RKHS regression: deeper understanding of its basis and penalty via eigenanalysis

Sunwoo Lim (In class project for "Functional Data Analysis", STA6320)

May 26, 2023

1. Introduction

Nonparametric regression task given by $Y_i = f(x_i)+\epsilon_i, \epsilon_i \overset{\text{iid}}{\sim} N(0,\sigma^2), i = 1, ..., n$ is used often in functional data analysis (FDA) mainly for 1) converting discontinuous data into a functional data and 2) mean estimation in sparse functional setting. Mainly three classes of models exist for estimating f: 1) basis selection (either splines or functional principal component analysis), 2) a fixed basis penalization (e.g, B-spline basis functions and roughness penalty added, or Fourier basis and harmonic acceleration penalty), and 3) reproducing kernel Hilbert space (RKHS) regression, also called as kernel ridge regression (KRR).

I focus on the last two models, common in that they are linear smoothers and the estimated curve lies in a finite dimensional vector space. However, they have distinctions mainly in two ways. First, while the dimension of the basis is $p < n$ for fixed basis penalization models, dimension of the basis for the KRR is n, which literally is a high-dimensional statistical situation. More importantly, while users predetermine the basis functions and the penalty functionals in the fixed basis penalization method, users predetermine the kernel function, which induces the RKHS, basis functions and the penalty functional.

A preliminary analysis to understand the basis functions and the penalty functional can be conducted by plotting different fitted curves $\hat{f}$ by different choices of bases, smoothing parameter λ and penalty functionals. However, since the fitted curve also relies on random noise, the analysis is not easy to interpret, which motivated me to more thoroughly analyze the basis functions and the penalty functionals; the underlying process that generates certain fitted curves both in two penalization models. Two main tools I use for certain understanding are 1) eigenanalysis of the smoother matrices and 2) plots of the fitted curves as a secondary tool that makes the eigenanalysis more convincing. I referred to Hastie et al. (2009) for the eigenanalysis and I further applied this framework to analyze the Kernel ridge regression.

2. Methodology

2.1 Review of fixed basis penalization

In first two subsections, I briefly review the two penalization methods for the nonparametric regression task. Let $\mathcal{S} = span\{\boldsymbol{B}_1, ..., \boldsymbol{B}_p\}$, a p-dimensional vector space spanned by p predetermined basis functions. Then, the task is to solve

$$\min_{f\in\mathcal{S}} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda J(f), \tag{1}$$

where $J(f)$ is a penalty functional. Through this project, I assume a p dimensional cubic B-spline basis(: $p-4$ interior knots) and a second order finite difference penalty on coefficients, which is similar as $\int f''^2$. This problem is called P-splines (Eilers and Marx, 1996). Letting $\boldsymbol{B} \in \mathbb{R}^{n\times p}$, a basis matrix and $\boldsymbol{P} \in \mathbb{R}^{p\times p}$, a penalty matrix, I obtain

$$\begin{aligned} &\min_{\boldsymbol{\theta}\in\mathbb{R}^p} (\boldsymbol{y} - \boldsymbol{B\theta})^T(\boldsymbol{y} - \boldsymbol{B\theta}) + \lambda\boldsymbol{\theta}^T\boldsymbol{P\theta} \\ &\leftrightarrow \hat{\boldsymbol{\theta}} = (\boldsymbol{B}^T\boldsymbol{B} + \lambda\boldsymbol{P})^{-1}\boldsymbol{B}^T\boldsymbol{y} \\ \leftrightarrow \hat{f}(x) = &\sum_{j=1}^{p} \hat{\theta}_j B_j(x), \text{ which is a linear smoother.} \end{aligned} \tag{2}$$

2.2 Kernel ridge regression

Let $\mathcal{X}$ be the domain of the function. Let $\mathcal{H}$ be the RKHS induced by kernel $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. KRR solves

$$\min_{f\in\mathcal{H}} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda||f||^2_{\mathcal{H}}. \tag{3}$$

