Communications for Statistical Applications and Methods
2022, Vol. 29, No. 6, 721–733

https://doi.org/10.29220/CSAM.2022.29.6.721
Print ISSN 2287-7843 / Online ISSN 2383-4757

Comprehensive studies of Grassmann manifold optimization and sequential candidate set algorithm in a principal fitted component model

Chaeyoung Lee[a], Jae Keun Yoo[1,a]

[a]Department of Statistics, Ewha Womans University, Korea

Abstract

In this paper we compare parameter estimation by Grassmann manifold optimization and sequential candidate set algorithm in a structured principal fitted component (PFC) model. The structured PFC model extends the form of the covariance matrix of a random error to relieve the limits that occur due to too simple form of the matrix. However, unlike other PFC models, structured PFC model does not have a closed form for parameter estimation in dimension reduction which signals the need of numerical computation. The numerical computation can be done through Grassmann manifold optimization and sequential candidate set algorithm. We conducted numerical studies to compare the two methods by computing the results of sequential dimension testing and trace correlation values where we can compare the performance in determining dimension and estimating the basis. We could conclude that Grassmann manifold optimization outperforms sequential candidate set algorithm in dimension determination, while sequential candidate set algorithm is better in basis estimation when conducting dimension reduction. We also applied the methods in real data which derived the same result.

Keywords: Grassmann manifold, principal fitted component, semi-parametric dimension reduction, sequential fitting

1. Introduction

As the scale of data researchers utilize gets bigger and bigger, dimension reduction has become an emerging issue when analyzing data. So far principal component analysis has been the most widely used method. When the primary interest for a regression $Y|\mathbf{X} \in \mathbb{R}^p$ is given in the dimension reduction of $\mathbf{X}$, the principal component analysis utilizes information abstracted from the marginal distribution of $\mathbf{X}$ alone and does not relate to the response Y. The so-called principal fitted component (PFC) (Cook, 2007) should be one of its alternatives, which cooperates the relation between Y and $\mathbf{X}$ for reducing the dimension of $\mathbf{X}$. For other methodologies to reduce the dimension of $\mathbf{X}$, readers are recommended to read Yoo (2016). The PFC is a model-based dimension reduction method, and it has three variations, depending on the forms of the covariance matrix of a random error. More details about the variations will be discussed in later sections.

For Chaeyoung Lee and Jae Keun Yoo, this work was supported by the MSIT(Ministry of Science,ICT), Korea, under the High-Potential Individuals Global Training Program (RS-2022-00154879) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation). For Jae Keun Yoo, this work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Korean Ministry of Education (NRF-2021R1F1A1059844).

[1] Corresponding author: Department of Statistics, Ewha Womans University, 11-1 Daehyun-Dong, Seodaemun-Gu, Seoul 03760, Korea. E-mail: peter.yoo@ewha.ac.kr

Published 30 November 2022 / journal homepage: http://csam.or.kr

In this paper, the so-called structured PFC model is primarily focused on. Different from the other two variations, the structured PFC does not have a closed form in estimating a parameter to attain dimension reduction. In Cook (2007), a sequential estimation procedure is originally suggested within a set of candidate vectors for the dimension reduction. However, according to Cook (2007), the estimation must be done through Grassmann manifold optimization. Recently, Adragni *et al.* (2012) developed an R-package for the Grassmann manifold optimization, which makes the estimation under the structured PFC model possible.

So far, any decent numerical studies have never been done for estimation through the manifold optimization and never compared the two estimation approaches under the structured PFC. The purpose of this paper is to compare the estimation performances between sequential candidate set procedure and Grassmann manifold optimization intensively for various forward and inverse regression setups.

The organization of the paper is as follows. In Section 2, various forms of principal fitted component model are explained and in Section 3, Grassmann manifold optimization and sequential candidate set algorithm are explained in detail. The same section also introduce likelihood-ratio test statistics and a way to measure how well dimension reduction is done. Section 4 is devoted to the numerical studies we conducted for various forward and inverse simulated regressions and in Section 5, a real data example is presented. Lastly we summarize our work in Section 6.

For notational convenience, we define that $\boldsymbol{\Sigma} = \text{cov}(\mathbf{X})$ and for a $p \times q$ matrix $\mathbf{B}$, $\mathcal{S}_{\mathbf{B}}$ stands for a subspace spanned by the columns of $\mathbf{B}$.

2. Principal fitted component models

2.1. Isotonic principal fitted component model

Consider the following inverse regression model for $\mathbf{X} \in \mathbb{R}^p$ given $Y = y$:

$$\mathbf{X}_y = \boldsymbol{\mu} + \boldsymbol{\Gamma}\boldsymbol{\nu}_y + \sigma\boldsymbol{\varepsilon}, \tag{2.1}$$

where $\boldsymbol{\mu} \in \mathbb{R}^{p\times d}$, $d < p$ and $\boldsymbol{\Gamma}^{\mathrm{T}}\boldsymbol{\Gamma} = \mathbf{I}_d$. The vector $\boldsymbol{\nu}_y \in \mathbb{R}^d$ is an unknown function of y that is assumed to have a positive definite sample covariance matrix and is centered to have 0 as the mean so that $\sum_y \boldsymbol{\nu}_y = 0$. Also it is assumed that $\boldsymbol{\varepsilon} \sim N(0, \mathbf{I}_p)$.

According to Cook (2007), for $\boldsymbol{\Gamma}$ in model (2.1), we have that

$$Y \perp\!\!\!\perp \mathbf{X}|\boldsymbol{\Gamma}^{\mathrm{T}}\mathbf{X},$$

where $\perp\!\!\!\perp$ stands for statistical independence. In Cook (2007), conditional independence indicates that $\boldsymbol{\Gamma}^{\mathrm{T}}\mathbf{X}$ can replace $\mathbf{X}$ without loss of information on a regression of $Y|\mathbf{X}$. This indicates that the $p \times d$ matrix $\boldsymbol{\Gamma}$ is the key to attain dimension reduction of $\mathbf{X}$, so its recovery should be the primary interest. It is natural that $\boldsymbol{\Gamma}$ is estimated through maximizing the likelihood function of (2.1) .

Then, the maximum likelihood estimator of $\boldsymbol{\Gamma}$ is the first d-largest eigenvectors of $\hat{\boldsymbol{\Sigma}}$, where $\hat{\boldsymbol{\Sigma}}$ is the usual moment estimator of $\boldsymbol{\Sigma}$. Since this reduction coincides the principal component analysis for $\mathbf{X}$ and the random vector $\boldsymbol{\varepsilon}$ assumes to have the identity covariance matrix, the model in (2.1) is called an isotonic principal component model.

