

# Scalar on Function Regression with Applications to Near-Infrared Spectroscopy

Jonathan Willnow, Jakob Juergens, Jonghun Baek

18.01.2022

Research Module in Econometrics and Statistics  
Winter Semester 2021/2022

# Introduction

**Near-Infrared (NIR) Spectroscopy** enables fast diagnostics by using the NIR region of the electromagnetic spectrum

- ▶ Spectroscopy results in high-dimensional dataset: Gasoline dataset ( $60 \times 401$ )
- ▶ This set of measurements serves as set of discretized approximations of smooth spectral curves

$$\{x_i(t_{j,i}) \in \mathbb{R} \mid i = 1, 2, \dots, N, j = 1, 2, \dots, J_i, t_{j,i} \in [T_1, T_2]\}$$

- ▶ Continuous underlying process where  $x_i(t)$  exists  $\forall t \in [T_1, T_2]$  but is only observed at  $t_{j,i}$
- ▶ Regression to determine relationship between octane rating and spectral curves

# Random Function

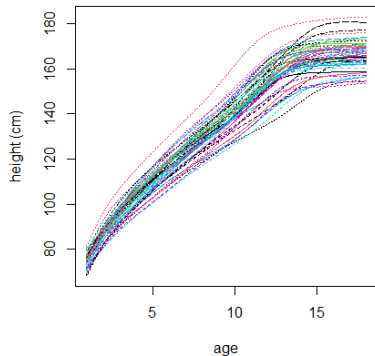
A **Random Variable** is a function  $X : \Omega \rightarrow \mathcal{S}$  which is defined on a common probability space  $(\Omega, \mathcal{F}, \mathbb{P})$  where  $\Omega$  is a probability space with a  $\sigma$ -algebra  $\mathcal{F}$  and a probability measure  $\mathbb{P}$ .

- ▶ If  $\mathcal{S} = \mathbb{R}$  then  $X$  is a random variable
- ▶ If  $\mathcal{S} = \mathbb{R}^n$  then  $X$  is a random vector
- ▶ If  $\mathcal{S}$  is a space of functions,  $X$  is called a **Random Function**

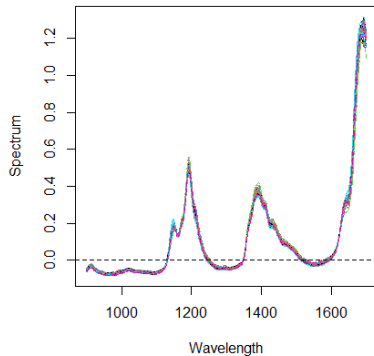
A **Realization** of a random function  $X(\omega)$  is a function

$$x_0(t) = X(\omega_0)(t) \quad t \in \mathbb{E}, \omega_0 \in \Omega$$

# Plots



Growth curves of  
54 girls age 1-18



NIR spectrum of 60  
gasoline samples

# Square Integrable Function

A function is called **Square Integrable** written  $f(t) \in \mathbb{L}^2[0, 1]$  if

$$\int_0^1 (f(t))^2 dt < \infty$$

- ▶ Without loss of generality, the interval is defined in  $[0, 1]$ .

Let  $f, g \in \mathbb{L}^2[0, 1]$ , then we can define inner product by

$$\langle f, g \rangle = \int_0^1 f(t)g(t)dt$$

- ▶ Orthogonality of two different functions with  $\langle f, g \rangle = 0$

# Square Integrable Function

A natural **Norm** to be defined on  $\mathbb{L}^2[0, 1]$  is the following norm induced by the inner product.

$$\|f\| = \sqrt{\langle f, f \rangle} = \sqrt{\int_0^1 f(t)^2 dt}$$

This norm naturally induces a **Distance** on  $\mathbb{L}^2[0, 1]$ .

$$d(f, g) = \|f - g\| = \sqrt{\langle f - g, f - g \rangle} = \sqrt{\int_0^1 [f(t) - g(t)]^2 dt}$$

