Some understandings and questions

2019-01-24

I am very interested in reading the papers. However, I haven't learned functional data analysis before. Therefore, I would like to wrap up my understandings about functional data analysis, clustering, and preconditioning. Besides, I tried to reproduce the simulation results in the 2007 paper. Here are also some questions I met when reading the papers. Hope this tiny document can make our meeting more efficient.

Understandings and Questions

1. Functional data analysis (FDA) and longitudinal data analysis

I feel the data structure in functional data analysis similar to longitudinal data analysis (e.g. ECG data, collecting data from different people at a different time). Then what are the differences between these two methods?

I guess the difference is that FDA treats the observed data functions as single entities, rather than the sequence of individual observations. Besides, FDA does not have parametric assumptions while LDA needs. And the data in LDA may be more sparse than FDA.

2. The procedure of clustering FDA

There are four methods mentioned in the paper:

- Raw data
- B-spline basis and Fourier basis.
- Power basis, L2 metric

Raw data

I just wanted to make sure my understanding about them is correct. For the raw data method, we just put the $y_i s$ in the k-means algorithm without any transformations. If we have n observations, each observation has m observed points, then the input data is a n * m matrix and each observation is a m-dimensional real vector. Put this data into the k-means algorithm, which then clusters them into K groups.

Question 1:

If we would like to use raw data, does that mean the data must have a very neat structure and no missing data? For example, if the ECG points were collected at different time points for different people, does the method still work?

Basis

We can also estimate the functions and put the coefficients into the k-means to reduce the dimension. That is, the function $f()$ can be estimated by the function, which consists of a linear combination of K known basis functions, such as Fourier basis, B-splines basis.

n observations

$$y_i = f(t_i) + \epsilon_i, i = 1, ..., n,$$

And

$$f(t) = \sum_{k=1}^{k=K} b_k x_k(t)$$

x_k is the basis function.

And the coeffficents can be esitmated through least square methods, similar with linear regression:

$$\hat{b} = (X^T X)^{-1} X^T y$$

where,

$$\mathbf{X} = \begin{bmatrix} x_1(t_1) & x_2(t_1) & ... & x_K(t_1) \\ x_1(t_2) & x_2(t_2) & ... & x_K(t_2) \\ ... & ... & ... & ... \\ x_1(t_n) & x_2(t_n) & ... & x_K(t_n) \end{bmatrix}$$

The basis functions can be Fourier functions, B-splines, Wavelets and so on.

Question 2: How could we choose the estimate basis?

I found a statement online. If the data is periodic, Fourier may work better since Fourier series has a period. Otherwise, B-splines is more common.

And usually how many functions ($x_i(t)s$) we need to estimate the $f()$?

Question 3: Different basis can be transformed from one to each other?

The derivative in the paper said

$$\begin{aligned} \hat{b}_B &\approx (\hat{X}'_B \hat{X}_B)^{-1} \hat{X}'_B y \\ &= (\hat{X}'_B P'_F P_F X_B)^{-1} X'_B P'_F y = (\hat{X}'_B P'_F P_F X_B)^{-1} X'_B P_F y && \text{since } P_F = P'_F \\ &= (\hat{X}'_B P_F X_B)^{-1} X'_B X_F (X'_F X_F)^{-1} X'_F y && \text{since } P_F \text{ is idempotent} \\ &= T \hat{b}_F \end{aligned}$$

Does that mean the estimated coefficients from one basis function can be transformed from another one basis function?

Why is that $\hat{b}_B \approx (\hat{X}'_B \hat{X}_B)^{-1} \hat{X}'_B y$. Why they are not equal, i.g. $\hat{b}_B = (\hat{X}'_B \hat{X}_B)^{-1} \hat{X}'_B y$?

The thing I feel very interesting is that changing the regression coefficients can affect the results a lot.