However, this reduction does not utilize the relation with $\mathbf{X}$ and the response Y, so it is changed as follows by replacing $\boldsymbol{\nu}_y$ with $\boldsymbol{\beta}\mathbf{f}_y$:

$$\mathbf{X}_y = \boldsymbol{\mu} + \boldsymbol{\Gamma}\boldsymbol{\beta}\mathbf{f}_y + \sigma\boldsymbol{\varepsilon}, \tag{2.2}$$

where $\boldsymbol{\beta} \in \mathbb{R}^{d\times r}$ is unknown and $\mathbf{f}_y \in \mathbb{R}^r$ is a known function of the response with $\sum_y \mathbf{f}_y = 0$. For the choices for $\mathbf{f}_y$, Cook (2007) suggests one or combinations of Y, Y^2, $\exp(Y)$ and q-dimensional dummy variables constructed by its categorization with $(q+1)$ levels. Then, again, for model (2.2), we have that $Y \perp\!\!\!\perp \mathbf{X}|\boldsymbol{\Gamma}^{\mathrm{T}}\mathbf{X}$, and $\boldsymbol{\Gamma}$ is estimated by maximizing its likelihood function and the maximum likelihood estimator of $\boldsymbol{\Gamma}$ under model (2.2) is the first d-largest eigenvectors of $\hat{\boldsymbol{\Sigma}}_{\mathrm{fit}} = \mathbb{X}^{\mathrm{T}}\mathbf{P}_F\mathbb{X}/n$, where $\mathbb{X}$ and $\mathbb{F}$ stand for the matrices with stacking $(\mathbf{X}_y - \bar{\mathbf{X}})^{\mathrm{T}}$ and $\mathbf{f}_y^{\mathrm{T}}$ as rows, respectively, and $\mathbf{P}_F = \mathbb{F}(\mathbb{F}^{\mathrm{T}}\mathbb{F})^{-1}\mathbb{F}^{\mathrm{T}}$. The model in (2.2) is called an isotonic principal fitted component (PFC) model, because $\mathbf{f}_y$ is utilized when estimating $\boldsymbol{\Gamma}$.

2.2. Structured principal fitted component

The isotonic principal fitted component model may be limited in its use due to the simple structure of the covariance matrix of $\boldsymbol{\varepsilon}$, and therefore it can be extended as follows:

$$\mathbf{X}_y = \boldsymbol{\mu} + \boldsymbol{\Gamma}\boldsymbol{\beta}\mathbf{f}_y + \boldsymbol{\Gamma}\boldsymbol{\Omega}^{\frac{1}{2}}\boldsymbol{\varepsilon} + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0^{\frac{1}{2}}\boldsymbol{\varepsilon}_0, \tag{2.3}$$

where $\boldsymbol{\Omega} \in \mathbb{R}^{d\times d}$ and $\boldsymbol{\Omega}_0 \in \mathbb{R}^{(p-d)\times(p-d)}$ are full-rank positive-definite matrices with $\boldsymbol{\Omega}^{1/2}\boldsymbol{\Omega}^{1/2} = \boldsymbol{\Omega}$ and $\boldsymbol{\Omega}_0^{1/2}\boldsymbol{\Omega}_0^{1/2} = \boldsymbol{\Omega}_0$. Also $\boldsymbol{\Gamma}_0 \in \mathbb{R}^{p\times(p-d)}$ is an orthogonal complement of $\boldsymbol{\Gamma}$ with $\boldsymbol{\Gamma}_0^{\mathrm{T}}\boldsymbol{\Gamma} = 0$ and $\boldsymbol{\Gamma}_0^{\mathrm{T}}\boldsymbol{\Gamma}_0 = \mathbf{I}_{(p-d)}$. The two random errors, $\boldsymbol{\varepsilon} \in \mathbb{R}^d$ and $\boldsymbol{\varepsilon}_0 \in \mathbb{R}^{(p-d)}$, are assumed to be independent and normally distributed with zero means and the identity covariance matrices.

The model in (2.3) is equivalently expressed as follows:

$$\mathbf{X}_y = \boldsymbol{\mu} + \boldsymbol{\Gamma}\boldsymbol{\beta}\mathbf{f}_y + \boldsymbol{\varepsilon}^*,$$

where $\boldsymbol{\varepsilon}^* \in \mathbb{R}^p \sim N(0, \boldsymbol{\Gamma}\boldsymbol{\Omega}\boldsymbol{\Gamma}^{\mathrm{T}} + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0\boldsymbol{\Gamma}_0^{\mathrm{T}})$.

Under model (2.3), we still have that $Y \perp\!\!\!\perp \mathbf{X}|\boldsymbol{\Gamma}^{\mathrm{T}}\mathbf{X}$. However, unfortunately, we do not have a closed form for the maximum likelihood estimator of $\boldsymbol{\Gamma}$ here and further details regarding this fact will be discussed in later sections. The model (2.3) is called a *structured principal fitted component model*, which is the primary model discussed in this paper.

2.3. Unstructured principal fitted component

The structured PFC model is generalized by releasing the restriction of the covariance matrix of the random error. Suppose that $\mathbf{X}_y$ has the following PFC model with $\mathrm{cov}(\boldsymbol{\varepsilon}) = \boldsymbol{\Delta}$, which is unknown and positive-definite:

$$\mathbf{X}_{\mathbf{y}} = \boldsymbol{\mu} + \boldsymbol{\Gamma}\boldsymbol{\beta}\mathbf{f}_y + \boldsymbol{\varepsilon}. \tag{2.4}$$

Different from the first two PFC models, we have $Y \perp\!\!\!\perp \mathbf{X}|(\boldsymbol{\Delta}^{-1}\boldsymbol{\Gamma})^{\mathrm{T}}\mathbf{X}$, so $(\boldsymbol{\Delta}^{-1}\boldsymbol{\Gamma})^{\mathrm{T}}\mathbf{X}$ replaces $\mathbf{X}$ without loss of information on $Y|\mathbf{X}$. The maximum likelihood estimator $\hat{\boldsymbol{\Delta}}$of $\boldsymbol{\Delta}$ is $\hat{\boldsymbol{\Sigma}}_{\mathrm{res}}$, which is equal to $\hat{\boldsymbol{\Sigma}} - \hat{\boldsymbol{\Sigma}}_{\mathrm{fit}}$. Cook and Forzani (2009) show that $\boldsymbol{\Delta}^{-1}\boldsymbol{\Gamma}$ is estimated by the the first d-largest eigenvectors of $\hat{\boldsymbol{\Delta}}^{-1/2}\hat{\boldsymbol{\Sigma}}_{\mathrm{fit}}\hat{\boldsymbol{\Delta}}^{-1/2}$. Since the covariance structure of $\boldsymbol{\varepsilon}$ is fully free, the model in (2.4) is called an unstructured principal fitted component. Like the isotonic PFC, the unstructured PFC has a closed form to estimate the parameters required for dimension reduction.