By the Rietz's representator theorem, the problem tranforms into a finite dimensional solution

$$\begin{aligned}\min_{\boldsymbol{\alpha}\in\mathbb{R}^n}\sum_{i=1}^{n}\left(y_i-\sum_{j=1}^{n}\alpha_j K(x_i,x_j)\right)^2+\lambda\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j<K(\cdot,x_i),K(\cdot,x_j)>_{\mathcal{H}}\\ \leftrightarrow \min_{\boldsymbol{\alpha}\in\mathbb{R}^n}||\boldsymbol{y}-\boldsymbol{K\alpha}||_2^2+\lambda\boldsymbol{\alpha}^T\boldsymbol{K\alpha}\\ \leftrightarrow \hat{\boldsymbol{\alpha}}=(\boldsymbol{K}^T\boldsymbol{K}+\lambda\boldsymbol{I})^{-1}\boldsymbol{K}^T\boldsymbol{y}\\ \leftrightarrow \hat{f}(x)=\sum_{j=1}^{n}\hat{\alpha}_j K(x,x_j),\ \text{which also is a linear smoother.}\end{aligned} \tag{4}$$

2.3 Eigenanalysis of the smoother matrix

Although the framework and the dimension of fixed basis penalization method and KRR are distinct, the eigenanalysis framework can be applied to both methods.

$$\begin{aligned}\hat{f}=\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{B}+\lambda\boldsymbol{P})^{-1}\boldsymbol{B}^T\boldsymbol{y}=\boldsymbol{S}_\lambda\boldsymbol{y}=\boldsymbol{S}\boldsymbol{y}\text{: for smoothing splines}\\ \hat{f}=\boldsymbol{K}(\boldsymbol{K}^T\boldsymbol{K}+\lambda\boldsymbol{I})^{-1}\boldsymbol{K}^T\boldsymbol{y}=\boldsymbol{S}_\lambda\boldsymbol{y}\text{ : for KRR.}\end{aligned} \tag{5}$$

$\boldsymbol{S}_\lambda$ in (5) is called a 'smoother matrix'. I list important facts about smoother matrices which is crucial in understanding the basis functions and the penalty of each method.

1. $\boldsymbol{S}_\lambda^2 \preceq \boldsymbol{S}_\lambda$, which means that $\boldsymbol{S}_\lambda$ is $\boldsymbol{S}_\lambda^2$ plus a positive semidefinite matrix. Since $\boldsymbol{S}_\lambda$ is positive semidefinite, this implies that eigenvalues of $\boldsymbol{S}_\lambda$ are between 1 and 0 (including both). Eigencomponents with eigenvalues 1 are not penalized and the rest are penalized.
2. Eigendecomposition of $\boldsymbol{S}_\lambda$ leads to $\boldsymbol{S}_\lambda=\sum_{i=1}^{n}\rho_i\boldsymbol{v}_i\boldsymbol{v}_i^T$. This leads to $\boldsymbol{S}_\lambda\boldsymbol{y}=\sum_{i=1}^{n}\rho_i<\boldsymbol{v}_i,\boldsymbol{y}>\boldsymbol{v_i}$. $<\boldsymbol{v}_i,\boldsymbol{y}>$ is a similarity measure between the noisy data $\boldsymbol{y}$ and the eigenfunction $\boldsymbol{v}_i$ and ρ_i is the corresponding eigenvalue.
3. Arranging the eigencomponents by decreasing eigenvalues, when $i<j$, $\boldsymbol{v}_j$ tends to be more complex than $\boldsymbol{v}_i$.
4. For the same index i of the eigencomponent, increasing λ, the eigenvectors remain the same, with smaller eigenvalues.