Simulation

I tried to repeat the results (figure 3 and figure 4) in the 2007 paper. However, my results are not very good. I would like to show what I did to make sure whether the procedures were correct or not. Consider K = 3 clusters and a linear function

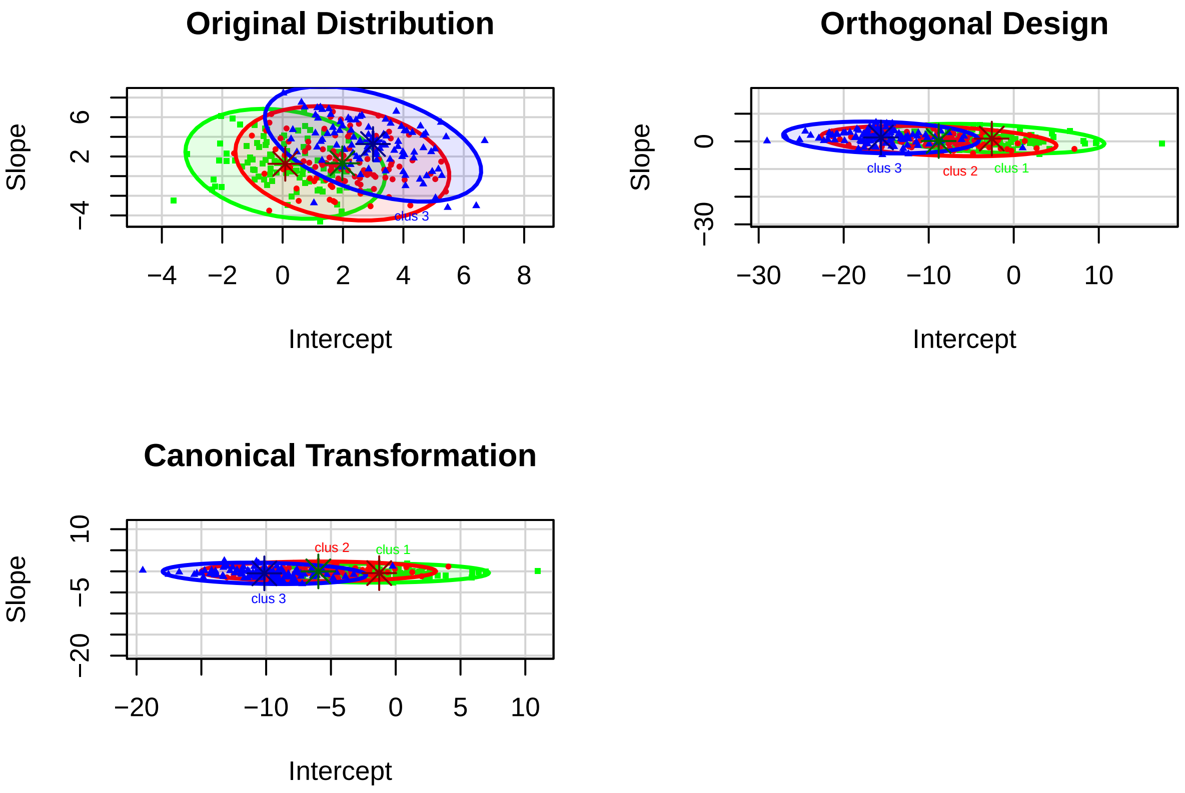
$$y(t) = b_0 + b_1 t + \epsilon,$$

Where $\epsilon \sim N(0, 0.25)$. The b_0 and b_1 are simulated from a multinormal distribution for each of the three clusters:

- cluster 1
$$\begin{pmatrix} b_0 \\ b_1 \end{pmatrix} = N(\begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 & -1 \\ -1 & 6 \end{pmatrix})$$
- cluster 2
$$\begin{pmatrix} b_0 \\ b_1 \end{pmatrix} = N(\begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 & -1 \\ -1 & 6 \end{pmatrix})$$
- cluster 3
$$\begin{pmatrix} b_0 \\ b_1 \end{pmatrix} = N(\begin{pmatrix} 3 \\ 3 \end{pmatrix}, \begin{pmatrix} 2 & -1 \\ -1 & 6 \end{pmatrix})$$

  And $\pi_1 = \pi_2 = \pi_3 = 1/3$. $t = 0.1, 0.2, 0.3, .., 1.0$. For each cluster, 100 $b_0 s, b_1 s$ were simulated from the above multivariate normal distributions.

The Figure 3



The results do not look similar to the one in the paper.

Coefficient distribution based on the original distribution

With the original data, I just put the simulated b_0 and b_1 in the plots and drew their two-dimensional ellipse based on mean and covariance.