3. Estimation in structured principal fitted component model

3.1. Estimation of structured principal fitted component model and its benefit

According to Cook (2007), the log-likelihood $\mathcal{L}_{\text{sPFC}}$ under the structured PFC model in (2.3) is partially maximized for $\boldsymbol{\Gamma}$ as follows:

$$\mathcal{L}_{\text{sPFC}}(\boldsymbol{\Gamma}) = \left(-\frac{n}{2}\right)\log\left|\boldsymbol{\Gamma}_0^{\text{T}}\hat{\boldsymbol{\Sigma}}\boldsymbol{\Gamma}_0\right| - \frac{n}{2}\log\left|\boldsymbol{\Gamma}^{\text{T}}\hat{\boldsymbol{\Sigma}}_{\text{res}}\boldsymbol{\Gamma}\right|. \tag{3.1}$$

As observed from (3.1), the maximum likelihood estimation of $\boldsymbol{\Gamma}$ is clearly associated with both $\hat{\boldsymbol{\Sigma}}$ and $\hat{\boldsymbol{\Sigma}}_{\text{res}}$ simultaneously. This is why there is no closed form for $\boldsymbol{\Gamma}$. Still, our goal is to estimate $\boldsymbol{\Gamma}$ by maximizing (3.1). In other words, our goal is to restore $\boldsymbol{\Gamma}_0$ to minimize the partial log-likelihood in (3.1). When we assume that we know $\boldsymbol{\Gamma}_0$, it should be noted that there is no unique orthogonal complement $\boldsymbol{\Gamma}$ of $\boldsymbol{\Gamma}_0$. Therefore when the column space $\mathcal{S}_{\boldsymbol{\Gamma}}$ of $\boldsymbol{\Gamma}$ is the main interest, any orthogonal complement of $\boldsymbol{\Gamma}_0$ should be fine, because it spans the same subspace. Because of this no uniqueness of $\boldsymbol{\Gamma}$, our new primary target becomes $\mathcal{S}_{\boldsymbol{\Gamma}}$ instead of $\boldsymbol{\Gamma}$ itself and this is why $\boldsymbol{\Gamma}$ should be estimated in a manifold.

The structured principal fitted component model has advantage over the isotonic PFC model, because the error covariance matrix that the latter assumes is too simple. In this sense, the unstructured PFC model might seem like the most appealing choice, but this model requires the inverse of $\hat{\boldsymbol{\Sigma}}_{\text{res}}$ which is the maximum likelihood estimator of $\boldsymbol{\Delta}$. Therefore, if $\boldsymbol{\Delta}$ is not well-estimated due to its relatively small sample size compared to the number of predictors, the accuracy in dimension reduction should be problematic. On the other hand, since the covariance of the random error in the structured PFC model is not estimable, there is no need to estimate it when we consider the structured PFC model. This saves the number of parameters in estimation, which a clear advantage over the unstructured PFC model. The so-called envelope model developed by Cook *et al.* (2010) is based on the structured PFC model, and there are many descendants of the envelope model (Cook, 2018). This becomes a strong statement why the structured PFC model and its estimation are important to investigate.

3.2. Grassmann manifold optimization

Grassmann manifold optimization is specialized for orthogonally constrained optimization problems. Let's say the objective of an optimization is maximizing $f(\mathbf{U})$ where $\mathbf{U}$ is a $(p \times d), d < p$ semi orthogonal matrix where $\mathbf{U}^{\text{T}}\mathbf{U} = \mathbf{I}_d$. The function f is invariant to right orthogonal transformation, $f(\mathbf{UO}) = f(\mathbf{U})$, for any $d \times d$ orthogonal matrix $\mathbf{O}$. This invariant characteristic of f allows any orthonormal basis of the subspace spanned by the columns of $\mathbf{U}$ to be the argument of f. Therefore the optimization can be handled over the set of d-dimensional linear subspace of $\mathbb{R}^p$ and this collection of linear subspace compose a Grassmann manifold. A function that is optimized over Grassmann manifold is parametrized and maximized in terms of $d(p-d)$ angles. Although optimization in such setting may seem difficult, the optimization can be done very smoothly with efficient algorithms using directional derivatives on the manifold with the help of the geometry of Grassmann manifold (Gallivan *et al.*, 2003).

We will explain optimization in Grassmann manifold in more detail with an example of approximating n centered p-dimensional vectors $\mathbf{X}_1, \ldots, \mathbf{X}_n$, so that $\sum \mathbf{X}_{i=1}$ is 0 with corresponding vectors $\mathbf{B}_i$ that lie in d-dimensional subspace of $\mathbb{R}^p$. In such setting, $\mathbf{B}_i$ can be expressed as $\mathbf{B}_i = \mathbf{UC}_i$ where $\mathbf{U}$ is a semi-orthogonal basis matrix for d-dimensional subspace and $\mathbf{C}_i$ is the coordinates of $\mathbf{B}_i$ in terms of $\mathbf{U}$. Then the goal of this example will be minimizing $\sum_{i=1} \|\mathbf{X}_i - \mathbf{UC}_i\|^2$ and as it is minimized

over $\mathbf{C}_i$ with $\mathbf{U}$ fixed, it can be rewritten as (3.2). Therefore, maximizing trace($\mathbf{U}\mathbf{U}^{\mathrm{T}}\hat{\mathbf{\Sigma}}$) where $\hat{\mathbf{\Sigma}}$ is the sample covariance matrix of $\mathbf{X}_i$'s is equivalent to minimizing $\sum_{i=1} \|\mathbf{X}_i - \mathbf{U}\mathbf{C}_i\|^2$.

$$\min_{C_i} \sum_{i=1} \|\mathbf{X}_i - \mathbf{U}\mathbf{C}_i\|^2 = \sum_{i=1} \left\|\mathbf{X}_i - \mathbf{U}\mathbf{U}^{\mathrm{T}}\mathbf{X}_i\right\|^2. \tag{3.2}$$

In this example, function f is also invariant as $f(\mathbf{U}) = \text{trace}(\mathbf{U}\mathbf{U}^{\mathrm{T}}\hat{\mathbf{\Sigma}})$. Our goal of this optimization can be easily achieved as it is known that $f(\mathbf{U})$ is maximized when the first d-largest eigenvectors of $\hat{\mathbf{\Sigma}}$ are chosen as the columns of $\mathbf{U}$. However in more complicated settings, the optimization needs somewhat complicated numerical computations.

Under structured PFC model, we can consider the Equation in (3.1) as the objective function since we estimate $\mathbf{\Gamma}$ through maximizing the partial log-likelihood. Therefore, the Grassmann manifold becomes the parameter space for $\mathcal{S}_{\mathbf{\Gamma}}$.

The basic gradient algorithm of optimization without any restriction is simple and intuitive, but it has the risk of falling into local optimums, so there is a modified version of the basic algorithm, stochastic gradient algorithm. It contains various options including the option of applying simulated annealing to the optimization process to avoid converging to local optimum especially when $f(\mathbf{U})$ is multimodal. Following Adragni *et al.* (2012), we will first review the basic gradient algorithm briefly and move on to stochastic gradient algorithm which are the two referenced algorithms for the GrassmannOptim R-package.

