldsymbol{\Sigma}^0}(\Phi^{-1}(\Phi(f_1^0(x_1))), \ldots, \Phi^{-1}(\Phi(f_d^0(x_d)))) \\
&= \Phi_{\boldsymbol{\Sigma}^0}(\Phi^{-1}(F_1(x_1)), \ldots, \Phi^{-1}(F_d(x_d))) \\
&= C_{\boldsymbol{\Sigma}^0}(F_1(x_1), \ldots, F_d(x_d))
\end{aligned}$$

where $\Phi_{\boldsymbol{\Sigma}^0}$ is cdf of $\mathrm{N}_d(\mathbf{0}, \boldsymbol{\Sigma}^0)$ and $C_{\boldsymbol{\Sigma}^0}$ is a Gaussian copula with correlation matrix $\boldsymbol{\Sigma}^0$. So $\boldsymbol{X} \sim \mathrm{NPN}_d(\boldsymbol{\Sigma}^0, f^0)$ actually means $(F_1(X_1), \ldots, F_d(X_d))^T \sim C_{\boldsymbol{\Sigma}^0}$ or $(\Phi^{-1}(F_1(X_1)), \ldots, \Phi^{-1}(F_d(X_d)))^T \sim \mathrm{N}(\mathbf{0}, \boldsymbol{\Sigma}^0)$.

2.2 Estimation of $\boldsymbol{\Sigma}^0$

For nonparanormal distribution, we already know $F_j(x_j) = \Phi(f_j^0(x_j))$, then one natural way is to first estimate $f_j^0(x_j) = \Phi^{-1}(F_j(x_j))$, and then use the estimated $\hat{f}_j^0$ to transform the data to multivariate normal and estimate $\boldsymbol{\Sigma}^0$ as if we have data from a multivariate normal distribution. Suppose we have n observations $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ from nonparanormal distribution, and $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{id})^T$, then Liu et al. (2012) gave an estimation of f_j^0 by

$$\hat{f}_j^0(t) = \Phi^{-1}\left(T_{\delta_n}[\hat{F}_j(t)]\right)$$

where T_{δ_n} is a Winsorization (or truncation) operator defined as $T_{\delta_n}(x) = \delta_n I(x < \delta_n) + xI(\delta_n \le x \le 1 - \delta_n) + (1 - \delta_n)I(x > 1 - \delta_n)$ with $\delta_n = 1/(4n^{1/4}\sqrt{\pi \log n})$. And $\hat{F}_j(t) = \frac{1}{n+1}\sum_{i=1}^n I(x_{ij} \le t)$ is the scaled empirical cdf of X_j. Then

$$\hat{\boldsymbol{\Sigma}}_{jk}^0 = \frac{\frac{1}{n}\sum_{i=1}^n \hat{f}_j^0(x_{ij})\hat{f}_k^0(x_{ik})}{\sqrt{\frac{1}{n}\sum_{i=1}^n \left(\hat{f}_j^0(x_{ij})\right)^2}\sqrt{\frac{1}{n}\sum_{i=1}^n \left(\hat{f}_k^0(x_{ik})\right)^2}}, \quad \hat{\boldsymbol{\Sigma}}^0 = [\hat{\boldsymbol{\Sigma}}_{jk}^0]$$

Another way to estimate $\boldsymbol{\Sigma}^0$ directly without estimating f_j^0 by Liu et al. (2012) uses Spearman's ρ and Kendall's τ statistics. Let r_{ij} be the rank of x_{ij} among $x_{1j}, \ldots, x_{nj}$ and $\bar{r}_j = \frac{1}{n}\sum_{i=1}^n r_{ij}$,

then

$$\hat{\rho}_{jk} = \frac{\sum_{i=1}^{n}(r_{ij} - \bar{r}_j)(r_{ik} - \bar{r}_k)}{\sqrt{\sum_{i=1}^{n}(r_{ij} - \bar{r}_j)^2 \cdot \sum_{i=1}^{n}(r_{ik} - \bar{r}_k)}}$$
$$\hat{\tau}_{jk} = \frac{2}{n(n-1)} \sum_{1 \le i < i' \le n} \text{sign}\,(x_{ij} - x_{i'j})\,(x_{ik} - x_{i'k})$$

For a random vector $\boldsymbol{X} = (X_1, X_2, \ldots, X_p)^T$, The population version of Spearman's ρ and Kendall's τ are

$$\rho_{jk} = \text{Corr}(F_j(X_j), F_k(X_k)),\ \tau_{jk} = P\left((X_j - \tilde{X}_j)(X_k - \tilde{X}_k) > 0\right) - P\left((X_j - \tilde{X}_j)(X_k - \tilde{X}_k) < 0\right)$$

where $(\tilde{X}_j, \tilde{X}_k)$ is a independent copy of (X_j, X_k).

Lemma 2.1 (Liu et al. (2012)). *Assuming* $\boldsymbol{X} \sim \text{NPN}(\boldsymbol{\Sigma}^0, f^0)$, *we have*

$$\boldsymbol{\Sigma}^0_{jk} = 2\sin\left(\frac{\pi}{6}\rho_{jk}\right) = \sin\left(\frac{\pi}{2}\tau_{jk}\right)$$

By Lemma 2.1, we estimate $\boldsymbol{\Sigma}^0$ by

$$\hat{\boldsymbol{\Sigma}}^0_{jk} = 2\sin\left(\frac{\pi}{6}\hat{\rho}_{jk}\right)$$

or

$$\hat{\boldsymbol{\Sigma}}^0_{jk} = \sin\left(\frac{\pi}{2}\hat{\tau}_{jk}\right)$$

For $\hat{\boldsymbol{\Sigma}}^0 = \left[2\sin\left(\frac{\pi}{6}\hat{\rho}_{jk}\right)\right]$, Han and Liu (2014) also showed when $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n \overset{\text{iid}}{\sim} \text{NPN}_d(\boldsymbol{\Sigma}^0, f^0)$, for any $n > 21/\log d + 2$,

$$P\left(\|\hat{\boldsymbol{\Sigma}}^0 - \boldsymbol{\Sigma}^0\|_{\max} \le 8\pi\sqrt{\frac{\log d}{n}}\right) \ge 1 - \frac{2}{d^2}$$

where $\|\boldsymbol{M}\|_{\max} = \max\{|\boldsymbol{M}_{ij}|\}$.

After obtaining an estimate of $\boldsymbol{\Sigma}^0$, we can use this estimated correlation matrix to perform PCA. Based on the assumption that the data are from a nonparanormal family, i.e. the data are generated through a multivariate normal random variable after a transformation, the dimension reduction would be optimal with the latent correlation matrix $\boldsymbol{\Sigma}^0$.

3 Copula PCA for discrete data

3.1 The generalized distributional transform and continuous latent variables

Now we want to study the case when our data are discrete. The main idea still to estimate the latent correlation matrix of the latent multivariate normal distribution. Therefore we now assume our data are from a discrete Gaussian copula family. Note that in estimating the underlying copula in the continuous case, we rely on the fact that by distributional transformation, the marginal distribution is transformed to be Uniform(0,1). But for discrete marginal distributions, the usual distributional transformation does not transform the distribution to Uniform(0,1).It only changes the sample space, while the marginal distribution remains the same. So to get an underlying copula in the discrete case, we introduce the generalized distributional transformation.

Definition 3.1 (Generalized distributional transformation, Rüschendorf (2013), Chapter 1)**.** Let Y be a real random variable with distributional function F and let V be a random variable independent of Y, such that $V \sim \text{Uniform}(0,1)$. The generalized distributional function is defined by

$$F(x,\lambda) = P(Y < x) + \lambda P(Y = x) = F(x-) + (F(x) - F(x-))\lambda$$

and we call

$$U = F(Y,V)$$

the generalized distributional transform of Y.

Theorem 3.1 (Rüschendorf (2013), Chapter 1)**.** *Let $U = F(Y,V)$ be the distributional transform of Y as defined above, then*

$$U \sim \text{Uniform}(0,1) \text{ and } Y = F^{-1}(U) \text{ a.s.}$$

where

$$F^{-1}(t) = \inf\{x \in \mathbb{R} : F(x) \geq t\}$$

is the generalized inverse of F, or the quantile transform of F.

Proof. We first prove $Y = F^{-1}(U)$ a.s..

Let $F(x)$ be the cdf of Y, and $U = F(Y,V)$ be the generalized distributional transformation of Y. For $\omega \in \Omega$ such that $Y(\omega)$ is a point of continuity of F, the generalized transformation is just $U(\omega) = F(Y(\omega), V(\omega)) = F(Y(\omega))$. Thus $F^{-1}(U(\omega)) = Y(\omega)$.