# Basis Expansion

**Basis Expansion** is a linear combination of functions as described:

$$x_i(t) = \sum_{j \in \mathcal{I}} c_{i,j} \phi_j(t) \approx \sum_{j=1}^L c_{i,j} \phi_j(t), \quad i = 1, \dots, n, \quad \forall t \in \mathbb{E}$$

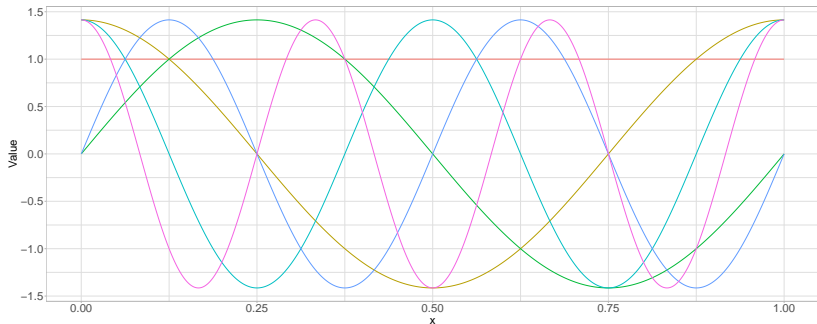
where  $\phi_j(t)$  is the  $j^{\text{th}}$  basis function of the expansion and  $c_{i,j}$  is the corresponding coefficient. We truncate the basis at  $L$  to:

- ▶ make the function smoother
- ▶ replace the original curves  $x_i(t)$  with a smaller collection of  $c_{n,m}$

# Basis Functions

**Fourier Basis Functions** are elements of the set:

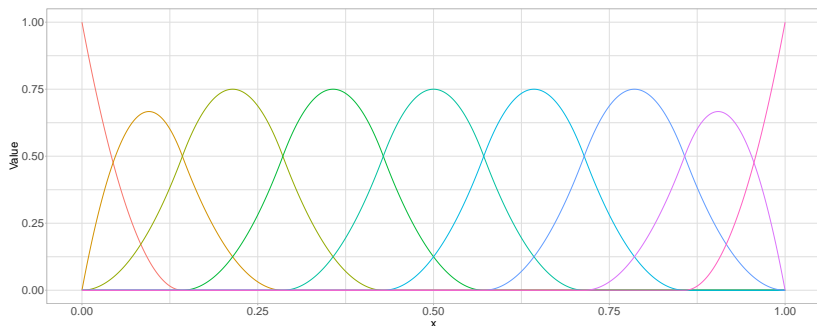
$$\{\sqrt{2} \sin(2\pi nx) | n \in \mathbb{N}\} \cup \{\sqrt{2} \cos(2\pi nx) | n \in \mathbb{N}\} \cup \{1\}$$





# Basis Functions

**B-spline Basis Functions** are piece-wise polynomial functions defined by an order and a set of knots.



# Trade Off between Bias and Variance

How do we choose the number  $L$  of basis functions?

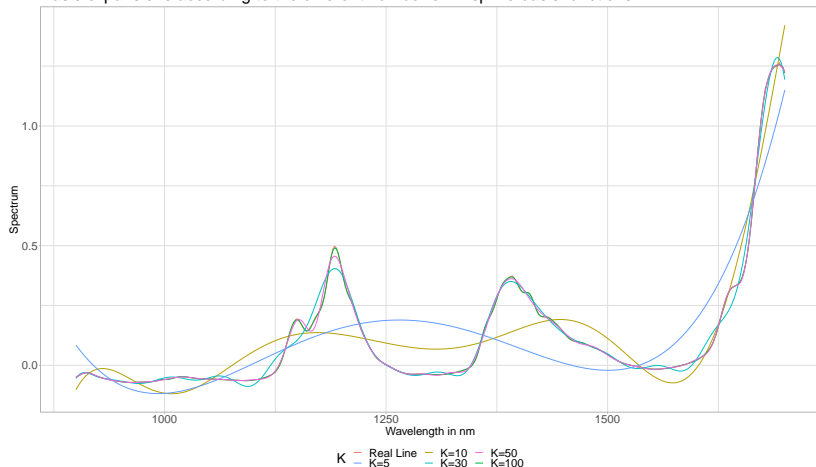
$$\mathbf{MSE}[\hat{X}(t)] = \mathbf{Bias}^2[\hat{X}(t)] + \mathbf{Var}[\hat{X}(t)]$$

$$\mathbf{IMSE}[\hat{X}] = \int_0^1 \mathbf{MSE}[\hat{X}(t)] dt$$

- ▶ The larger  $L$ , the better the fit to the data, but also more fitting noise
- ▶ If  $L$  is too small, the expansion would miss some significant information