3.2.1. Basic gradient algorithm

When $\mathbf{U}$ is a $p \times d$ semi-orthogonal matrix, $f(\mathbf{U})$ is an invariant function and $\mathcal{S}_{\mathbf{U}} = \{\mathbf{U}\mathbf{O}|\mathbf{O} \in \mathbb{R}^{d\times d}, \mathbf{O}\mathbf{O}^{\mathrm{T}} = \mathbf{I}_d\} \in \mathcal{G}_{(d,p)}$. In this setting, the goal is to find the subspace $\hat{\mathcal{S}}_{\mathbf{U}} = \text{argmax} f(\mathbf{U})$ when the parameter space for $\mathcal{S}_{\mathbf{U}}$ is the Grassmann manifold. Three matrices need to be defined beforehand. $\mathbf{V}$ is a $(p \times (p-d))$ semi-orthogonal matrix which is a completion of $\mathbf{U}$ so that $\mathbf{Q} = (\mathbf{U}, \mathbf{V})$ is a $(p \times p)$ orthogonal matrix. The directional derivative of f, $\mathbf{B} = (\triangledown f(\mathbf{U}))^{\mathrm{T}}\mathbf{V}$, is a $(p \times (p-d))$ matrix and $\mathbf{A} = \begin{bmatrix} \mathbf{0}_d & \mathbf{B} \\ -\mathbf{B}^{\mathrm{T}} & \mathbf{0}_{p-d} \end{bmatrix}$, a $(p \times p)$ skew-symmetric matrix, is also needed. The basic gradient algorithm proceeds as the following.

1. We begin with the initial matrix $\mathbf{Q}$ at $t = 0$.

2. Repeat step 2 until $\|\mathbf{B}\| < \epsilon$.

   (a) Get directional derivative $\mathbf{B}$ using $\mathbf{Q}$. Using $\mathbf{B}$, calculate matrix $\mathbf{A}$.

   (b) Update $\mathbf{Q}$ to $\mathbf{Q}_{t+1} = \mathbf{Q}_t\{\delta\mathbf{A}\}$, $\delta \in (0, 1)$ so that $f(\mathbf{U}_{t+1}) > f(\mathbf{U}_t)$.

3. Then $\hat{\mathcal{S}}_U$ is the subspace spanned by the first d-columns of the lastly updated $\mathbf{Q}$.

When determining a specific matrix as the initial matrix is difficult, the algorithm will randomly set the initial matrix. If there is no analytical expression for $\mathbf{B}$, $\mathbf{B}$ can be approximated as $\alpha_{ij} = \{f(\tilde{\mathbf{U}}) - f(\mathbf{U})\}/\epsilon$, $i = 1, \ldots, d$, $j = 1, \ldots, p-d$. $\tilde{\mathbf{U}} = \mathbf{Q}\exp\{\epsilon\boldsymbol{E}_{ij}\}$ where $\boldsymbol{E}_{ij}$ is a (p, p) skew-symmetric matrix of all 0 except 1 in (i, j) and (j, i). Also, for a more thorough inspection in step 2, we can generate several candidates of $\mathbf{Q}_{t+1}$ by choosing a sequence of values for δ within the range of $(0, 1)$ at each iteration instead of choosing only one value for δ at a time.

3.2.2. Stochastic gradient algorithm

The alternative of the basic algorithm is the stochastic gradient algorithm which is a modified algorithm to lower the possibility of converging into local optimums by including additional steps and options.

It consists of two iterative loops; the inner loop and the outer loop. The main goal of the inner loop is to avoid local optimum by searching the neighborhood of the candidate for maximum number of iterations. The outer loop acts as a cooling procedure of the annealing process. At first, we start with a high temperature and cools the temperature down according to the cooling ratio until the temperature reaches a preset threshold. This process allows the inner loop to search more thoroughly in various temperatures. Since the outer loop is included, initial temperature T_0 and the cooling ratio τ is additionally required before getting into the algorithm. Stochastic gradient algorithm proceeds as the following.

1. We start with the starting subspace $\mathbf{U}_t$ and repeat step 1 M times.
   (a) Compute the directional derivative $\mathbf{B}$.
   (b) Generate a $d \times (p-d)$ matrix $\mathbf{W}$ of w_{ij} where w_{ij}'s are $d(p-d)$ independent realizations from standard normal distribution.
   (c) Get candidate value $\mathbf{Y}$ following $\mathbf{Q}_{t+1} = \mathbf{Q}_t\{\delta\mathbf{A}\}$ starting from $\mathbf{Q}_t = (\mathbf{U}_t, \mathbf{V}_t)$ in the direction of $\mathbf{B} + \sqrt{T_t}\mathbf{W}$.
   (d) Set $\mathbf{U}_{t+1} = \mathbf{Y}$ with prob $\min\{\exp(d_f/T_t), 1\}$. Else, set $\mathbf{U}_{t+1} = \mathbf{U}_t$ and go back to step 1-(a).
2. Decrease T_t to $T_{t+1} = d_f/\tau,\ \tau > 1$.
3. Let $t = t+1$ and go back to step 1 if T_{t+1} is higher than the threshold. Stop otherwise.

3.3. Sequential candidate set algorithm

Under structured PFC model, $\hat{\mathcal{S}}_{\mathbf{\Gamma}}$ can also be estimated by sequential candidate set algorithm. If the likelihood is over a multimodal surface, a simple gradient optimization process may not lead to the optimal solution. In this case, sequential candidate set algorithm may lower this possibility.

We will start with briefly going over sPFC_{PC} method which is a method where $\mathcal{L}_{\text{sPFC}}$ is evaluated according to PC directions. In sPFC_{PC}, the PC directions make up one possible candidate set. Therefore, we consider all possible subsets of d-sample PC directions $\hat{\boldsymbol{\gamma}}_{(1)}, \ldots, \hat{\boldsymbol{\gamma}}_{(d)}$ and calculate $\mathcal{L}_{\text{sPFC}}$ for each subsets. From all the possible subsets, we choose the subset with the highest $\mathcal{L}_{\text{sPFC}}$. However when there are too many combinations of possible subsets, this method may be hard to implement. An alternative method is the sequential candidate set algorithm which handles this problem by choosing PC directions sequentially in each step instead of going over all possible subsets at once. Sequential candidate set algorithm proceeds as the following.