For ω when $Y(\omega)$ is not a point of continuity, we can take those ω s.t. $V(\omega) > 0$ among them and we have $P(\{\omega : V(\omega) = 0\}) = 0$. Since $0 < V(\omega) \leq 1$, we have

$$U(\omega) = F(Y(\omega)-) + (F(Y(\omega)) - F(Y(\omega)-))V(\omega) \leq F(Y(\omega)-) + F(Y(\omega)) - F(Y(\omega)-) = F(Y(\omega)).$$

Since $F^{-1}(U(\omega)) = \inf\{x : F(x) \geq U(\omega)\}$, therefore $Y(\omega) \geq F^{-1}(U(\omega))$. On the other hand, for a small $\epsilon > 0$, with $V(\omega) > 0$, we have

$$F(Y(\omega) - \epsilon)) \leq F(Y(\omega)-) < F(Y(\omega)-) + P(Y(\omega))V(\omega) = U(\omega).$$

Hence $Y(\omega) \leq F^{-1}(U(\omega))$. So $F^{-1}(U) = Y$ a.s.

Next we prove $U \sim \text{Uniform}(0,1)$.

Let $x_u = F^{-1}(u)$, then when $X < x_u$, $F(X,V) \leq F(X) < u$. Therefore

$$\begin{aligned}\{U \leq u\} &= \{F(X-) + (F(X) - F(X-))V \leq u\} \\ &= \{X < x_u\} \cup (\{X = x_u\} \cap \{F(x_u-) + (F(x_u) - F(x_u-))V \leq u\})\end{aligned}$$

Hence

$$\begin{aligned}P(U \leq u) &= P(X < x_u) + P(X = x_u)P(F(x_u-) + (F(x_u) - F(x_u-))V \leq u) \\ &= F(x_u-) + P(x_u)\frac{u - F(x_u)}{P(x_u)} = u\end{aligned}$$

Thus $U \sim \text{Uniform}(0,1)$ □

Now suppose that $\boldsymbol{X}$ is from a discrete distribution of the Gaussian copula family. Then

$$F(x_1, \ldots, x_d) = C_{\boldsymbol{\Sigma}^0}(F_1(x_1), \ldots, F_d(x_d)),$$

and with $U_i = F(X_i, V_i)$, by Theorem 3.1 and the definition of quantile transform, we also know that

$$\begin{aligned}F(x_1, \ldots, x_d) &= P(X_1 \leq x_1, \ldots, X_d \leq x_d) = P(F_1^{-1}(U_1) \leq x_1, \ldots, F_d^{-1}(U_d) \leq x_d) \\ &= P(U_1 \leq F_1(x_1), \ldots, U_d \leq F_d(x_d)) \\ &= C(F_1(x_1), \ldots, F_d(x_d))\end{aligned}$$

Here $(U_1, \ldots, U_d)^T = (F_1(X_1, V_1), \ldots, F_d(X_d, V_d) \sim C$ is a copula . Therefore for discrete data, we can assume that after marginally applying the generalized distributional transform the copula we get is a Gaussian copula, i.e. we get that

$$(U_1, \ldots, U_d)^T = (F_1(X_1, V_1), \ldots, F(X_d, V_d))^T \sim C_{\mathbf{\Sigma}^0}$$

or

$$(\Phi^{-1}(U_1), \ldots, \Phi^{-1}(U_d))^T = (\Phi^{-1}(F_1(X_1, V_1)), \ldots, \Phi^{-1}(F(X_d, V_d)))^T \sim N(\mathbf{0}, \mathbf{\Sigma}^0)$$

Based on our assumption, $(U_1, \ldots, U_d)^T$ is also a continuous random variable from nonparanormal family. Becasue it is now from a Gaussian copula family, and for continuous random variable these two families are the same.

We can also consider $U_j = F_j(X_j, V_j)$, $j = 1, \ldots, d$ to be our latent variables that constitute a Gaussian copula, where $F_j(x_j) = P(X_j \leq x_j)$ is the marginal cdf of $\boldsymbol{X}$. And our data is generated by a quantile transformation from U_j, i.e. $X_j = F_j^{-1}(U_j)$. Hence to estimate the latent correlation matrix in Gaussian copula, we can transform the discrete data marginally with generalized distributional transformation, and then use the transformed data to get the estimate as we did in the continuous case.

3.2 Transform the discrete data with empirical distribution

We transform the discrete data $(X_1, \ldots, X_d)^T$ to the latent continuous $(U_1, \ldots, U_d)^T$ using the empirical cumulative function (empirical cdf) and empirical probability mass function (emperical pmf). Suppose for X_j there are m possible values which are $c_1 < c_2, \ldots < c_m$, then the empirical pmf is

$$\hat{P}_j(c_l) = \frac{1}{n} \sum_{i=1}^{n} I(x_{ij} = c_l)$$

and the empirical cdf is

$$\hat{F}_j(t) = \sum_{i=1}^{n} I(x_{ij} \leq t)$$

and

$$\hat{F}_j(c_l) = \sum_{k=1}^{l} \hat{P}_j(c_k)$$

Then our transformation would be

$$\hat{F}_j(c_l) = \hat{F}_j(c_{l-1}) + \hat{P}_j(c_l) V_j$$

when $l = 1$, we denote $\hat{F}_j(c_0) = 0$, and V_j is an independent random variable following Uniform(0,1).

And the quantile transform would is

$$\hat{F}_j^{-1}(u) = \sum_{k=1}^{m} c_k I(\hat{F}_j(c_{k-1}) < u \leq \hat{F}_j(c_k))$$

We perform the copula PCA with $\mathbf{\Sigma}^0$ estimated by Spearman's ρ or Kendall's τ from the data we obtained with generalized distributional transformation, which is $\{\boldsymbol{U}_i : \boldsymbol{U}_i = (F_j(X_{ij}, V_{ij}))_{j=1}^d, i = 1, \ldots, n\}$ and V_{ij}'s are independent Uniform$(0, 1)$. Then with the projection matrix formed by the first several principle components, we can do dimension reduction by projecting $(\Phi^{-1}(U_1), \ldots, \Phi^{-1}(U_d))^T$ to a lower dimension, since by our assumption of Gaussian copula, $(\Phi^{-1}(U_1), \ldots, \Phi^{-1}(U_d))^T \sim N(\mathbf{0}, \mathbf{\Sigma}^0)$. And the projected data are the scores corresponding to the principle components obtained by discrete copula PCA.

3.3 Data recovery

We can also transform the data projected to a lower dimension back to the categorical data.

An observation $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{id})^T$ is transformed to $\boldsymbol{y}_i = (y_{i1}, \ldots, y_{id})^T$ by

$$y_{ij} = \Phi^{-1}(\hat{F}_j(x_{ij}))$$

where $\hat{F}_j$ is emperical generalized distributional transformation. Suppose we have estimated $\hat{\boldsymbol{\Sigma}}^0$, the variance-covariance matrix of $\boldsymbol{y}_i$'s, with Spearman's ρ or Kendall's τ , and decomposed it to be

$$\hat{\boldsymbol{\Sigma}}^0 = \sum_{k=1}^{d} \lambda_k \boldsymbol{v}_k \boldsymbol{v}_k^T$$

where $\boldsymbol{v}_k$'s are orthogonal unit vectors. And using $\boldsymbol{v}_k$'s as our new basis in the linear space for $\boldsymbol{y}_i$'s, we have

$$\boldsymbol{y}_i^T = \sum_{k=1}^{d} \boldsymbol{y}_i^T \boldsymbol{v}_k \boldsymbol{v}_k^T$$

Having decided the number of components to adequately describe the variation in the data, say r, we can take first several PCs, say $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r$, and obtain a reduced-rank representation of $\boldsymbol{y}_i$ to be

$$\hat{\boldsymbol{y}}_i^T = \sum_{k=1}^{r} \boldsymbol{y}_i^T \boldsymbol{v}_k \boldsymbol{v}_k^T$$

After getting the recovered $\boldsymbol{y}_i$, Theorem 3.1 ensures that

$$\hat{x}_{ij} = \hat{F}_j^{-1}(\Phi(\hat{y}_{ij}))$$

So let our original data be $\boldsymbol{X}$, which is a $n \times d$ matrix, and define the following matrix functions

$$\begin{aligned}
\hat{F}(\boldsymbol{X}_{m\times n}) &= [\hat{F}_j(x_{ij})]_{m\times n} \\
\hat{F}^{-1}(\boldsymbol{X}_{m\times n}) &= [\hat{F}_j^{-1}(x_{ij})]_{m\times n} \\
\Phi(\boldsymbol{X}_{m\times n}) &= [\Phi(x_{ij})]_{m\times n} \\
\Phi^{-1}(\boldsymbol{X}_{m\times n}) &= [\Phi^{-1}(x_{ij})]_{m\times n}
\end{aligned}$$

And let $\boldsymbol{V}_r = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r)$, then the recovered data $\hat{\boldsymbol{X}}$ is

$$\hat{\boldsymbol{X}} = \hat{\boldsymbol{F}}^{-1}(\Phi(\Phi^{-1}(\hat{\boldsymbol{F}}(\boldsymbol{X}))\boldsymbol{V}_r\boldsymbol{V}_r^T))$$

4 Categorical sequence data with multinomial marginal distribution

In this section, we focus our attention on multidimensional sequence of categorized data, where each observation has p components, and every components takes a value from k possible states and these states are unordered. Therefore the response for each component is actually following a multinomial distribution with k classes and the total number of trials n is 1. For component j, we can write the marginal distribution to be Multinomial$(1, \boldsymbol{p}_j)$, where $\boldsymbol{p}_j = (p_{j1}, \ldots, p_{jk})^T$ and $\sum_{i=1}^{k} p_{ji} = 1$.