Thus, the key is to analyze $\rho_i\boldsymbol{v}_i$, which both conveys messages of which eigencompoennt $\boldsymbol{v}_i$ is used as a compoent of $\boldsymbol{S_\lambda y}$ and how much shrinkage effect ρ_i is imposed on it.

3. Simulation analysis

3.1. Simulation settings

I conducted a simulation analysis with three different regression functions illustrated in Figure 1. I set $n=800$ with $\sigma=0.4$ for f_1 and f_2, $\sigma=0.1$ for f_3. f_1 is a sinusoid, f_2 is a linear combination of Gaussian densities and f_3 is a nonlinear combination of sinusoids, which is not smooth.

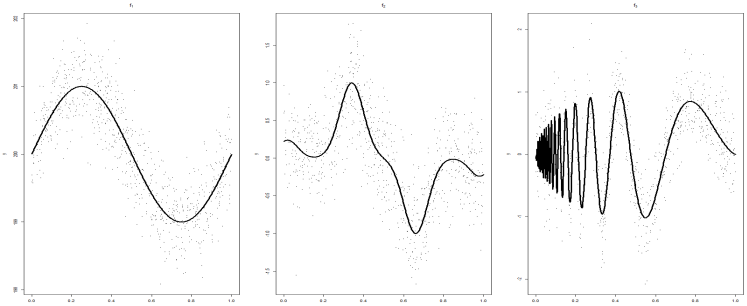

Figure 1: Regression functions (line) and training data (dots) f_1 : left, f_2 : middle, f_3 : right

3.2. Models to consider

In this section, we consider four different models, each with two different hyperparameter settings. First, I consider cubic P-splines with dimension 24 and 400 respectively. For kernel methods, I consider Laplacian kernel $K(x,y;\gamma)=exp(-\gamma||x-y||_1)$ with $\gamma=0.1$ and 1, Gaussian kernel $K(x,y;\sigma)=exp(-0.5||x-y||_2^2/\sigma^2)$ with $\sigma=0.1$ and 0.05 and finally, Periodic kernel $K(x,y;k)=exp(sin(k\pi|x-y|)^2))$ with $k=1$ and 0.5.

Figure 2 have contour plots of different kernel matrices. The shape of $\boldsymbol{K}$ in Laplacian and Gaussian kernels are similar but have differences. Laplacian kernels tend to be dense, while Gaussian kernels tend to be sparse (in this case, banded). This is because in univariate setting ($\mathcal{X} \subset \mathbb{R}$), $||x-y||_2^2 = (x-y)^2$ gets really small when x and y are near compared to $||x-y||_1 = |x-y|$.

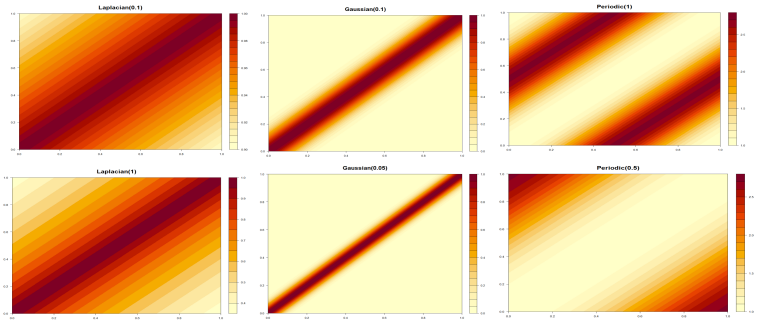

Figure 2: Contour plot of kernel matrices $\boldsymbol{K} \in \mathbb{R}^{n \times n}$

In each kernel and a hyperparameter, I chose different λ values illustrated in later sections.