# Trade Off between Bias and Variance

Basis expansions according to the different number of B-spline basis functions



# Estimation via Basis Representation

Assume the following **Data Generating Process**

$$Y(\omega) = \alpha + \int_0^1 \beta(s)X(\omega)(s)ds + \epsilon(\omega)$$

- $Y(\omega)$  and  $\epsilon(\omega)$  realize in  $\mathbb{R}$  and  $X(\omega)$  realizes in  $\mathbb{L}^2[0, 1]$

Let  $\{\phi_i(t) \mid i \in \mathcal{I}\}$  be a basis leading to the following representation

$$\beta(t) = \sum_{j \in \mathcal{I}} c_j \phi_j(t) \approx \sum_{j=1}^L c_j \phi_j(t)$$

# Estimation via Basis Representation

We can transform the data generating process into:

$$\begin{aligned} Y(\omega) &= \alpha + \int_0^1 \left[ \left( \sum_{j \in \mathcal{I}} c_j \phi_j(s) \right) X(\omega)(s) \right] ds + \epsilon(\omega) \\ &= \alpha + \sum_{j \in \mathcal{I}} \left[ c_j \int_0^1 X(\omega)(s) \phi_j(s) ds \right] + \epsilon(\omega) \\ &= \alpha + \sum_{j \in \mathcal{I}} c_j Z_j(\omega) + \epsilon(\omega) \approx \alpha + \sum_{j=1}^L c_j Z_j(\omega) + \epsilon(\omega) \end{aligned}$$

Where a  $Z_j(\omega)$  is a **scalar random variable**.

► Equation with Approximated Functions

# Estimation via Basis Representation

Truncating the functional basis allows us to estimate coefficients using **Multivariate Regression** leading to an estimated coefficient vector  $\hat{\mathbf{c}} \in \mathbb{R}^L$  and an estimated coefficient function  $\hat{\beta}_L(t)$ :

$$\hat{\beta}_L(t) = \sum_{j=1}^L \hat{c}_{L,j} \phi_j(t)$$

This is dependent on...

- ▶ The basis  $(\phi_j(t))_{j \in \mathcal{I}}$  for the estimation of  $\beta(t)$
- ▶ The basis  $(\psi_j(t))_{j \in \mathcal{L}}$  used for the observations
- ▶ The truncation parameter  $L$
- ▶ The truncation parameter for the observations  $K$

# Karhunen-Loève Expansion

**Mean Function:**

$$\mu(t) = \mathbb{E} [X(\omega)(t)]$$

**Autocovariance Function:**

$$c(t, s) = \mathbb{E} [ (X(\omega)(t) - \mu(t)) (X(\omega)(s) - \mu(s)) ]$$

The **Eigenvalues** and **Eigenfunctions**:  $\{(\lambda_i, \nu_i) \mid i \in \mathcal{I}\}$  are solutions of the following equation:

$$\int_0^1 c(t, s) \nu(s) ds = \lambda \nu(t)$$

## Karhunen-Loève Expansion

A random function  $X(\omega)$  can be expressed in terms of its **Mean Function** and its **Eigenfunctions**:

$$X(\omega)(t) = \mu(t) + \sum_{j=1}^{\infty} \xi_j(\omega) \nu_j(t)$$

Where the  $\xi_j$  are **Scalar-Valued Random Variables** with the following properties.