4.1 The optimal mapping method

In this case, our marginal distribution is not univariate. To apply the discrete copula PCA, one way is to convert the marginal distribution to be univariate by a mapping. We can map these k classes to a k numbers we pick, say $r_1, r_2, \ldots, r_k$. Then if we have an observation, whose response for a component class l, then we place the number r_l in that component. In this way, every observation will be converted to a vector of length p, and each component takes values from $\{r_1, \ldots, r_k\}$. In other words, by defining a mapping from the k possible classes (or states) to the numbers $\{r_1, \ldots, r_k\}$, we can induce a mapping from the original sample space $\mathcal{X}$ to a new sample space $\mathcal{Y}$, and the point in the new sample space is vector of length p. After the mapping, our mapped data will be marginally univariate with discrete distributions. Then we can apply the discrete copula PCA to the mapped data. What's more, we can set some criteria and find an optimal mapping. And the the discrete copula PCA result from optimal mapped data will be our PCA result for the multinomial-marginal data.

Note that in discrete copula PCA, we only work with the marginally generalized distributional transformed data. Suppose that we have k states $\{s_1, \ldots, s_k\}$, and we have twp mappings $\phi(s_j) = r_j, \varphi(s_j) = t_j \, j = 1, \ldots, k$. Then if the vectors $(r_1, \ldots, r_k)^T$ and $(t_1, \ldots, t_k)^T$ have the same order, then the mapped data will result in the same generalized distributional transformed data no matter which mapping we chose. Formally, we have the following definition and theorem.

Definition 4.1. For a vector of length n, say $(x_1, \ldots, x_n)^T$, assuming that we have $x_{i_1} < x_{i_2} < \cdots < x_{i_n}$, where $(i_1, i_2, \ldots, i_n)$ is a permutation of $(1, 2, \ldots, n)$, we define the order function o to be

$$o((x_1, \ldots, x_n)) = (i_1, i_2, \ldots, i_n)$$

In other words, the order function returns the order of each x_j in the vector $(x_1, \ldots, x_n)$.

Definition 4.2. Let $S = \{s_1, \ldots, s_k\}$ be the set of k classes (or states). Suppose we have two mappings ϕ and φ from S to $\mathbb{R}$. We say ϕ and φ have the order if

$$o((\phi(s_1), \ldots, \phi(s_k))) = o((\varphi(s_1), \ldots, \varphi(s_k)))$$

Theorem 4.1. *Let $S = \{s_1, \ldots, s_k\}$ be the set of k classes (or states). Suppose $\boldsymbol{x} = (x_1, \ldots, x_p)^T \in S^p$ is an observation from the distribution whose marginal distribution is multinomial with k classes. For two mappings ϕ and φ such that $\phi(s_i) = r_i, \varphi(s_j) = t_i$, $i = 1, 2, \ldots, k$, if $o((r_1, \ldots, r_k)) = o((t_1, \ldots, t_k))$, we have*

$$F_{\phi,j}(\phi(x_j), V) = F_{\varphi,j}(\varphi(x_j), V), \, j = 1, 2, \ldots, p.$$

where $F_{\phi,j}$ and $F_{\varphi,j}$ are the marginal distributions of the j-th component based on the mapped data using ϕ and φ.

In other words, the marginally generalized transformed data based on the mapped data will be the same as long as the mappings from S to $\mathbb{R}$ have the same order.

Proof. For the j-th component, let the marginal distribution be Multinomial$(1, \boldsymbol{p}_j)$, where $\boldsymbol{p}_j = (p_{j1}, \ldots, p_{jk})^T$ and $\sum_{i=1}^k p_{jk} = 1$. Let x_j be the j-th coordinate of an observation $\boldsymbol{x}$, and let two mappings $\phi : S \mapsto \mathbb{R}$ and $\varphi : S \mapsto \mathbb{R}$ have the same order $o((\phi(s_1), \ldots, \phi(s_k))) = o((\varphi(s_1), \ldots, \varphi(s_k))) = (i_1, i_2, \ldots, i_k)$. It is easy to show that

$$F_{\phi,j}(\phi(s_l)) = \sum_{i=1}^{l} p_{i_l} = F_{\varphi,j}(\varphi(s_l))$$

Also

$$F_{\phi,j}(\phi(s_l)-) = F_{\phi,j}(\phi(s_m)) = F_{\varphi,j}(\varphi(s_m)) = F_{\varphi,j}(\varphi(s_l)-)$$

where $s_m = \max\{s_i : \phi(s_i) < \phi(s_l)\} = \max\{s_i : \varphi(s_i) < \phi(s_l)\}$. Hence if $x_j = s_l$, we have

$$\begin{aligned} F_{\phi,j}(\phi(x_j), V) &= F_{\phi,j}(\phi(s_l)-)) + (F_{\phi,j}(\phi(s_l))) - F_{\phi,j}(\phi(s_l)-)))V \\ &= F_{\varphi,j}(\varphi(s_l)-)) + (F_{\varphi,j}(\varphi(s_l))) - F_{\varphi,j}(\varphi(s_l)-)))V \\ &= F_{\varphi,j}(\varphi(x_j), V) \end{aligned}$$

□

By Theorem 4.1, we know the mappings of the same order will not affect the discrete copula PCA. Since for k classes, there are only $k!$ different orders, and thus we can reduce the space of possible mapping to only $k!$ mappings which produce at most $k!$ different results, and then pick one based on some criteria. Without loss of generality, we can consider the mappings from S to $\{1, 2, \ldots, k\}$. Then we define the optimal mapping as follows.

Definition 4.3. Let $S = \{s_1, \ldots, s_k\}$ be the state space, then an optimal mapping $\phi : S \mapsto \mathbb{R}$ is a best mapping from the collection $\{f : S \mapsto \{1, 2, \ldots, k\}\}$ in terms of a desired (or relevant) criterion.

Here we propose two possible criteria.

- **Leading eigenvalue criterion**: The optimal mapping is the mapping such that, with the mapped data by this mapping, the largest eigenvalue in the eigen-decomposition of the latent correlation matrix $\mathbf{\Sigma}^0$ is the maximum among all $k!$ mappings.
- ***r*-component recovery rate criterion**: The optimal mapping is the mapping such that, with the mapped data by this mapping, the r-component recovery rate of the mapped data is the highest among all $k!$ mappings, where the r-component recovery rate is defined as the proportion of reconstructed data with first r principal components that match the original data.

4.1.1 The Bernoulli marginals case

A special case for the multinomial-marginal distribution is that of Bernoulli marginals. A Bernoulli marginal distribution is a multinomial distribution with 2 classes. So based on the discussion above, there are only two possible mappings that lead to potentially different result. Suppose that we map $\{s_1, s_2\}$ to $\{0, 1\}$. Then the two possible mappings are $s_1 \to 1, s_2 \to 0$ and $s_1 \to 0, s_2 \to 1$. By intuition, we should get only one unique result no matter which mapping we chose. For example, in a binary trial, we can encode success as 1 and failure as 0, or vise versa. And we expect that these two encodings of success and failure will not change the result. And in fact, it can be proved that under the assumption of Gaussian copula, these 2 mappings will give us the same result.