3.3. P-splines

In this section, I illustrate the result using P-splines. Color 'red' is the case when $\lambda = 0$ and color 'purple' is when λ is the biggest. Figures 3 and 4 represent the out of sample fitted curve on the left hand side and the eigenanalysis on the right hand side respectively. Inspecting on Figure 3, when $\lambda = 0$, first 24 eigenvalues are 1, while when $\lambda > 0$, eigenvalues monotonically decrease. Eigenvectors of smoother matrices $\boldsymbol{S}_\lambda, \lambda > 0$ are the same for all λ, but they are totally different from those of the projection matrix $\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{B})^{-1}\boldsymbol{B}^T$. In P-splines, first two eigenvalues are 1 for all λ and eigenvectors are at most linear. This is a natural result because $\int f''^2 = 0$ when f is at most a linear function on $\mathcal{X}$. Eigenvectors look like polynomials in increasing degree.

In Figure 4, with more basis functions, the eigenvalues are bigger for same λ (represented in same color). This means $\boldsymbol{S}_\lambda \boldsymbol{y}$ requires a more complex representation, utilizing more complex eigencomponent, resulting in a wigglier fit.

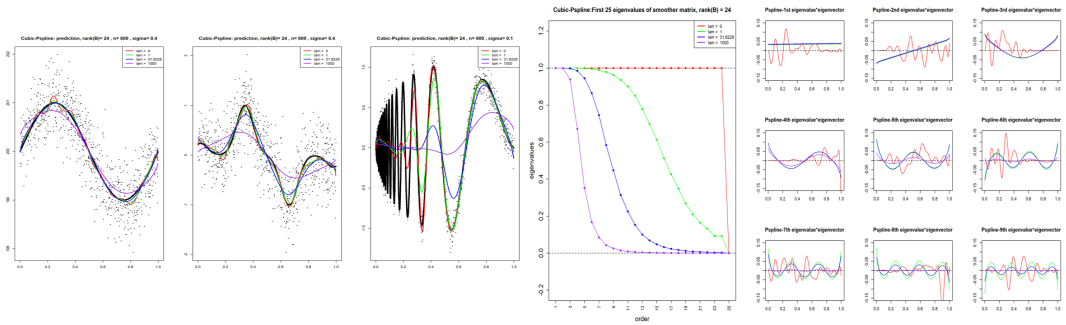

Figure 3: P-splines with dimension = 24 (20 interior knots), left: test fitted, right: eigenanalysis

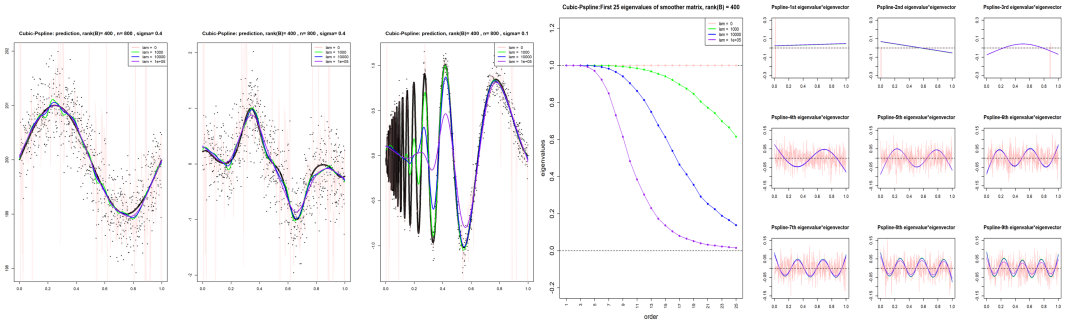

Figure 4: P-splines with dimension = 400 (396 interior knots), left: test fitted, right: eigenanalysis

3.4. Kernel methods

In this section, we deal with three different kernels, each with two different hyperparameters. Figures 5 and 6 represent the analyses of Laplacian kernel. Conducting a similar analysis as in P-splines, the first eigenfunction (constant) is not penalized

($\rho_1 = 1$). Increasing the index, the eigenvalues monotonically decrease, and eigenvectors show a sinusoidal pattern with increasing frequency (complexity). When $\gamma = 1$, eigenvalues decrease slower than when $\gamma = 0.1$ for same λ, making a wigglier out of sample prediction, because the shape of the eigenfunctions are similar for $\gamma = 1$ and $\gamma = 0.1$.