1.  $\mathbb{E}[\xi_i(\omega)] = 0$
2.  $\text{Var}(\xi_i(\omega)) = \lambda_i$
3.  $\text{Cov}(\xi_i(\omega), \xi_j(\omega)) = 0$  for  $i \neq j$

This is called the **Karhunen-Loève Expansion** of  $X(\omega)$  and the Eigenfunctions can serve as a basis.



# Functional Principal Component Analysis

**Principal Component Analysis** can be extended to functional regressors in the form of **Functional Principal Component Analysis** (FPCA).

**Empirical Mean Function:**

$$\hat{\mu}(t) = \frac{1}{n} \sum_{j=1}^n x_j(t)$$

**Empirical Autocovariance Function:**

$$\hat{c}(t, s) = \frac{1}{n} \sum_{j=1}^n (x_j(t) - \hat{\mu}(t)) (x_j(s) - \hat{\mu}(s))$$

# Functional Principal Component Analysis

The **Eigenvalues** and **Eigenfunctions**:  $\{(\hat{\lambda}_i, \hat{\nu}_i) \mid i \in \mathcal{I}\}$  are solutions of the following equation:

$$\int_0^1 \hat{c}(t, s) \hat{\nu}(s) ds = \hat{\lambda} \hat{\nu}(t)$$

The  $\{\hat{\nu}_i(s) \mid i \in \mathcal{I}\}$  are called **Functional Principal Components** and can serve as a basis for representing the original curves.

The corresponding scores  $\hat{\xi}_i$  can be derived as

$$\hat{\xi}_j(\omega) = \int_0^1 (F(\omega)(s) - \hat{\mu}(s)) \hat{\nu}_j(s) ds$$

# Simulation Setup

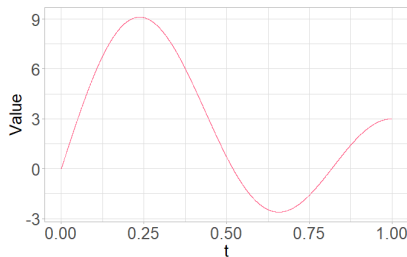
Use the **Gasoline Dataset** (NIR-spectroscopy,  $60 \times 401$ ) to generate **Similar Curves**:

$$\tilde{X}(\omega)(t) = \hat{\mu}(t) + \sum_{j=1}^J \tilde{\xi}_j(\omega) \hat{\nu}_j(t)$$

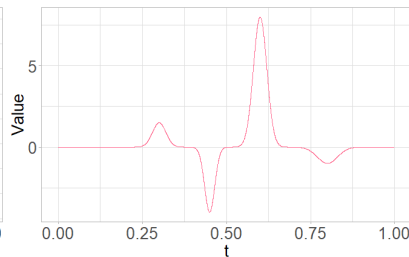
- ▶  $\tilde{\xi}_j \sim \mathcal{N}(0, \hat{\lambda}_j)$  and  $\tilde{\xi}_j \perp \tilde{\xi}_k$  for  $j \neq k$
- ▶ Simplification: the  $\tilde{\xi}_j$  do not follow a normal distribution
- ▶  $\tilde{X}(\omega)(t)$ ,  $\hat{\mu}(t)$  and  $\hat{\nu}_j(t)$  are approximated as vectors in  $\mathbb{R}^{401}$

## Simulation Setup cont.

Following **Reiss and Ogden (2007)**, let  $f_1(t)$  and  $f_2(t)$  be two coefficient functions:



$f_1(t)$ , smooth function



$f_2(t)$ , bumpy function

## Simulation Setup cont.

$$Y_{1,f} = \langle NIR, f \rangle + Z \left( \frac{\text{var}(\langle NIR, f \rangle)}{0.9} - \text{var}(\langle NIR, f \rangle) \right)$$

$$Y_{2,f} = \langle NIR, f \rangle + Z \left( \frac{\text{var}(\langle NIR, f \rangle)}{0.6} - \text{var}(\langle NIR, f \rangle) \right)$$

Let these be two responses for  $f \in \{f_1(t), f_2(t)\}$  with  $Z \sim \mathcal{N}(0, 1)$ .