1. We find $\hat{\boldsymbol{\gamma}}_{(1)} = \text{argmax}\ \mathcal{L}_{\text{sPFC}}(\mathbf{h})$ where the likelihood maximum is calculated over $(p \times 1)$ vector $\mathbf{h}$ in PC candidate set $\mathbf{A} = \{\hat{\boldsymbol{\gamma}}_j, j = 1, \ldots, p\}$.
2. Then we find $\hat{\boldsymbol{\gamma}}_{(2)} = \text{argmax}\ \mathcal{L}_{\text{sPFC}}(\hat{\boldsymbol{\gamma}}_{(1)}, \mathbf{h})$, where the likelihood maximum is calculated over $(p \times 1)$ vector $\mathbf{h}$ in the reduced PC candidate set $\mathbf{A} - \{\hat{\boldsymbol{\gamma}}_{(1)}\}$.
3. We continue this process until the final maximization $\hat{\boldsymbol{\gamma}}_{(d)}$=argmax $\mathcal{L}_{\text{sPFC}}(\hat{\boldsymbol{\gamma}}_{(1)}, \ldots, \hat{\boldsymbol{\gamma}}_{(d-1)}, \mathbf{h})$ covers candidate set $\mathbf{A} - \{\hat{\boldsymbol{\gamma}}_{(1)}, \ldots, \hat{\boldsymbol{\gamma}}_{(d-1)}\}$.

Also, considering the eigenvectors of $\hat{\mathbf{\Sigma}}_{\text{fit}}$ and $\hat{\mathbf{\Sigma}}_{\text{res}}$ which are PFC directions and *residual component (RC) directions* respectively as candidate sets is also possible. As a result, there are three possible candidate sets that can be considered; sets consisted of the PC, PFC, and RC directions. If the estimator is evaluated over all three candidate sets, the method is called sPFC_{all}. Since sPFC_{all} contains all possible candidate sets, the maximum likelihood by sPFC_{all} is bigger or, at least, equal to those by $\hat{\mathbf{\Sigma}}$, $\hat{\mathbf{\Sigma}}_{\text{fit}}$ and $\hat{\mathbf{\Sigma}}_{\text{res}}$. Naturally, we will use sPFC_{all} in our numerical studies as default for the best results.

3.4. Dimension estimation

When estimating $\mathbf{\Gamma}$, d is required. Once d is fixed, $\mathbf{\Gamma}$ is estimated through Grassmann optimization or sequential candidate algorithm. In practice, however, d is unknown and therefore needs to be estimated. Following Yoo (2018), this estimation is done through a series of hypothesis tests and this process is called sequential dimension testing. When $m = 0, \ldots, p-1$, it starts by testing $H_0 : d = m$ versus $H_1 : d = p$ with $m = 0$. If H_0 is rejected from this first testing, we do the same testing after increasing m to 1. We continue this process until H_0 is not rejected for the first time and the final m is the determined dimension. For instance, if the null hypothesis $H_0 : d = 1$ is not rejected for the first time, we conclude that $\hat{d} = 1$. Define that $\mathcal{L}^{\text{full}} = -n/2 \log|\hat{\mathbf{\Sigma}}_{\text{res}}|$ and $\mathcal{L}^{m} = -n/2 \log|\hat{\mathbf{\Gamma}}^{\mathrm{T}} \hat{\mathbf{\Sigma}}_{\text{res}} \hat{\mathbf{\Gamma}}| - n/2 \log|\hat{\mathbf{\Gamma}}_0^{\mathrm{T}} \hat{\mathbf{\Sigma}} \hat{\mathbf{\Gamma}}_0|$. Then the statistic of the test for $m = 0, \ldots, p-1$ is

$$\hat{\lambda}^m = -2\left(\mathcal{L}^{\mathrm{m}} - \mathcal{L}^{\text{full}}\right) \sim \chi^2_{p(p-m)}.$$

4. Numerical simulation

In this section, we summarize the results of the numerical simulations we conducted to compare the parameter estimation of Grassmann manifold optimization and sequential candidate set algorithm. We focused on comparing the determined dimension by sequential dimension testing with $\alpha = 0.05$. To better understand how well the two methods estimate $\hat{\mathcal{S}}_{\mathbf{\Gamma}}$ compared to the true $\mathcal{S}_{\mathbf{\Gamma}}$, we also calculated the trace correlation of $\hat{\mathcal{S}}_{\mathbf{\Gamma}}$ and $\mathcal{S}_{\mathbf{\Gamma}}$.

To evaluate the estimation of how well $\mathbf{\Gamma} \in \mathbb{R}^{p\times d}$ is estimated by its estimate $\hat{\mathbf{\Gamma}} \in \mathbb{R}^{p\times d}$, we calculated trace correlation

$$r = \sqrt{\frac{1}{d}\text{trace}\left(\hat{\mathbf{\Gamma}}\left(\hat{\mathbf{\Gamma}}^{\mathrm{T}}\hat{\mathbf{\Gamma}}\right)^{-1}\hat{\mathbf{\Gamma}}^{\mathrm{T}}\mathbf{\Gamma}\left(\mathbf{\Gamma}^{\mathrm{T}}\mathbf{\Gamma}\right)^{-1}\mathbf{\Gamma}^{\mathrm{T}}\right)}$$

for all replications. The trace correlation includes information of the correlation between $\mathbf{\Gamma}$ and $\hat{\mathbf{\Gamma}}$. Thus, bigger r indicates that the two values are closer to each other than a smaller r.

4.1. Model construction

Three models in total were considered for numerical simulation. We set $p = 5$ and $n = 100$ for all models. Model 1 is under inverse regression setting as the following

$$\text{Model 1} \ : \mathbf{X}_y = \mathbf{\Gamma}\boldsymbol{\beta} y + \boldsymbol{\varepsilon}^*,$$

where $\mathbf{\Gamma} = (1,0,0,0,0)(0,1,0,0,0)^{\mathrm{T}}$, $\boldsymbol{\beta} = (1,1)^{\mathrm{T}}$, and $\boldsymbol{\varepsilon}^* \in \mathbb{R}^p \sim N(0, \mathbf{\Gamma}\mathbf{\Omega}\mathbf{\Gamma}^{\mathrm{T}} + \mathbf{\Gamma}_0\mathbf{\Omega}_0\mathbf{\Gamma}_0^{\mathrm{T}})$. $\mathbf{\Gamma}_0$ is an orthogonal complement of $\mathbf{\Gamma}$. Also, we set $\mathbf{\Omega} = \sigma I_d$, $\mathbf{\Omega}_0 = \sigma_0 I_{(p-d)}$ and y's in model 1 were generated as random variables that follow $N(0, \sigma_y)$.