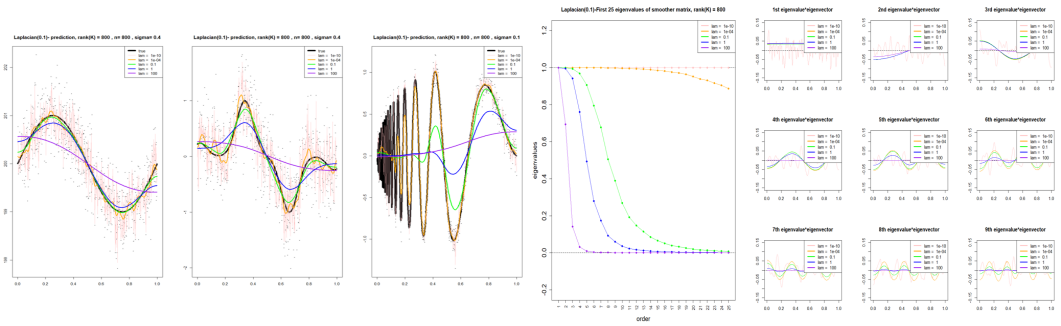

Figure 5: Laplacian(0.1), left: test fitted, right: eigenanalysis

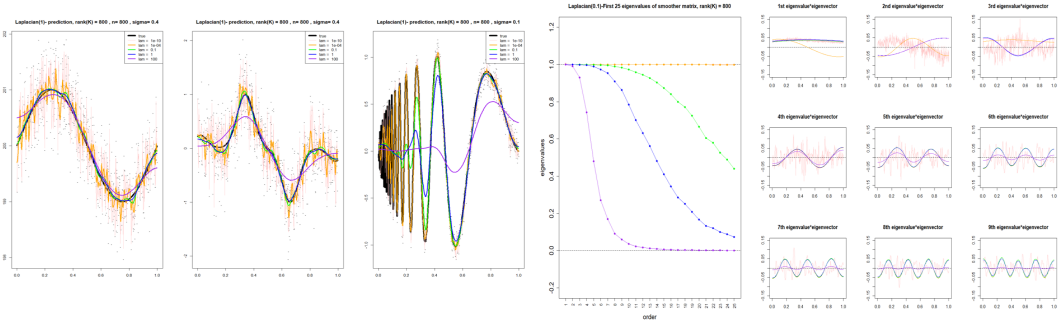

Figure 6: Laplacian(1), left: test fitted, right: eigenanalysis

Figures 7 and 8 are the analyses of Gaussian kernel. Eigenvalues of $\sigma = 0.05$ decrease slower than the case of $\sigma = 0.1$. Shapes of eigenfunctions look like polynomials with distinct local extreme values.