- ▶ Four combinations with different number of cubic bspline basis-function  $n_{basis} \in \{4, 5, \dots, 25\}$  and fourier functions  $\{1, 3, \dots, 25\}$  to perform regression using basis expansion and the FPCR approach.
- ▶ Compare results via criteria (CV, Mallows CP, ...)

# Simulation - Interpretation of Results

## Basis Expansion Regression

- ▶ Smooth true coefficient functions requires smaller number of  $n_{basis}$  and vice versa
- ▶ Setup with higher noise requires smoother function.
- ▶ B spline setup performs overall a bit better.

## FPCR

- ▶ Two PC enough to explain variation.
- ▶ Results quite similar, but bspline setup better for smooth function with noisy response.

Overall, fourier setup has lower complexity.

▶ Basis Expansion Results

▶ FPCR Results

# Application Setup

- ▶ Use insights from the simulation study to uncover dependence.
- ▶ Similar setup, but using only bspline basis expansion and initial 60 spectral curves.
- ▶ Validation set approach: Scores of test data needs to be estimated by the training data.
- ▶ Report results by MSE scaled by variance.

# Further Reading



Gy. Bohács, Z. Ovádi, A. Salgó (1998). "Prediction of Gasoline Properties with near Infrared Spectroscopy". In: *Journal of near infrared spectroscopy*. 6, pp. 341–348.



Hsing, Tailen and Randall Eubank (Mar. 2015). *Theoretical Foundations of Functional Data Analysis, with an Introduction to Linear Operators*. en. Google-Books-ID: om9uBwAAQBAJ. John Wiley & Sons. ISBN: 978-1-118-76256-1.



Kokoszka, Piotr and Matthew Reimherr (Aug. 2017). *Introduction to Functional Data Analysis*. Englisch. 1st ed. Boca Raton: Chapman and Hall/CRC. ISBN: 978-1-4987-4634-2.



Li, Yuanpeng et al. (2020). "Early Diagnosis of Type 2 Diabetes Based on Near-Infrared Spectroscopy Combined With Machine Learning and Aquaphotomics". In: *Frontiers in Chemistry* 8, p. 1133. ISSN: 2296-2646. DOI: 10.3389/fchem.2020.580489. URL: <https://www.frontiersin.org/article/10.3389/fchem.2020.580489>.



Ramsay, James and B. W. Silverman (2005). *Functional Data Analysis*. en. 2nd ed. Springer Series in Statistics. New York: Springer-Verlag. ISBN: 978-0-387-40080-8. DOI: 10.1007/b98888. URL: <https://www.springer.com/de/book/9780387400808> (visited on 10/23/2021).



Reiss, Philip T. and R. Todd Ogden (2007). "Functional Principal Component Regression and Functional Partial Least Squares". In: *Journal of the American Statistical Association* 102.479, pp. 984–996. ISSN: 0162-1459. DOI: 10.1198/016214507000000527.



# Spectral Representation of Random Vectors

Let  $X(\omega)$  be a random vector realizing in  $\mathbb{R}^p$ .

- ▶ Let  $\mu_x = \mathbb{E}(X)$  and  $\Sigma_X = \text{Cov}(X)$
- ▶ Let  $\{\gamma_i \mid i = 1, \dots, p\}$  be the orthonormal **Eigenvectors** of  $\Sigma_X$
- ▶ Let  $\{\lambda_i \mid i = 1, \dots, p\}$  be the corresponding **Eigenvalues** of  $\Sigma_X$

Then  $X$  can also be represented as

$$X(\omega) = \mu_x + \sum_{i=1}^p \xi_i(\omega) \gamma_i$$

where the  $\xi_i(\omega)$  have the following properties

1.  $\mathbb{E}[\xi_i(\omega)] = 0$
2.  $\text{Var}(\xi_i(\omega)) = \lambda_i$
3.  $\text{Cov}(\xi_i(\omega), \xi_j(\omega)) = 0$  for  $i \neq j$

# Principal Component Analysis

A related concept is **Principal Component Analysis** (PCA).