In the case of model 1, we varied the values of σ, σ_0, and σ_y for the numerical simulation. σ and σ_0 construct the error term of the generated $\mathbf{X}_y$ and σ_y determines the variance of the randomly

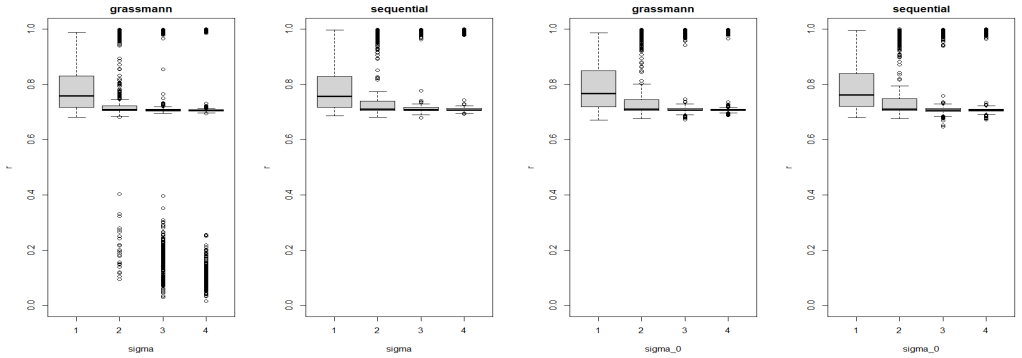

Figure 1: *M1: When σ is 1, 2, 3, 4 while $\sigma_y = \sigma_0 = 1$(first two plots) and σ_0 is 0.6, 1.5, 3, 4 while $\sigma = \sigma_y = 1$(last two plots).*

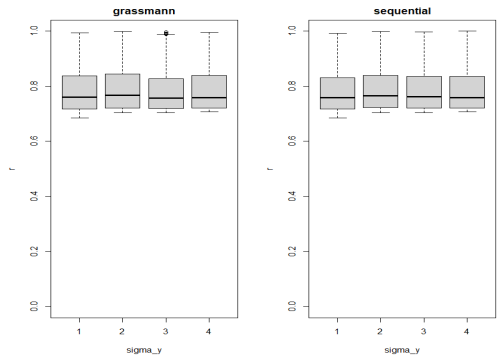

Figure 2: *M1: When σ_y is 1, 2, 3, 4 while $\sigma = \sigma_0 = 1$.*

generated y. For σ and σ_y we tested using values $1, 2, 3, 4$ and with σ_0, we used $0.6, 1.5, 3, 4$ for a broader observation.

Unlike model 1, model 2 and model 3 are under forward regression setting which were generated as the following

$$\text{Model 2} \ : Y = 1.5(X_1 + 5)(X_2 + X_3 + 2) + 0.5\varepsilon,$$
$$\text{Model 3} \ : Y = X_1(X_1 + X_2 + X_3 + 1) + 0.5\varepsilon,$$

where $\varepsilon \sim N(0, \sigma^2)$. $\mathbf{X}$'s in model 2 and model 3 were generated by $X_1 = W_1$, $X_2 = V_1 + W_2/2$, $X_3 = -V_1 + W_2/2$, $X_4 = V_2 + V_3$, and $X_5 = -V_2 + V_3$ when $\mathbf{V} = (V_1, V_2, V_3)^{\mathrm{T}}$ and $\mathbf{W} = (W_1, W_1)^{\mathrm{T}}$. $\mathbf{V}$ and $\mathbf{W}$ were drawn both from normal distribution and non-normal distribution to see if there are noticeable differences in the two cases. For the normal case, $\mathbf{V}$ and $\mathbf{W}$ were both generated from $N(0, 1)$ and for the non-normal case, $\mathbf{V}$ and $\mathbf{W}$ were generated from t_6, Ga(0.25, 1) respectively. We varied the value of σ using values $1, 2, 3, 4$ for the numerical simulation.

4.2. Numerical simulation results

All three models were generated 1,000 times and the generated data was fitted using the structured PFC model in (2.3). For model fitting, we set $\mathbf{f}_y = (y, y^2, y^3)^{\mathrm{T}}$ and $d = 2$.

Table 1: M1: Percentages of determined $\hat{d}$ by sequential dimension test when $\sigma = 1, 2, 3, 4$ and $\sigma_y = \sigma_0 = 1$

	Grassmann				Sequential			
	$\hat{d} = 0$	$\hat{d} = 1$	$\hat{d} = 2$	$\hat{d} > 2$	$\hat{d} = 0$	$\hat{d} = 1$	$\hat{d} = 2$	$\hat{d} > 2$
$\sigma = 1$	0.0	21.8	78.1	0.10	0.0	99.9	0.1	0.0
$\sigma = 2$	0.1	17.4	78.9	3.60	0.0	99.4	0.6	0.0
$\sigma = 3$	1.4	14.2	69.7	14.7	1.3	98.3	0.4	0.0
$\sigma = 4$	7.6	13.0	62.7	16.7	5.5	93.5	0.9	0.1

Table 2: M1: Percentages of determined $\hat{d}$ by sequential dimension test when $\sigma_y = 1, 2, 3, 4$ and $\sigma = \sigma_0 = 1$

	Grassmann				Sequential			
	$\hat{d} = 0$	$\hat{d} = 1$	$\hat{d} = 2$	$\hat{d} > 2$	$\hat{d} = 0$	$\hat{d} = 1$	$\hat{d} = 2$	$\hat{d} > 2$
$\sigma_y = 1$	0	22.1	77.8	0.1	0	99.7	0.3	0
$\sigma_y = 2$	0	9.2	90.7	0.1	0	99.7	0.3	0
$\sigma_y = 3$	0	6.2	93.6	0.2	0	99.6	0.4	0
$\sigma_y = 4$	0	6.2	93.6	0.2	0	99.7	0.3	0

Table 3: M1: Percentages of determined $\hat{d}$ by sequential dimension test when $\sigma_0 = 0.6, 1.5, 3, 4$ and $\sigma = \sigma_y = 1$

	Grassmann				Sequential			
	$\hat{d} = 0$	$\hat{d} = 1$	$\hat{d} = 2$	$\hat{d} > 2$	$\hat{d} = 0$	$\hat{d} = 1$	$\hat{d} = 2$	$\hat{d} > 2$
$\sigma_0 = 0.6$	0	17.1	82.8	0.1	0	99.5	0.50	0.0
$\sigma_0 = 1.5$	0	15.3	84.3	0.4	0	99.9	0.10	0.0
$\sigma_0 = 3$	0	16.0	83.8	0.2	0	94.5	0.53	0.2
$\sigma_0 = 4$	0	17.2	82.7	0.1	0	85.5	14.0	0.5

We compared the numerical simulation results of the three models using trace correlation values and the results of sequential dimension testing.

4.2.1. Model 1: Normal cases in inverse regression setting

Box plots in Figure 1 and Figure 2 summarize the simulation results of the trace correlation obtained from model 1. Overall, sequential candidate set algorithm is slightly better but does not have noticeable difference with Grassmann manifold optimization. Trace correlation obtained from the two methods are mostly around 0.8 and show almost identical trends. Among σ_y, σ_0, and σ, σ showed the biggest fluctuation in the trace correlation value especially when Grassmann optimization was used in estimation. Trace correlation was highest when σ is 1 and for 2, 3, and 4, the value decreased a little for both methods. Therefore, we can conclude that the varying the distribution of $\mathbf{Y}$ does not have much influence in estimation compared to varying the errors of $\mathbf{X}_y$ since the varying values of σ_y do not make much significant difference.