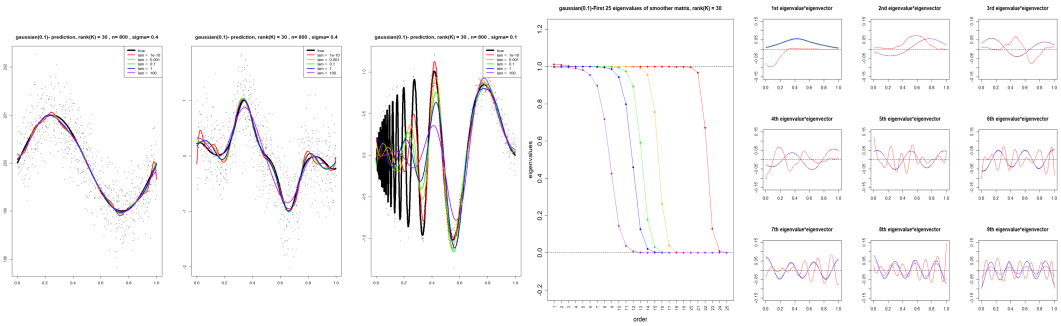

Figure 7: Gaussian(0.1), left: test fitted, right: eigenanalysis

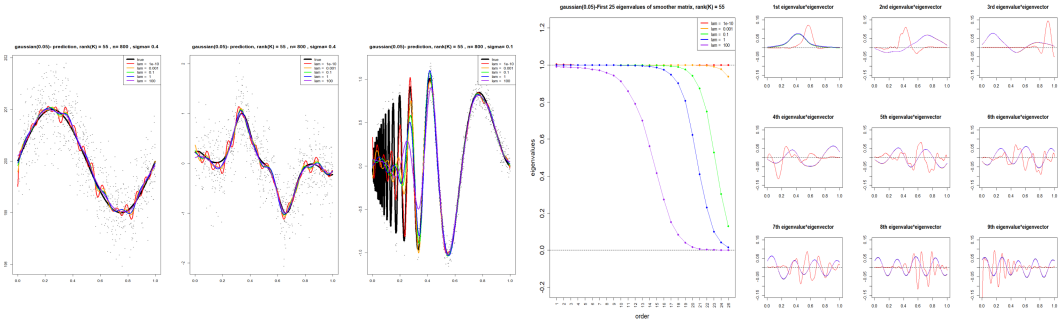

Figure 8: Gaussian(0.05), left: test fitted, right: eigenanalysis

Figures 9 and 10 are the analyses of Gaussian kernel. Eigenvalues of $k = 1$ decrease slower than the case of $k = 0.5$, but

overall, eigenvalues rapidly shrink to 0. Also, the shape of eigenvectors look like sinusoids, meaning that linear combination of small number of sinusoids are used to represent $\boldsymbol{S}_\lambda \boldsymbol{y}$.