Theorem 4.2. *Suppose we have* $\boldsymbol{X} = (X_1, \ldots, X_p)$, $X_j \in \{0, 1\}$ *such that* $\boldsymbol{F}(\boldsymbol{X}, \boldsymbol{V}) \sim C_{\mathbf{\Sigma}}$. *Here* $\boldsymbol{F}(\boldsymbol{X}, \boldsymbol{V}) = (F_1(X_1, V_1), F_2(X_2, V_2), \ldots, F_p(X_p, V_p))$. *Then we have*

$$\boldsymbol{F}'(1 - \boldsymbol{X}, \boldsymbol{V}) \sim C_{\mathbf{\Sigma}}$$

where $\boldsymbol{F}'$ *is the vector of marginal cdfs of* $1 - \boldsymbol{X}$

Proof. We have $\boldsymbol{F}'(\boldsymbol{x}) = F_{1-\boldsymbol{X}}(\boldsymbol{x}) = (F'_1(x_1), \ldots, F'_p(x_p)$, and $F'_j(x) = P(1 - X_j \le x) = P(X_j \ge 1 - x) = 1 - P(X_j < 1 - x) = 1 - F_j((1 - x)-)$. Also we have $F'_j(x-) = P(1 - X_j < x) = 1 - P(X_j \ge 1 - x) = 1 - F_j(1 - x)$. Hence

$$\begin{aligned} F'_j(1 - x_j, V_j) =& F'_j((1 - x_j)-) + (F'_j(1 - x_j) - F'_j((1 - x_j)-))V_j \\ &= 1 - F_j(x_j) + (F_j(x_j) - F_j(x_j-))V_j \\ &= 1 - F_j(x_j-) + (F_j(x_j) - F_j(x_j-))(V_j - 1) \end{aligned}$$

Then letting $U_j = 1 - V_j \sim \text{Uniform}(0,1)$, we have

$$1-F'_j(1-X_j, V) = F_j(X_j-)+(F_j(X_j)-F_j(X_j-))(1-V_j) = F_j(X_j-)+(F_j(X_j)-F_j(X_j-))U_j = F_j(X_j, U_j)$$

Since $\boldsymbol{F}(\boldsymbol{X}, \boldsymbol{U}) \sim C_{\boldsymbol{\Sigma}}$, we know $1 - \boldsymbol{F}'(1 - \boldsymbol{X}, \boldsymbol{V}) \sim C_{\boldsymbol{\Sigma}}$. Let $\boldsymbol{Y} = 1 - \boldsymbol{F}'(1 - \boldsymbol{X}, V)$, and the probability density function (pdf) of Y is $c_{\boldsymbol{\Sigma}}(y_1, \ldots, y_p)$. Then $\boldsymbol{Z} = \boldsymbol{F}'(1-\boldsymbol{X}, \boldsymbol{V}) = 1-\boldsymbol{Y}$, therefore the pdf of $\boldsymbol{F}'(1 - \boldsymbol{X}, \boldsymbol{V})$ is $c_{\boldsymbol{\Sigma}}(1 - z_1, \ldots, 1 - z_p)$. Because for the Gaussian copula, we have

$$c_{\boldsymbol{\Sigma}}(1 - z_1, \ldots, 1 - z_p) = c_{\boldsymbol{\Sigma}}(z_1, \ldots, z_p)$$

Then $\boldsymbol{Z} = \boldsymbol{F}'(1 - \boldsymbol{X}, \boldsymbol{V})$ also follows the same Gaussian copula $C_{\boldsymbol{\Sigma}}$. □

By Theorem 4.2, we know that these two mappings of different orders will lead to the same Gaussian copula. Therefore performing discrete PCA with mapped data from these two mappings will eventually produce the same result.

5 Application: senators' voting data set

The 109th US Congress, comprising the Senate and the House of Representatives, was the legislative branch of the US government from January 3, 2005 to January 3, 2007. During this period, 542 bills were voted on by the US Senate. Each of 100 Senators either voted in favor or against or failed to record their vote on each of these bills. So for this data set, each observation is a senator, and each component is a call bill. Possibles values are $-1, 0, 1$, representing "Yes","Absent" and "No". Therefore the marginal distribution of each component is a multinomial distribution. We apply optimal mapping method and discrete copula PCA to this data set.

We first map the 3 classes in the multinomial distribution to 1, 2 and 3. Then each marginal distribution becomes univariate with support $\{1, 2, 3\}$. There are 3! different mappings from {"Yes", "Absent", "No"} to $\{1, 2, 3\}$. For each of them, we transform the data to continuous latent variables (U_i's in section 3.1) with empirical marginal distribution and then calculate the estimation of $\boldsymbol{\Sigma}^0$.

We used Spearman's ρ to estimate the latent correlation matrix. The criterion we used to pick the optimal mapping here is the r-component recovery rate criterion stated the section 4. r is chosen to be 2 here for the purpose of displaying the data with first two principle components. The optimal mapping we picked is then ϕ (("Yes", "Absent", "No")) $= (2, 1, 3)$ with a recovery rate of 85.87% using first and second PCs.

Figure 1 shows the r-component recovery rate (defined in 4) against r, the number of PCs used to reconstruct the data (only first 20 PCs are presented). We can see with first 5 PCs, the recovery rate is already above 90%. Table 1 shows the recovery rate with first n PCs. Up to 20 PCs are presented in the table.

Table 1: Recovery rate with first n PCs

# of PCs	1	2	3	4	5
Recovery rate	72.65%	85.87%	88.92%	90.00%	90.80%
# of PCs	6	7	8	9	10
Recovery rate	90.99%	91.55%	91.93%	92.15%	92.37%
# of PCs	11	12	13	14	15
Recovery rate	92.57%	92.75%	92.81%	93.01%	93.24%
# of PCs	16	17	18	19	20
Recovery rate	93.42%	93.63%	93.78%	93.99%	94.13%

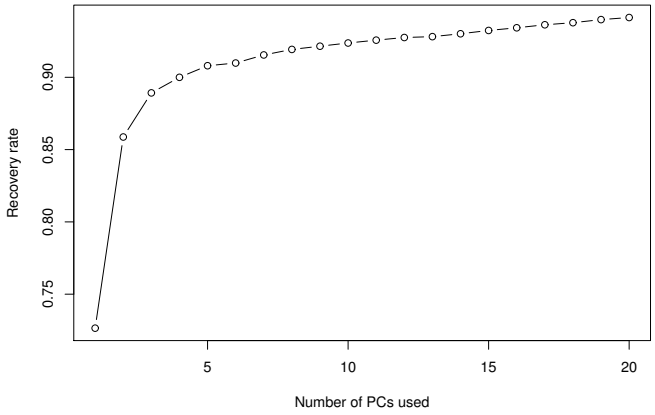
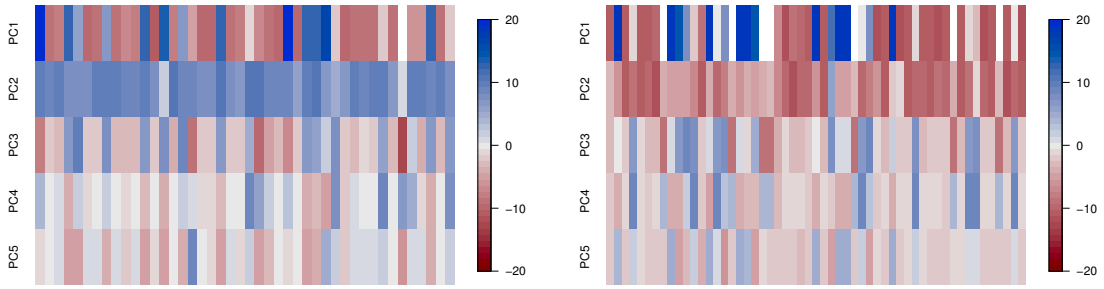


Figure 1: Recovery rate with first n PCs

Since with first 2 PCs, we can already recover 85.87% and with first 5 PCs, we can recover 90.8% of the data, then we can consider using first 5 PCs to reduce the dimension. We present the scores of the first 5 PCs in Figure 2 for senators from Democratic Party and Republican Party. Red color indicates that the score is negative while the blue color indicates that the score is positive. From these two figures, we can see scores for PC1 are very different within each of the parties. There are large positive scores as well as large negative scores. For PC2, interestingly, most Democratic senators have positive scores while most Republican senators have negative scores. The remaining 3 PCs are obviously smaller in terms of absolute values compared to the first 2 PCs. From here we can see PC2 is a component that separate 2 parties.



(a) Democratic (b) Republican

Figure 2: Scores of first 5 PCs

We then visualize the data with PC1 and PC2. In Figure 3, the senators from Democratic Party are colored blue and senators from Republican Party are colored red. Again, in this visualization, PC2 separate these two parties. What's more, we can see there are 2 possible groups within Democratic senators and 3 possible groups within Republican Party. This indicates that senators in on cluster might belong to a bloc.