$\Sigma_X$  unknown  $\rightarrow$  **sample analogues**

- ▶ Let  $\mathbf{X} \in \mathbb{R}^{n \times p}$  contain the standardized regressors
- ▶ Let  $\hat{\Sigma}_X = \frac{\mathbf{X}'\mathbf{X}}{n}$
- ▶ Let  $\{\hat{\gamma}_i \mid i = 1, \dots, p\}$  be the orthonormal **Eigenvectors** of  $\hat{\Sigma}_X$
- ▶ Let  $\{\hat{\lambda}_i \mid i = 1, \dots, p\}$  be the corresponding **Eigenvalues** of  $\hat{\Sigma}_X$

Then  $Z_i(\omega) = \hat{\gamma}_i' X(\omega)$  is called the  $i$ 'th principal component and

1.  $\mathbb{E}[Z_i(\omega)] = 0$
2.  $\text{Var}(Z_i(\omega)) = \hat{\lambda}_i$
3.  $\text{Cov}(Z_i(\omega), Z_j(\omega)) = 0$  for  $i \neq j$

▶ Functional Principal Component Analysis

# Estimation with Approximated Functions

$$\begin{aligned} Y(\omega) &= \alpha + \int_0^1 \beta(s) X(\omega)(s) ds + \epsilon(\omega) \\ &= \alpha + \int_0^1 \left[ \left( \sum_{j \in \mathcal{I}} c_j \phi_j(s) \right) \left( \sum_{k \in \mathcal{L}} d_k(\omega) \psi_k(s) \right) \right] ds + \epsilon(\omega) \\ &= \alpha + \sum_{j \in \mathcal{I}} \left[ c_j \sum_{k \in \mathcal{L}} d_k(\omega) \int_0^1 \phi_j(s) \psi_k(s) ds \right] + \epsilon(\omega) \\ &= \alpha + \sum_{j \in \mathcal{I}} \left[ c_j \sum_{k \in \mathcal{L}} Z_{j,k}(\omega) \right] + \epsilon(\omega) \\ &\approx \alpha + \sum_{j=1}^L \left[ c_j \sum_{k=1}^K Z_{j,k}(\omega) \right] + \epsilon(\omega) = \alpha + \sum_{j=1}^L \left[ c_j \tilde{Z}_j^K(\omega) \right] + \epsilon(\omega) \end{aligned}$$

► Equation without Approximated Functions

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↺ 🔍 ↻

## Simulation Results - bspline basis expansion

| f1_e1_spline | f1_e2_spline | f2_e1_spline | f2_e2_spline | n_basis |
|--------------|--------------|--------------|--------------|---------|
| 2.1318       | 71.71        | 0.6082       | 1.5345       | 4       |
| 2.0076       | 71.9395      | 0.3427       | 1.2906       | 5       |
| 1.9937       | 72.2563      | 0.2456       | 1.2014       | 6       |
| 2.0365       | 73.7857      | 0.2797       | 1.2495       | 7       |
| 2.1861       | 79.0485      | 0.1044       | 1.1458       | 8       |
| 2.0511       | 74.1784      | 0.0356       | 1.0217       | 9       |
| 2.1052       | 76.1884      | 0.0297       | 1.0393       | 10      |
| 2.1012       | 76.4031      | 0.0296       | 1.0425       | 11      |
| 2.3815       | 86.1707      | 0.037        | 1.1819       | 12      |
| 2.2114       | 80.6208      | 0.0363       | 1.1024       | 13      |
| 2.4495       | 87.6977      | 0.038        | 1.2126       | 14      |
| 2.2887       | 83.4755      | 0.0315       | 1.1363       | 15      |
| 2.5491       | 93.593       | 0.0352       | 1.2652       | 16      |