The results of the sequential dimension test for model 1 is in Table 1–3. The biggest difference between the two methods is that sequential candidate set algorithm tends to underestimate in all cases. On the other hand, estimation by Grassmann manifold optimization determined $\hat{d}$ as 2 which is the true value with at least 60 percent in all cases. The best result with Grassmann optimization is when $\sigma_y = 4$, where almost 94 percent of all trials determined $\hat{d}$ as 2. However, the percent drops to 77 when $\sigma_y = 1$, which indicates that changing the value of σ_y can lead to some fluctuation in the dimension test results when compared with the other cases.

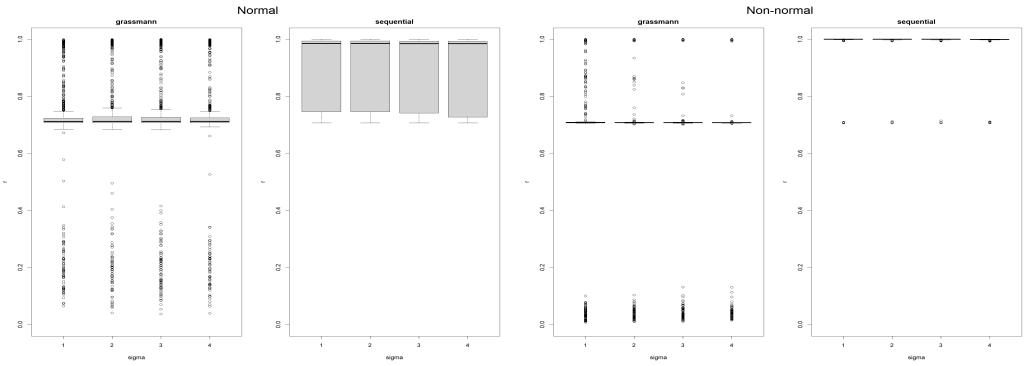

Figure 3: *M2: box plot of trace correlation when σ is 1, 2, 3, 4 in normal case(first two plots) and in non-normal case(last two plots).*

Table 4: M2:Percentages of determined $\hat{d}$ by sequential dimension test when σ is $1, 2, 3, 4$ in normal case and in non-normal case

	Normal								Non-normal							
	Grassmann				Sequential				Grassmann				Sequential			
	$\hat{d}=0$	$\hat{d}=1$	$\hat{d}=2$	$\hat{d}>2$	$\hat{d}=0$	$\hat{d}=1$	$\hat{d}=2$	$\hat{d}>2$	$\hat{d}=0$	$\hat{d}=1$	$\hat{d}=2$	$\hat{d}>2$	$\hat{d}=0$	$\hat{d}=1$	$\hat{d}=2$	$\hat{d}>2$
$\sigma=1$	0	1.8	78.7	19.5	0	93.9	6.0	0.1	0	0.2	34.5	65.3	0	49.3	49.8	0.9
$\sigma=2$	0	2.5	81.6	15.9	0	93.3	6.6	0.1	0	1.2	47.9	50.9	0	60.5	38.9	0.6
$\sigma=3$	0	2.6	83.4	14.0	0	93.7	6.3	0.0	0	1.8	54.2	44.0	0	68.8	30.8	0.4
$\sigma=4$	0	3.5	83.8	12.7	0	94.0	6.0	0.0	0	1.6	61.8	36.6	0	72.8	26.6	0.6

4.2.2. Model 2: Normal and non-normal cases in forward regression setting

Box plots in Figure 3 summarize the simulation results obtained from model 2. In model 2, we also compared normal and non-normal cases. In both cases, sequential candidate set algorithm outperformed Grassmann manifold optimization in estimation and was more robust as well. From the similar trace correlation values for all the four different values of σ, we can infer that σ-values are not significant in basis estimation in this case. Also there is little difference between the normal and non-normal cases as we can observe from the box plots.

Table 4 summarizes the percentage of the determined dimensions in normal and non-normal cases of model 2. The performance of normal cases was better showing a similar trend with model 1. Grassmann manifold optimization mostly determined dimension as 2 while sequential candidate set algorithm underestimated the dimensions. Also, as σ increased, the percentage of determining $\hat{d}$ as 2 increased which is similar with the results obtained from model 1. On the other hand, the result of the non-normal case was quite different from the normal case. Although the fact that as σ increase the performance of dimension determination gets better still remains, the percentage of determining $\hat{d}$ as 2 decreased to 34 percent when $\sigma = 1$. Instead, the percentage of overestimation increased to 65 percent. On contrary, when sequential candidate set algorithm was used, performance was better when σ was smaller and when $\sigma = 1$, almost 50 percent of the tests determined $\hat{d}$ as 2 which is the best result obtained with sequential candidate set algorithm.

4.2.3. Model 3: Normal and non-normal cases in forward regression setting

Figure 4 shows the simulation results of the normal and non-normal cases of model 3. Like model 2, sequential candidate set algorithm show better performance in basis estimation than Grassmann

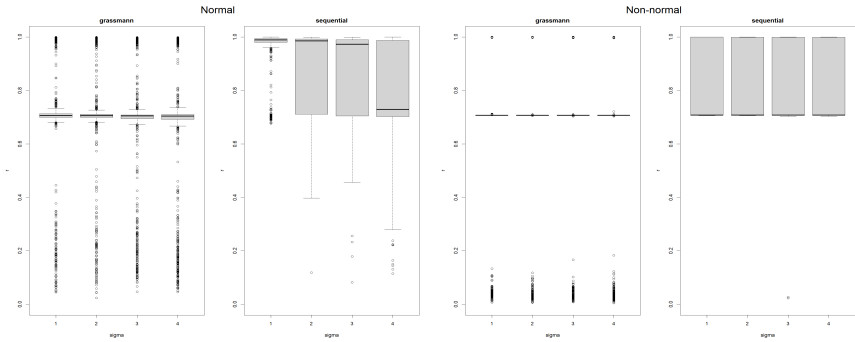

Figure 4: *M3: box plot of trace correlation when σ is 1, 2, 3, 4 in normal case(first two plots) and in non-normal case(last two plots).*

Table 5: M3: Percentages of determined $\hat{d}$ by sequential dimension test when σ is $1, 2, 3, 4$

	Normal								Non-normal							
	Grassmann				Sequential				Grassmann				Sequential			
	$\hat{d}=0$	$\hat{d}=1$	$\hat{d}=2$	$\hat{d}>2$	$\hat{d}=0$	$\hat{d}=1$	$\hat{d}=2$	$\hat{d}>2$	$\hat{d}=0$	$\hat{d}=1$	$\hat{d}=2$	$\hat{d}>2$	$\hat{d}=0$	$\hat{d}=1$	$\hat{d}=2$	$\hat{d}>2$
$\sigma=1$	10.4	10.5	51.9	27.2	9.8	62.5	26.9	0.8	0.0	3.7	79.4	16.9	0.0	92.4	7.0	0.6
$\sigma=2$	25.6	14.3	45.2	14.9	25.3	63.4	11.0	0.3	0.0	5.8	79.8	14.4	0.2	90.9	7.7	1.2
$\sigma=3$	42.9	11.2	39.2	6.7	44.8	51.5	3.5	0.2	2.7	5.2	76.3	15.8	2.4	90.0	6.9	0.7
$\sigma=4$	60.9	9.3	25.0	4.8	62.2	35.7	2.1	0.0	7.9	5.1	69.3	17.7	7.7	85.7	6.1	0.5

optimization. Also the performance is better when σ is smaller which is clearly apparent in the normal case.