We also note that Chafee and Nelson are far away from the majority of their own parties, which make sense based on our external knowledge outside this data set

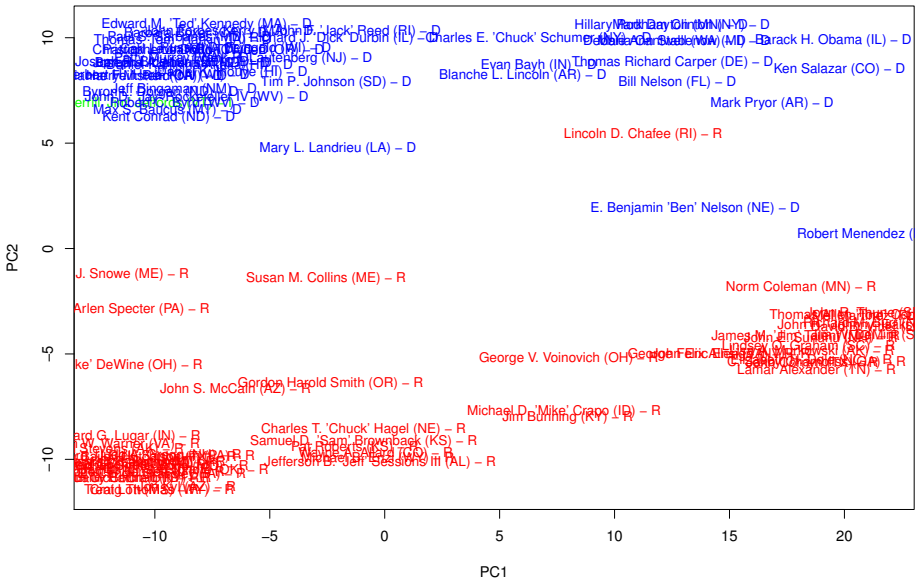


Figure 3: Visualization with PC1 and PC2

We are also interested in the loadings of PC2. Because based on the visualization and scores, PC2 is an important component. So we want to know what the loading for PC2 is like. However, to look at the values of 542 loadings is impossible. What we do is to separate negative loadings and positive loadings, and utilize word cloud to visualize the two parts of loadings. The size of word corresponds to the absolute value of that loading. From Figure 4 and Figure 5, we find Budget, Spending and Taxes bills appear to be high in both positive and negative loadings. But there are still to much bills. So we combine all bills of the same type and visualize negative and positive loadings in Figure 6 and Figure 7. We find Budget, Spending and Taxes and Appropriations are both high in negative and positive loadings, and for negative loadings, Executive Branch is also high.

Figure 4: Negative loadings for all bills

Figure 5: Positive loadings for all bills

Figure 6: Negative loadings for all bill types

Figure 7: Positive loadings for all bill types

6 Matrix Gaussian copula method

6.1 The matrix Gaussian copula

Another way to represent the data with multinomial marginal distribution is to use a random matrix. Suppose we have k classes in the multinomial distribution, and that there are p components, then the random matrix is a $p \times k$ matrix. Each row is considered to be a multivariate marginal distribution, and for our data the marginal distribution is multinomial distribution whose number of trials is 1. By imitating the univariate marginal distribution case, we have the following definitions to define the matrix Gaussian copula and the matrix Gaussian copula family.

In this section, we use the following notation. Suppose $\boldsymbol{M}$ is a matrix, we denote

- $\boldsymbol{M}_{i\cdot}$: the i-th row of $\boldsymbol{M}$ as a column vector
- $\boldsymbol{M}_{\cdot j}$: the j-th column of $\boldsymbol{M}$
- $\boldsymbol{M}_{i[j-1]}$: the i-th row and the first $j-1$ columns of $\boldsymbol{M}$ as a column vector
- $\boldsymbol{M}_{\cdot[j-1]}$: the first $j-1$ columns of $\boldsymbol{M}$

Definition 6.1. (Multivariate quantile transform) Let F be a k-dimensional distribution function (cdf) and let $V_1, \ldots, V_k$ be iid Uniform$(0,1)$ distributed random variables. Then the multivariate quantile transform $\boldsymbol{Y} = \tau_F(\boldsymbol{V})$ is define recursively as

$$\begin{aligned} Y_1 &= F_1^{-1}(V_1) \\ Y_j &= F_{j|1,\ldots,j-1}^{-1}(V_j|Y_1, \ldots, Y_{j-1}),\ 2 \le j \le k \end{aligned}$$

where $F_{j|1,\ldots,j-1}$ denote the conditional cdf of the j-th component conditioning on the first $j-1$ components.

By multivariate quantile transformation, we have $\boldsymbol{Y} = \tau_F^{-1}(\boldsymbol{V}) \sim F$.

Definition 6.2. (Multivariate distributional transform) Let $\boldsymbol{X}$ be a k-dimensional random vector, and F be the cdf of $\boldsymbol{X}$, the multivariate distributional transform is defined as

$$\tau_F(\boldsymbol{x}) = (F_1(x_1), F_{2|1}(x_2|x_1), \ldots, F_{k|1,\ldots,k-1}(x_k|x_1, \ldots, x_{k-1}))$$

By multivariate distributional transformation $\tau_F(\boldsymbol{X}) \sim$ Uniform$((0,1)^k)$. And $\tau_F^{-1}(\tau_F(\boldsymbol{X})) = \boldsymbol{X}$.

Recall that a Gaussian copula is defined as marginally quantile transformed random vector from a multivariate normal distribution. Let $\boldsymbol{X} \sim N(\boldsymbol{0}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is a correlation matrix, then the marginal distribution is $N(0,1)$, and the Gaussian copula is $(\Phi(X_1), \ldots, \Phi(X_p))$. Here $U_i = \Phi(X_i) \sim$ Uniform$(0,1)$, and we denote $(U_1, \ldots, U_p) \sim C_\Sigma$.

In the case when the observation is matrix, we define matrix Gaussian copula through matrix normal distribution. We treat the row as our multivariate marginal in the matrix case.

Definition 6.3. Let $\boldsymbol{X}_{p\times k} \sim N_{p\times k}(\boldsymbol{0}_{p\times k}, R_{p\times p}, \Sigma_{k\times k})$ be a random matrix from matrix normal distribution, where R and Σ are correlation matrix. In other words, vec$(\boldsymbol{X}) \sim N_{kp}(\boldsymbol{0}, \Sigma \otimes R)$. Each row follows a $N(\boldsymbol{0}, \Sigma)$ multivariate normal distribution. Denote the multivariate distributional transformation as τ_Σ, then the matrix Gaussian copula $C_{\Sigma,R}$ is

$$(\tau_\Sigma(\boldsymbol{X}_{1\cdot}), \ldots, \tau_\Sigma(\boldsymbol{X}_{p\cdot}))^T \sim C_{\Sigma,R}$$

where $\boldsymbol{X}_{i,}$ is the i-th row of $\boldsymbol{X}$, and the transformed random matrix

$$\boldsymbol{U} = (\boldsymbol{U}_{1\cdot} \ldots, \boldsymbol{U}_{p\cdot})^T = (\tau_\Sigma(\boldsymbol{X}_{1\cdot}), \ldots, \tau_\Sigma(\boldsymbol{X}_{p\cdot}))^T$$

is a $p \times k$ matrix with marginals $\boldsymbol{U}_{i\cdot} \sim$ Uniform$((0,1)^k)$.

The following lemma and theorem can help to give an equivalent definition of matrix Gaussian copula in a simpler way.

Lemma 6.1. *For continuous random variables, let $F_{X|\boldsymbol{Y}}$ be the conditional cdf of X given $\boldsymbol{Y}$, then $F_{X|\boldsymbol{Y}}(X|\boldsymbol{Y}) \sim \text{Uniform}(0,1)$ and is independent of $\boldsymbol{Y}$.*

Proof. By Theorem 3.1, for a given $\boldsymbol{y}$, we know $F_{X|\boldsymbol{Y}}(X|\boldsymbol{Y})|(\boldsymbol{Y}=\boldsymbol{y}) = F_{X|\boldsymbol{Y}}(X|\boldsymbol{y})|(\boldsymbol{Y}=\boldsymbol{y})$ is $\text{Uniform}(0,1)$. Since it does not depend on the choice of $\boldsymbol{y}$, we have $F_{X|\boldsymbol{Y}}(X|\boldsymbol{Y}) \sim \text{Uniform}(0,1)$ and is independent of $\boldsymbol{Y}$.

□

Lemma 6.2. *Let $\boldsymbol{X} \sim N_{p\times k}(\boldsymbol{0}, R, \Sigma)$, then $X_{ij}|\boldsymbol{X}_{,1} = \boldsymbol{x}_{,1}, \ldots, \boldsymbol{X}_{,j-1} = \boldsymbol{x}_{,j-1}$ has the same distribution as $X_{ij}|X_{i1} = x_{i1}, \ldots, X_{i,j-1} = x_{i,j-1}$.*

Proof. Since each row follows $N(0,\Sigma)$, we then have

$$(X_{i1}, \ldots, X_{i,j-1}, X_{ij}) \sim N\left(\begin{bmatrix}\boldsymbol{0}\\ 0\end{bmatrix}, \begin{bmatrix}\Sigma_{j-1} & \boldsymbol{b}_{j-1}\\ \boldsymbol{b}_{j-1}^T & 1\end{bmatrix}\right)$$

where

$$\Sigma_{j-1} = \begin{bmatrix}I_{j-1} & \boldsymbol{0}\end{bmatrix}\Sigma\begin{bmatrix}I_{j-1} & \boldsymbol{0}\end{bmatrix}$$
$$\boldsymbol{b}_{j-1} = \begin{bmatrix}I_{j-1} & \boldsymbol{0}\end{bmatrix}\Sigma\boldsymbol{e}_j$$