► Results

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↺ 🔍 ↻

## Simulation Results - fourier basis expansion

| f1_e1_spline | f1_e2_spline | f2_e1_spline | f2_e2_spline | n_basis |
|--------------|--------------|--------------|--------------|---------|
| 0            | 0            | 0            | 0            | 1       |
| 2.0292       | 72.0085      | 0.512        | 1.4747       | 3       |
| 2.022        | 72.6017      | 0.1616       | 1.1375       | 5       |
| 2.0355       | 73.1836      | 0.0293       | 0.9907       | 7       |
| 2.062        | 73.9631      | 0.029        | 0.9994       | 9       |
| 2.0881       | 75.0921      | 0.0291       | 1.0139       | 11      |
| 2.0997       | 75.9422      | 0.0294       | 1.0291       | 13      |
| 2.1087       | 76.7123      | 0.0298       | 1.0371       | 15      |
| 2.1301       | 77.4908      | 0.03         | 1.0398       | 17      |
| 2.1535       | 78.3943      | 0.0303       | 1.0509       | 19      |
| 2.1775       | 79.2058      | 0.0307       | 1.0617       | 21      |
| 2.2058       | 80.3801      | 0.031        | 1.077        | 23      |
| 2.2372       | 81.509       | 0.0315       | 1.0905       | 25      |

► Results



## Simulation Results - bspline FPCR ( $n_{harm} = 2$ )

| $f_1$ , error 1 | $f_1$ , error 2 | $f_2$ , error 1 | $f_2$ , error 2 | expl. variance | n_basis |
|-----------------|-----------------|-----------------|-----------------|----------------|---------|
| 2.9163          | 10.8768         | 0.9599          | 1.6431          | 1              | 4       |
| 2.345           | 10.585          | 0.8649          | 1.5631          | 0.9776         | 5       |
| 2.4187          | 10.6889         | 0.8739          | 1.5759          | 0.9556         | 6       |
| 2.4971          | 10.7287         | 0.8625          | 1.5723          | 0.9472         | 7       |
| 2.5799          | 10.6971         | 0.8716          | 1.575           | 0.9239         | 8       |
| 2.6828          | 10.7669         | 0.853           | 1.5661          | 0.9178         | 9       |
| 2.825           | 10.7906         | 0.8304          | 1.5622          | 0.8976         | 10      |
| 2.9082          | 10.7774         | 0.8237          | 1.5424          | 0.906          | 11      |
| 2.9519          | 10.8126         | 0.8286          | 1.5561          | 0.9036         | 12      |
| 2.9972          | 10.8221         | 0.8404          | 1.5551          | 0.9052         | 13      |
| 2.9755          | 10.7706         | 0.8396          | 1.5614          | 0.9074         | 14      |
| 2.9762          | 10.7946         | 0.8476          | 1.557           | 0.9058         | 15      |
| 2.9627          | 10.8067         | 0.8609          | 1.5615          | 0.9061         | 16      |

► Results

## Simulation Results - fourier FPCR ( $n_{harm} = 2$ )

| $f_1$ , error 1 | $f_1$ , error 2 | $f_2$ , error 1 | $f_2$ , error 2 | expl. variance | n_basis |
|-----------------|-----------------|-----------------|-----------------|----------------|---------|
| 0               | 0               | 0               | 0               | 0              | 1       |
| 2.2059          | 11.2438         | 0.8768          | 1.5528          | 0.9846         | 3       |
| 2.2148          | 11.2567         | 0.8227          | 1.5215          | 0.9584         | 5       |
| 2.2635          | 11.2662         | 0.8828          | 1.5563          | 0.9489         | 7       |
| 2.2721          | 11.2692         | 0.8811          | 1.5531          | 0.9439         | 9       |
| 2.2797          | 11.2574         | 0.879           | 1.555           | 0.9397         | 11      |
| 2.3039          | 11.2708         | 0.8887          | 1.5591          | 0.9421         | 13      |
| 2.3248          | 11.2898         | 0.8743          | 1.5514          | 0.9283         | 15      |
| 2.3798          | 11.2957         | 0.875           | 1.5511          | 0.9182         | 17      |
| 2.4233          | 11.3049         | 0.8632          | 1.5453          | 0.9167         | 19      |
| 2.4589          | 11.3094         | 0.8619          | 1.5432          | 0.9133         | 21      |
| 2.5306          | 11.331          | 0.8562          | 1.5408          | 0.9082         | 23      |
| 2.5752          | 11.3272         | 0.8531          | 1.5393          | 0.9062         | 25      |

► Results