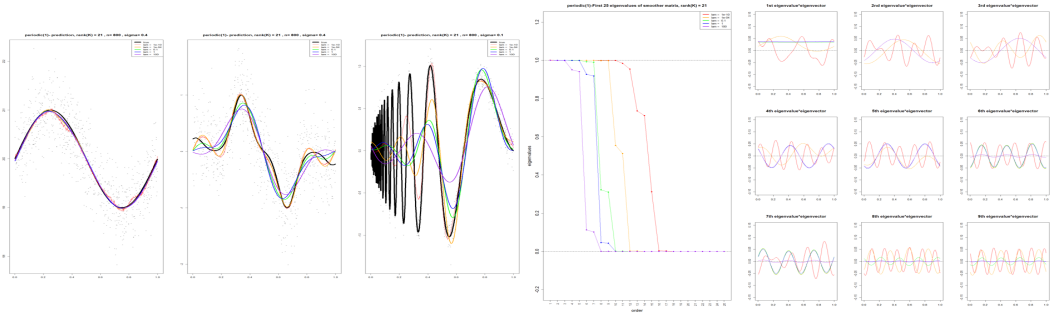

Figure 9: Periodic(1), left: test fitted, right: eigenanalysis

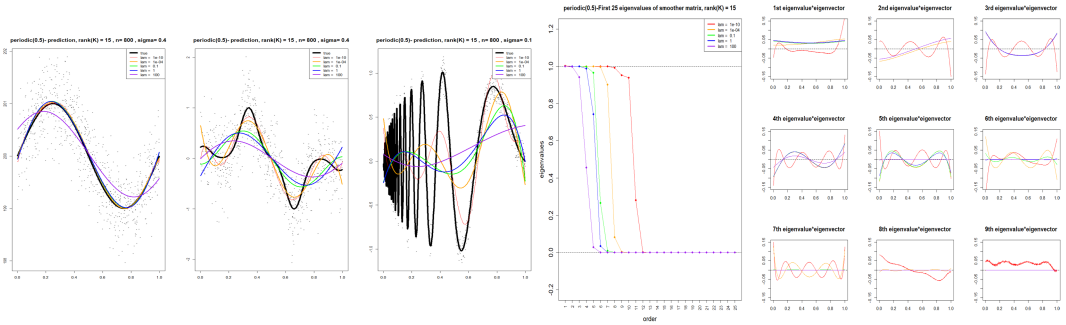

Figure 10: Periodic(0.5), left: test fitted, right: eigenanalysis

3.5. Comparison of each model by fitted curves

In this section, I form bivariate grid of hyperparameters of the kernels (or knots for P-splines) and λ. I set the grid of $\lambda = 10^{-4}, 10^{-3.5}, ..., 10^{3.5}, 10^4$ for all methods. For P-splines, I set the grid of number of interior knots as 0,1,...,100. For kernel methods, I set the grid of hyperparameters $10^{-4}, 10^{-3.5}, ..., 10^{3.5}, 10^4$. On these bivariate grids, I evaluate generalized cross validation (GCV) refering to Hastie et al. (2009):

$$n \times GCV = \sum_{i=1}^{n} \left(\frac{y_i - \hat{f}(x_i)}{1 - tr(\boldsymbol{S}_\lambda)/n} \right)^2 \tag{6}$$

I did not divide n because only the order of GCV's are important, not the actual values. Then, I optimized the hyperparameter and λ for each model and plotted Figures 11 to 13, the test fitted curves of f_1, f_2, f_3, respectively by the (frequentist) plug-in method.

Observing Figure 11, Gaussian kernels and periodic kernels have great out-of sample performance, P-splines show moderately good performance, and Laplace kernel shows somewhat unsatisfactory performance, expecially at the boundaries. This shows the overfitting behavior of Laplace kernel due to a dense nature of kernel matrix. But overall, distinctions are not too big.

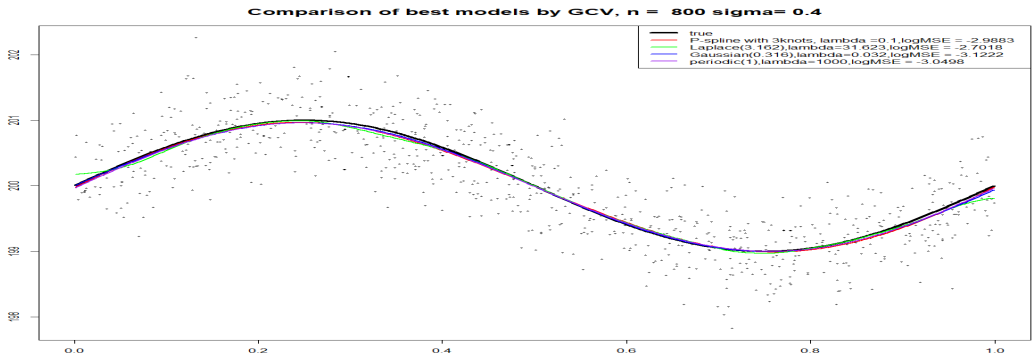

Figure 11: f_1: Comparison of models where hyperparameters and λ are chosen by GCV

In Figure 12, every model except from periodic kernel performs quite well. The fit of periodic kernel (purple) looks like a linear combination of sinuosids. This is why it shows opposite directions near boundaries of $\mathcal{X}$.