$\boldsymbol{e}_j$ is a vector of length k with 1 in the j-th position and 0's elsewhere. Then the conditional distribution of the j-th element of the i-th conditioning on the first $j-1$ elements of the i-th row is

$$X_{ij}|\boldsymbol{X}_{i[j-1]} = \boldsymbol{x}_{i[j-1]} \sim N\left(\boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{x}_{i[j-1]}, 1 - \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1}\right)$$

On the other hand,

$$\text{vec}(\boldsymbol{X}_{\cdot 1}, \ldots, \boldsymbol{X}_{\cdot(j-1)}, \boldsymbol{X}_{\cdot j}) \sim N\left(\begin{bmatrix}\boldsymbol{0}_{p(j-1)}\\ \boldsymbol{0}_p\end{bmatrix}, \begin{bmatrix}\Sigma_{j-1}\otimes R & \boldsymbol{b}_{j-1}\otimes R\\ \boldsymbol{b}_{j-1}^T\otimes R & R\end{bmatrix}\right)$$

Then

$$\boldsymbol{X}_{\cdot j}|\boldsymbol{X}_{\cdot[j-1]} = \boldsymbol{x}_{\cdot[j-1]} \sim N\left((\boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\otimes I)\boldsymbol{x}_{\cdot[j-1]}, R - \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1}R\right)$$

From the joint distribution $\boldsymbol{X}_{\cdot j}|\boldsymbol{X}_{\cdot[j-1]} = \boldsymbol{x}_{\cdot[j-1]}$, we get the marginal conditional distribution

$$X_{ij}|\boldsymbol{X}_{\cdot[j-1]} = \boldsymbol{x}_{\cdot[j-1]} \sim N\left(\boldsymbol{b}_{j-1}^T\Sigma_{j-1}\boldsymbol{x}_{i[j-1]}, 1 - \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1}\right)$$

Hence $X_{ij}|\boldsymbol{X}_{\cdot[j-1]} = \boldsymbol{x}_{\cdot[j-1]}$ has the same distribution as $X_{ij}|\boldsymbol{X}_{i[j-1]} = \boldsymbol{x}_{i[j-1]}$. □

Lemma 6.3. *Let F be the cdf of $N(\boldsymbol{0},\Sigma)$, and $\tau_\Sigma^{-1} = \tau_F^{-1}$ be the quantile transform from $\boldsymbol{V} = (V_1, \ldots, V_k)$ to $\boldsymbol{Y} = (Y_1, \ldots, Y_k) = \tau_F^{-1}(\boldsymbol{V})$, then recursively we have*

$$Y_1 = F_1^{-1}(V_1) = \Phi^{-1}(V_1)$$
$$Y_j = F_{j|1,\ldots,j-1}^{-1}(V_j|Y_1, \ldots, Y_{j-1}) = \Phi^{-1}(V_j)\sqrt{1 - \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{k-1}} + \boldsymbol{b}_{k-1}{}^T\Sigma_{j-1}^{-1}\boldsymbol{Y}^{(j-1)}$$

where $\boldsymbol{Y}^{(j-1)} = (Y_1, \ldots, Y_{j-1})$, Σ_{j-1} is the matrix of first $j-1$ columns and first $j-1$ rows of Σ, and $\boldsymbol{b}_{j-1}$ is the vector of j-th column and first $j-1$ rows, i.e.

$$\Sigma_{j-1} = \begin{bmatrix}I_{j-1} & \boldsymbol{0}\end{bmatrix}\Sigma\begin{bmatrix}I_{j-1} & \boldsymbol{0}\end{bmatrix}$$
$$\boldsymbol{b}_{j-1} = \begin{bmatrix}I_{j-1} & \boldsymbol{0}\end{bmatrix}\Sigma\boldsymbol{e}_j$$

$\boldsymbol{e}_j$ is a vector of length k with 1 in the j-th position and 0's elsewhere.

Proof. The marginal distribution of $N(\mathbf{0}, \Sigma)$ is $N(0,1)$, so

$$Y_1 = F_1^{-1}(V_1) = \Phi^{-1}(V_1)$$

And we know the cdf of $N(\mu, \sigma^2)$ can be written as

$$\Phi_{\mu,\sigma^2}(x) = \Phi\left(\frac{x-\mu}{\sigma}\right)$$

Therefore

$$\Phi^{-1}_{\mu,\sigma^2}(u) = \Phi^{-1}(u)\sigma + \mu$$

And by the conditional distribution we identified in Lemma 6.1, we have

$$Y_j = F^{-1}_{j|1,\dots,j-1}(V_j|Y_1,\dots,Y_{j-1}) = \Phi^{-1}(V_j)\sqrt{1 - \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1}} + \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{Y}^{(j-1)}$$

□

Theorem 6.4. *$C_{I,R} = C_{\Sigma,R}$. In other words, if $\boldsymbol{X} \sim N_{p\times k}(\mathbf{0}, R, \Sigma)$, and $(\tau_\Sigma(\boldsymbol{X}_{1\cdot}), \dots, \tau_\Sigma(\boldsymbol{X}_{p\cdot})) = (\boldsymbol{U}_{1\cdot}, \dots, \boldsymbol{U}_{p\cdot})^T = (\boldsymbol{U}_{\cdot 1}, \dots, \boldsymbol{U}_{\cdot k})$, then $(\Phi^{-1}(\boldsymbol{U}_{\cdot 1}), \dots, \Phi^{-1}(\boldsymbol{U}_{\cdot k})) \sim N_{p\times k}(\mathbf{0}, R, I)$.*

Proof. By Lemma 6.2, we know the distribution of X_{ij} conditioning on the first $j-1$ columns of $\boldsymbol{X}$ is the same as the distribution of X_{ij} conditioning on the first $j-1$ elements of *thei*-th row of $\boldsymbol{X}$. In multivariate marginal distributional transform, we use the the distribution of X_{ij} conditioning on the first $j-1$ elements of i-th row of $\boldsymbol{X}$. And because of the equivalence of the two conditional distribution, and also with Lemma 6.1, we immediately know that $\boldsymbol{U}_{\cdot 1}, \boldsymbol{U}_{\cdot 2}, \dots, \boldsymbol{U}_{\cdot k}$ are independent, and hence $\Phi^{-1}(\boldsymbol{U}_{\cdot 1}), \dots, \Phi^{-1}(\boldsymbol{U}_{\cdot k})$ are independent. By Lemma 6.3, we have the explicit form of multivariate marginal transform, and also from there we know $\mathrm{vec}(\Phi^{-1}(\boldsymbol{U}_{\cdot 1}), \dots, \Phi^{-1}(\boldsymbol{U}_{\cdot k}))$ is a linear transformation of $\mathrm{vec}(\boldsymbol{X})$. Thus now we only need to prove $\Phi^{-1}(\boldsymbol{U}_{\cdot j}) \sim N(\mathbf{0}, R)$, $j = 1, \dots, k$.

First,

$$\boldsymbol{X}_{\cdot 1} = \Phi^{-1}(\boldsymbol{U}_{\cdot 1}) \Rightarrow \Phi^{-1}(\boldsymbol{U}_{\cdot 1}) \sim N(\mathbf{0}, R)$$

because for matrix normal distribution $N_{p\times k}(\mathbf{0}, R, \Sigma)$, each column is following $N(\mathbf{0}, R)$.

For $j \geq 2$, we have

$$\boldsymbol{X}_{\cdot j} = \Phi^{-1}(\boldsymbol{U}_{\cdot j})\sqrt{1 - \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1}} + \boldsymbol{X}_{\cdot[j-1]}\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1}$$

Therefore

$$\mathrm{E}(\boldsymbol{X}_{\cdot j}) = \mathrm{E}(\Phi^{-1}(\boldsymbol{U}_{\cdot j}))\sqrt{1 - \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1}} + \mathrm{E}(\boldsymbol{X}_{\cdot[j-1]})\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1})$$

Since $\mathrm{E}(\boldsymbol{X}) = \mathbf{0}$, we have $\mathrm{E}(\boldsymbol{X}_{\cdot j}) = \mathbf{0}$ and $\mathrm{E}(\boldsymbol{X}_{\cdot[j-1]}) = \mathbf{0}$, and hence $\mathrm{E}(\Phi^{-1}(\boldsymbol{U}_{\cdot j})) = \mathbf{0}$.