Table 5 summarizes the results of the sequential dimension test of model 3. Results were better when Grassmann optimization was used in both normal and non-normal cases. Also for both normal and non-normal cases, the performance of dimension testing was better when σ was smaller. However unlike model 2, the performance of Grassmann optimization was better in non-normal case where at least 70 percent of all trials determined $\hat{d}$ as 2. On the other hand, sequential candidate set algorithm underestimated in most cases as usual. In the normal case, Grassmann optimization underestimated almost 70 percent of the total trials at most when the value of σ got bigger resulting in poorer performance compared to the non-normal case for both methods. As a result, when $\sigma = 4$, the percentage of determining the dimension as 2 dropped to 25 and 35 percent for Grassmann optimization and sequential algorithm respectively.

5. Application on real data: BigMac data

In this section we compared the estimation of Grassmann manifold optimization and sequential candidate set algorithm by applying the two methods to Big Mac data from Enz (1991).

This data includes economic data of $n = 45$ cities from 1991 with 9 predictors; bread, busfare, engsal, engtax, service, teachsal, teachtax, vacdays, and workhrs. They respectively describe minimum labor to purchase one kilogram of bread, lowest cost of 10 kilometers ride on public transportation, annual salary of electrical engineer, tax rate paid by electrical engineer, annual cost of 19 chosen services, salary of primary school teachers, tax rate for primary teachers, average days of vacation in one year, and average hours of work in one year. These predictors are expected to be related and therefore help explain the price of BigMac in each city. The response of this data is the average minutes of labor

Table 6: Comparison of time required in estimation (M2)

Grassmann Optimization	Sequential Candidate test
2.7711 secs	0.1026 secs

Table 7: BigMac data: p-values obtained from sequential dimension testing

	Grassmann Optimization				Sequential Candidate Set Algorithm			
	$H_0 : \hat{d} = 0$	$H_0 : \hat{d} = 1$	$H_0 : \hat{d} > 2$	$\hat{d} \geq 3$	$H_0 : \hat{d} = 0$	$H_0 : \hat{d} = 1$	$H_0 : \hat{d} > 2$	$\hat{d} \geq 3$
p-values	0.02438	0	0	0	0.02440	0.13319	0.36664	0.12305

needed to buy a BigMac in 45 different cities. *The Economist* magazine introduced this concept of BigMac index and implied that the price of BigMac can be used as an index to explain the economy of a country which the 9 predictors are assumed to represent in this dataset.

We used Grassmann manifold optimization and sequential candidate set algorithm to estimate $\mathbf{\Gamma}$ under structured PFC model with $\mathbf{f}_y = (y, y^2, y^3)$. The p-values in Table 7 are obtained from the sequential dimension testing when Grassmann manifold optimization and sequential candidate set algorithm were used to determine $\hat{d}$. Grassmann manifold optimization determined $\hat{d}$ as 3 while sequential candidate algorithm determined $\hat{d}$ as 1. Given the fact that SIR with 9 slices determined $\hat{d}$ as 4, we can conclude that like the numerical simulation result, Grassmann optimization is better in dimension determination. We can also check that sequential candidate set algorithm tends to underestimate dimension like the results of the numerical simulations.

6. Discussion

In this article, estimation by Grassmann manifold optimization and sequential candidate set algorithm in structured PFC model were compared. Trace correlation was calculated to measure the accuracy of the estimated $\hat{\Gamma}$ and results of sequential dimension testing were presented to compare the determined dimensions by the two methods.

In terms of basis estimation, we compared the trace correlation obtained from the two methods. We could conclude that in forward regression setting, sequential candidate set algorithm shows better performance with smaller variance. In inverse regression setting, sequential candidate set algorithm outperformed Grassmann manifold optimization in both performance and robustness.

From sequential dimension test results, we could conclude that Grassmann manifold optimization outperforms sequential candidate set algorithm in all settings and cases. The biggest cause of this result is the tendency of sequential candidate set algorithm to underestimate dimensions.

In addition to performance comparison of the two methods through trace correlation and sequential dimension testing, we also compared the computation time. Table 6 shows the computation time for one iteration of model 2. While Grassmann manifold optimization took 2.7711 seconds, sequential candidate set algorithm only needed 0.1026 seconds in computation. This shows that the accuracy of Grassmann optimization is at the expense of longer computation time.

Overall, among the two methods, there is no one optimal method that shows great performance in both $\hat{\Gamma}$ estimation and $\hat{d}$ determination. Therefore, sequential candidate set algorithm would be a better choice when gaining higher accuracy of estimating $\hat{\Gamma}$ is the main objective and Grassmann manifold optimization would be better when higher accuracy of $\hat{d}$ determination is the main objective.

References

Adragni KP, Cook RD, and Wu S (2012). GrassmannOptim: An R package for Grassmann Manifold Optimization, *Journal of Statistical Software*, **50**, 1–18.

Cook RD (2007). Fisher lecture: Dimension reduction in regression, *Statistical Science*, **22**, 1–26.

Cook RD (2018). *An Introduction to Envelopes: Dimension Reduction for Efficient Estimation in Multivariate Statistics*, John Wiley & Sons Hoboken, New Jersey.

Cook RD, Li B, and Chiaromonte F (2010). Envelope models for parsimonious and efficient multivariate linear regression, *Statistica Sinica*, **20**, 927–1010.

Cook RD and Forzani L (2009). Principal fitted components for dimension reduction in regression, *Statistical Science*, **485**, 485–501.

Enz R (1991). *Prices and Earnings Around the Globe*, Zurich, Switzerland: Union Bank of Switzerland.

Gallivan KA, Srivastava A, Xiuwen L, and Van Dooren P (2003). Efficient algorithms for inferences on Grassmann manifolds, In *Proceedings of IEEE Workshop on Statistical Signal Processing, 2003*, 315–318.

Yoo JK (2016). Tutorial: Methodologies for sufficient dimension reduction in regression, *Communications for Statistical Applications and Methods*, **23**, 95–117.

Yoo JK (2018). Response dimension reduction: Model-based approach, *Statistics*, **52**, 409–425.

Received August 25, 2022; Revised September 06, 2022; Accepted September 06, 2022