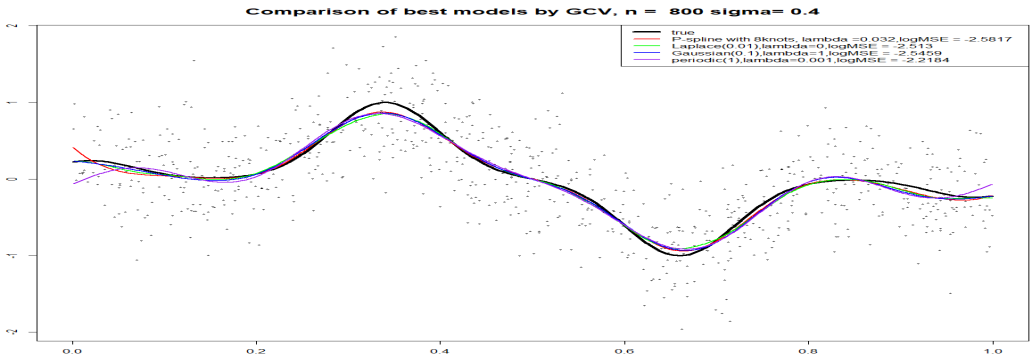

Figure 12: f_2: Comparison of models where hyperparameters and λ are chosen by GCV

Figure 13 looks interesting. Laplace kernels behave the best in this wiggly regression function. Gaussian and P-splines show moderately good behavior and the periodic kernel behaves the worst. The estimated curve using periodic kernels look like linear combination of small number of sinusoids. However, more than enough number of sinusoids may be necessary for this example, making the sinusoid kernel behave terrible. Also, a wiggly behavior of Laplace kernel worked really well in this case, compared to Gaussian kernels, which generally result in less estimated curves.

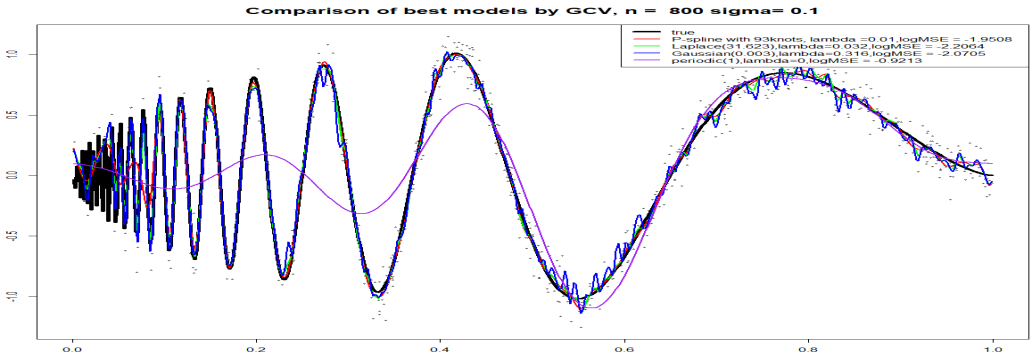

Figure 13: f_3: Comparison of models where hyperparameters and λ are chosen by GCV

4. Conclusion

I applied the eigenanalysis framework to explicitly understand the basis functions and the penalty functional induced by different kernels. When $\lambda > 0$, eigenvectors are identical with varying eigenvalues. Also, aligning the eigencomponents by decreasing order of eigenvalues, eigenvectors increase in complexity. Laplacian kernels normally result in more wiggly fit than Gaussian kernels, despite similar shape of the kernel matrices. This is due to the difference of $||x-y||_1$ and $||x-y||_2^2$. Periodic kernels result in a sinusoidal fit. The problem is, I do not know the information of the regression function beforehand. Thus, this project may not worth as a guideline. But, this project is for understanding the behavior of RKHS regression, where $\mathcal{H}$ is induced by different kernels. However, to provide a minimal guideline, since fixed basis penalization methods are more computationally efficient, numerically stable and can well adapt to various kinds of regression functions, in practical perspective, I recommend using fixed basis penalization methods, such as P-splines. Still, in this case, the number of knots is an important hyperparameter.

5. Code and Data Availability Statement

The R codes are uploaded in https://github.com/damelim/FDAproject.

References

Eilers, P. H. and Marx, B. D. (1996). Flexible smoothing with b-splines and penalties. *Statistical science*, 11(2):89–121.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.