Also

$$\mathrm{Var}(\boldsymbol{X}_{\cdot j}) = (1 - \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1})\,\mathrm{Var}(\Phi^{-1}(\boldsymbol{U}_{\cdot j})) + \mathrm{Var}\left(\begin{bmatrix} \boldsymbol{X}_{1[j-1]}{}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1} \\ \vdots \\ \boldsymbol{X}_{p[j-1]}{}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1} \end{bmatrix}\right)$$

Note that

$$\begin{bmatrix} \boldsymbol{X}_{1[j-1]}{}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1} \\ \vdots \\ \boldsymbol{X}_{p[j-1]}{}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{X}_{1[j-1]} \\ \vdots \\ \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{X}_{p[j-1]} \end{bmatrix} = I \otimes (\boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1})\mathrm{vec}(\boldsymbol{X}_{1[j-1]}, \dots, \boldsymbol{X}_{p[j-1]})$$

And

$$\mathrm{vec}((\boldsymbol{X}_{1[j-1]}, \dots, \boldsymbol{X}_{p[j-1]})) = \mathrm{vec}((\boldsymbol{X}_{\cdot 1}, \dots, \boldsymbol{X}_{\cdot(j-1)})^T) = \mathrm{vec}(\boldsymbol{X}_{\cdot[j-1]}^T) \sim N(\mathbf{0}, R \otimes \Sigma_{j-1})$$

$$\operatorname{Var}\left(\begin{bmatrix} \boldsymbol{X}_{1[j-1]}{}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1} \\ \vdots \\ \boldsymbol{X}_{p[j-1]}{}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1} \end{bmatrix}\right) = \left(I \times (\boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1})\right)(R \otimes \Sigma_{j-1})\left(I \otimes (\boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1})\right)$$
$$= R \otimes (\boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1})$$
$$= \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1}R$$

Thus

$$R = (1 - \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1})\operatorname{Var}(\Phi^{-1}(\boldsymbol{V}_j)) + \boldsymbol{b}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{b}_{j-1}R \Rightarrow \operatorname{Var}(\Phi^{-1}(\boldsymbol{V}_j)) = R$$

Hence we proved that $\operatorname{vec}(\Phi^{-1}(\boldsymbol{V}_1), \ldots, \Phi^{-1}(\boldsymbol{V}_k)) \sim N(\boldsymbol{0}, I \otimes R)$, i.e. $(\Phi^{-1}(\boldsymbol{V}_1), \ldots, \Phi^{-1}(\boldsymbol{V}_k)) \sim MN(\boldsymbol{0}, R, I)$. And therefore $C_{I,R} = C_{\Sigma,R}$.

□

By Theorem 6.4, we know that matrix Gaussian copula does not depends on Σ, therefore we can write $C_{\Sigma,R} = C_{I,R} = C_R$, and we can have the following equivalent definition of matrix Gaussian copula C_R.

Definition 6.4. Let $\boldsymbol{X}_{p\times k} \sim N_{p\times k}(\boldsymbol{0}_{p\times k}, R_{p\times p}, I_{k\times k})$ be a random matrix from a matrix normal distribution, and let

$$(\Phi(\boldsymbol{X}_{1\cdot}), \ldots, \Phi(\boldsymbol{X}_{p\cdot}))^T = (\boldsymbol{U}_{1\cdot}, \ldots, \boldsymbol{U}_{p\cdot})^T = (\Phi(\boldsymbol{X}_{\cdot 1}), \ldots, \Phi(\boldsymbol{X}_{\cdot k})) = (\boldsymbol{U}_{\cdot 1}, \ldots, \boldsymbol{U}_{\cdot k})$$

where $\boldsymbol{X}_{i\cdot}$ is the i-th row of $\boldsymbol{X}$, $\boldsymbol{X}_{\cdot j}$ is the j-th column of $\boldsymbol{X}$. Then the matrix Gaussian copula C_R is

$$(\boldsymbol{U}_{1\cdot}, \ldots, \boldsymbol{U}_{p\cdot})^T = (\boldsymbol{U}_{\cdot 1}, \ldots, \boldsymbol{U}_{\cdot k}) \sim C_R$$

Now we define the matrix Gaussian copula family.

Definition 6.5. A random matrix $\boldsymbol{X}_{p\times k}$ is said to be from a a matrix Gaussian copula family if the joint cdf F can be written as

$$F(\boldsymbol{x}_{1\cdot}, \ldots, \boldsymbol{x}_{p\cdot}) = C_R\left(\tau_{F_{(1)}}(\boldsymbol{x}_{1\cdot}), \ldots, \tau_{F_{(p)}}(\boldsymbol{x}_{p\cdot})\right)$$

where $\boldsymbol{x}_{i\cdot}$ is the i-th row of $\boldsymbol{x}_{p\times k}$, and $F_{(i)}$ is the cdf of i-th row $\boldsymbol{X}_{i\cdot}$.

In the definition of matrix Gaussian copula family, compared to the Gaussian copula family, the univariate marginal distribution transformation is changed to be multivariate marginal distributional transformation. And when $k = 1$, our matrix Gaussian copula family is the same as the Gaussian copula family.

6.2 Matrix Gaussian copula PCA for data with multinomial marginals

In the multinomial marginal case, we have discrete distributions. So to estimate the underlying Gaussian copula, we need to use the generalized multivariate distributional transformation. And we assume the marginally generalized multivariate distributional transformed random matrix is a matrix Gaussian copula.

Definition 6.6 (the generalized multivariate distributional transform, Rüschendorf (2013), Chapter 1)**.** Let $\boldsymbol{X}$ be an k-dimensional random vector and let $\boldsymbol{W} = (W_1, \ldots, W_k)$ be iid Uniform$(0, 1)$ random variables. For $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_k)$ define

$$\tau_F(\boldsymbol{x}, \boldsymbol{\lambda}) = (F_1(x_1, \lambda_1), F_{2|1}(x_2, \lambda_2|x_1), \ldots, F_{k|1,\ldots,k-1}(x_k, \lambda_k|x_1, \ldots, x_{k-1}))$$

where

$$F_1(x_1, \lambda_1) = F(x_1-) + (F_1(x_1) - F_1(x_1-))\lambda_1$$
$$F_{j|1,\ldots,j-1}(x_j, \lambda_j | x_1, \ldots, x_{j-1})$$
$$= F_{j|1,\ldots,j-1}(x_j - | x_1, \ldots, x_{j-1}) + (F_{j|1,\ldots,j-1}(x_j | x_1, \ldots, x_{j-1}) - F_{j|1,\ldots,j-1}(x_j - | x_1, \ldots, x_{j-1}))\lambda_j$$
$$, j = 2, \ldots, k$$

are the distributional transforms of the one-dimensional conditional distributions. Finally the generalized multivariate distributional transform of $\boldsymbol{X}$ is defined as

$$\boldsymbol{U} = \tau_F(\boldsymbol{X}, \boldsymbol{W})$$

Theorem 6.5 (Rüschendorf (2013), Chapter 1)**.** *Let $\boldsymbol{X}$ be a random vector and let $\boldsymbol{U} = \tau_F(\boldsymbol{X}, \boldsymbol{W})$ denote its multivariate distributional transform. Then*

$$\boldsymbol{U} \sim \text{Uniform}\left((0,1)^k\right)$$

that is the components U_i of $\boldsymbol{U}$ are iid Uniform$(0,1)$.

And multivariate quantile transform τ_F^{-1} is inverse to the generalized multivariate distributional transform

$$\boldsymbol{X} = \tau_F^{-1}(\boldsymbol{U}) = \tau_F^{-1}(\tau_F(\boldsymbol{X}, \boldsymbol{W})) \quad a.s.$$

Theorem 6.5 guarantees that by applying the generalized multivariate distributional transform marginally to each row of a random matrix, we can get a matrix copula. And it also ensures that we can reconstruct our discrete data with the multivariate quantile transform.

Now suppose we have p components, each of them is multinomial distributed. Suppose there are k classes for our multinomial marginal distribution, and the number of trials is 1. Since the sum of a multinomial distributed random vector is always 1, we can take the first $k-1$ elements and form a $p \times (k-1)$ random mart ix $\boldsymbol{X}$. And each row $\boldsymbol{X}_{i\cdot} \sim$ Multinomial$(1, \boldsymbol{p}_i = (p_{i1}, \ldots, p_{i,k-1}))$.