Coefficient distribution using an orthogonal design matrix

To transform with an orthogonal design matrix, SVD decomposition was performed on X. The X matrix is

$$X = \begin{pmatrix} 1 & 0.1 \\ 1 & 0.2 \\ ... & ... \\ 1 & 1.0 \end{pmatrix}$$

$X = UDV'$. And $X_0 = XVD^{-1}$ is an orthogonal design matrix and $b_0 = DVb$ is the vector of associated regression coefficients.

Therefore, to draw plots of intercepts and slopes, I times b matrix with DV, which is just b_0 above.

Coefficient distribution using a canonical transformation

The covariance matrix for b can be decoppsed as within cluster variance W and between cluster variance B, that is

$$cov(b) = W + B,$$

where

$$W = \sum_{j=1}^{k} \pi_j \Phi_j \text{ , and } B = \sum_{j=1}^{k} \pi_j (\mu_j - \mu)(\mu_j - \mu)'$$

And

$$W^{-1/2} B W^{-1/2} = HDH'$$

where H is a matrix where the columns are eigenvectors of $W^{-1/2}BW^{-1/2}$, D is the diagonal matrix where the diagonal values are eigenvalues of $W^{-1/2}BW^{-1/2}$ from biggest to smallest.

And then we know that the canonical transformation for cluster is

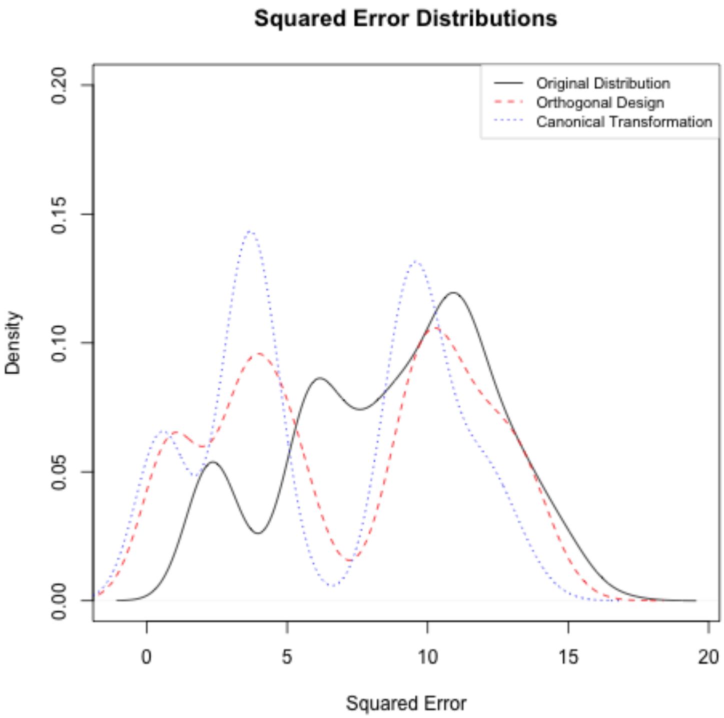
$$C\Gamma' b,$$

C is a diagnoal matrix and is chosen as (3.5,1) and $\Gamma = W^{-1/2}H$

Therefore, to draw plots of intercepts and slopes, I just times b matrix with $C\Gamma'$.

However, the final results do not look good. Is there some misunderstanding in those processes?

The Figure 4



I put the matrix of $[b_o, b_1]_{(300,2)}$ (3 clusters, each cluster has 100 observations, each observation has a simulated intercept and slop) into the k-means algorithm, as well as the matrix with above transformations. After calculations of the center, the sum of the squared error between the estimated center and the true center is calculated. The b matrix from transformation methods is then transformed back to the original version (for example, for the orthogonal design matrix transformation, the results time with $(DV)^{-1}$; for the canconical transformation, the results time with $(C\Gamma')^{-1}$). After 1000 replications of the above process, the mean squared error for each method is calculated and their densities are plotted.

However, my results have multiple peaks, which are different from the results in the paper.

Question 4:

However, in practice, we do not know the true mean value in each cluster. How could we estimate the effect of different methods? Using projection pursuit clustering?

Question 5:

The simulation process is different from the one in practice, right? In practice, we need to fit the function with basis models and estimate the coefficients, and then put the coefficients in the k-means algorithm, which is then similar to above, is that right?