By Theorem 6.5, let $\boldsymbol{W}_1, \boldsymbol{W}_2, \ldots, \boldsymbol{W}_p$ be iid Uniform$((0,1)^k)$, and $(\tau_{F_{(1)}}(\boldsymbol{X}_{1\cdot}, \boldsymbol{W}_1), \ldots, \tau_{F_{(p)}}(\boldsymbol{X}_{p\cdot}, \boldsymbol{W}_p))^T = (\boldsymbol{U}_{1\cdot}, \ldots, \boldsymbol{U}_{p\cdot})^T = (\boldsymbol{U}_{\cdot 1}, \ldots, \boldsymbol{U}_{\cdot(k-1)})$,we have for discrete multivariate marginal distribution,

$$F(\boldsymbol{x}_{1\cdot}, \ldots, \boldsymbol{x}_{\cdot}) = C(\tau_{F_{(1)}}(\boldsymbol{x}_{1\cdot}), \ldots, \tau_{F_{(p)}}(\boldsymbol{x}_{p\cdot}))$$

where $(\boldsymbol{U}_{1\cdot}, \ldots, \boldsymbol{U}_{p\cdot})^T = (\boldsymbol{U}_{\cdot 1}, \ldots, \boldsymbol{U}_{\cdot(k-1)}) \sim C$. So by the assumption that $\boldsymbol{X}$ if from matrix Gaussian copula family, we can assume

$$C = C_R$$

and thus by Theorem 6.4, we have

$$(\Phi^{-1}(\boldsymbol{U}_{\cdot 1}), \ldots, \Phi^{-1}(\boldsymbol{U}_{\cdot(k-1)})) \sim N_{p\times k}(\boldsymbol{0}, R, I)$$

which is equivalent to

$$\Phi^{-1}(\boldsymbol{U}_{\cdot j}) \overset{iid}{\sim} N(\boldsymbol{0}, R),\ j = 1, \ldots, k-1$$

Now we want to find the generalized multivariate distributional transform for the multivariate marginal distribution $F_{(i)} =$ Multinomial$(1, \boldsymbol{p}_i)$. Since for multinomial distribution, we have

$$X_{ij} | X_{i1} = x_{i1}, \ldots, X_{i,j-1} = x_{i,j-1} \sim \text{Binomial}\left(1 - \sum_{l=1}^{j-1} x_{il}, p_{ij}/(1 - \sum_{l=1}^{j-1} p_{il})\right)$$

Then by the definition of generalized distributional transform, we know the j-th element of transformed random vector $\boldsymbol{U}_{i\cdot}$ from $\boldsymbol{X}_{i\cdot}$ is

$$U_{ij} = \begin{cases} 0 & \text{, if one of } x_{i1}, \ldots, x_{i,j-1} \text{ is } 1 \\ \frac{1-\sum_{l=1}^{j} p_{ij}}{1-\sum_{l=1}^{j-1} p_{il}} W_{ij} & \text{, if } x_{i1} = \ldots, x_{i,j-1} = 0 \text{ and } X_{ij} = 0 \\ \frac{1-\sum_{l=1}^{j} p_{ij}}{1-\sum_{l=1}^{j-1} p_{il}} + \frac{p_{ij}}{1-\sum_{l=1}^{j-1} p_{il}} W_{ij} & \text{, if } x_{i1} = \ldots, x_{i,j-1} = 0 \text{ and } X_{ij} = 1 \end{cases}$$

We use the generalized multivariate transform for each of the observation (which is a $p \times (k-1)$ matrix). The estimated $\hat{p}_i, i = 1, \ldots, p$ is used. Let the transformed t-th observed random matrix be $\boldsymbol{U}^{(t)} = (\boldsymbol{U}_{1\cdot}^{(t)}, \ldots, \boldsymbol{U}_{p\cdot}^{(t)})^T = (\boldsymbol{U}_{\cdot 1}^{(t)}, \ldots, \boldsymbol{U}_{\cdot(k-1)}^{(t)}), t = 1, \ldots, n$. The columns $\Phi^{-1}(\boldsymbol{U}_{\cdot j}^{(t)})$ are iid $N(\boldsymbol{0}, R)$, hence we have

$$\Phi^{-1}(\boldsymbol{U}_{\cdot j}^{(t)}) \overset{iid}{\sim} N(\boldsymbol{0}, R), j = 1, \ldots, k-1, t = 1, \ldots, n$$

Hence we can estimate R by MLE

$$\hat{R} = \frac{1}{n(k-1)} \sum_{t=1}^{n} \sum_{j=1}^{k-1} (\boldsymbol{U}_{\cdot j}^{(t)} - \bar{\boldsymbol{U}})(\boldsymbol{U}_{\cdot j}^{(t)} - \bar{\boldsymbol{U}})^T, \bar{\boldsymbol{U}} = \frac{1}{n(k-1)} \sum_{t=1}^{n} \sum_{j=1}^{k-1} \boldsymbol{U}_{\cdot j}^{(t)}$$

Alternatively, since $\boldsymbol{U}_{\cdot j}^{(k)} \overset{iid}{\sim} C_R$, we can also use Spearman's ρ or Kendall's τ to estimate R. In this way the estimator is more robust to the deviation of estimated $\hat{\boldsymbol{p}}_i$ to the true $\boldsymbol{p}_i$.

Finally, with our estimated $\hat{R}$, we can perform PCA with this underlying correlation matrix. Since $\text{vec}(\Phi^{-1}(U_{\cdot 1}), \ldots, \Phi^{-1}(\boldsymbol{U}_{\cdot(k-1)})) \sim N(\boldsymbol{0}, I \otimes R)$, and suppose $R = P \Lambda P^T$, then the decomposition $I \otimes R$ is

$$I \otimes R = (I \otimes P)(I \otimes \Lambda)(I \otimes P^T)$$

we can then project $\text{vec}(\Phi^{-1}(\boldsymbol{V}_1), \ldots, \Phi^{-1}(\boldsymbol{V}_{k-1}))$ to $I \otimes P_{\cdot j}$ to get the score of j-th component, where $P_{\cdot j}$ is the j-th column of P. Therefore the score for j-th component is

$$(P_{\cdot j}^T \Phi^{-1}(\boldsymbol{U}_{\cdot 1}), \ldots, P_{\cdot j}^T \Phi^{-1}(\boldsymbol{U}_{\cdot(k-1)}))^T$$

,which is a vector of length $k-1$.

7 Discussion

Although the discrete Gaussian copula PCA and discrete matrix Gaussian copula PCA have some model assumptions, but it is still very flexible since we only made assumptions on the copula level. The univariate and multivariate marginals can be any distributions. What's more, using Spearman's ρ or Kendall's τ can make the estimation of underlying correlation matrix more robust to the deviation of empirical or estimated marginals to the true marginals. In this paper, all the marginals are discrete or multinomial. But since the generalized (multivariate) distributional transform is equivalent to the usual (multivariate) distributional transform, this method also works when the marginals are mixture of discrete and continuous univariate or multivariate marginals.

References

Theodore Wilbur Anderson, Theodore Wilbur Anderson, Theodore Wilbur Anderson, Theodore Wilbur Anderson, and Etats-Unis Mathématicien. *An introduction to multivariate statistical analysis*, volume 2. Wiley New York, 1958.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

Maria G Borgognone, Javier Bussi, and Guillermo Hough. Principal component analysis in sensory analysis: covariance or correlation matrix? *Food quality and preference*, 12(5-7):323–326, 2001.

W.L. Buntine and A. Jakulin. Discrete components analysis. In C. Saunders, M. Grobelnik, S. Gunn, and J. Shawe-Taylor, editors, *Subspace, Latent Structure and Feature Selection Techniques*. Springer-Verlag, 2006.

Wray Buntine. Variational extensions to em and multinomial pca. In *European Conference on Machine Learning*, pages 23–34. Springer, 2002.

John Canny. Gap: a factor model for discrete data. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 122–129. ACM, 2004.

Michael Collins, Sanjoy Dasgupta, and Robert E Schapire. A generalization of principal components analysis to the exponential family. In *Advances in neural information processing systems*, pages 617–624, 2002.

Alexander N Gorban, Balázs Kégl, Donald C Wunsch, Andrei Y Zinovyev, et al. *Principal manifolds for data visualization and dimension reduction*, volume 58. Springer, 2008.

Fang Han and Han Liu. High dimensional semiparametric scale-invariant principal component analysis. *IEEE transactions on pattern analysis and machine intelligence*, 36(10):2016–2032, 2014.

Thomas Hofmann. Probabilistic latent semantic indexing. In *ACM SIGIR Forum*, volume 51, pages 211–218. ACM, 2017.

Iain M Johnstone and Arthur Yu Lu. On consistency and sparsity for principal components analysis in high dimensions. *Journal of the American Statistical Association*, 104(486):682–693, 2009.

Hans-Peter Kriegel, Peer Kröger, Erich Schubert, and Arthur Zimek. A general framework for increasing the robustness of pca-based correlation clustering algorithms. In *International Conference on Scientific and Statistical Database Management*, pages 418–435. Springer, 2008.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788, 1999.

Han Liu, Fang Han, Ming Yuan, John Lafferty, and Larry Wasserman. The nonparanormal skeptic. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012*, volume 2, pages 1415–1422, 2012. ISBN 9781450312851.

Panos P Markopoulos, George N Karystinos, and Dimitris A Pados. Optimal algorithms for $l\_\{1\}$-subspace signal processing. *IEEE Transactions on Signal Processing*, 62(19):5046–5058, 2014.

Jonathan K Pritchard, Matthew Stephens, and Peter Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959, 2000.

Ludger Rüschendorf. Mathematical risk analysis. *Springer Ser. Oper. Res. Financ. Eng. Springer, Heidelberg*, 2013.

Max A Woodbury and Kenneth G Manton. A new procedure for analysis of medical classification. *Methods of Information in Medicine*, 21(04):210–220, 1982.