Erasmus
School of
Economics

Nonlinear Prediction by Kernels Made Explainable

Patrick Groenen and Michael Greenacre

September, 2023

Overview

1. Introduction
2. Linear Kernels
3. Nonlinear KRR
4. Interpretable Kernels
5. Approximated KRR
6. Conclusions

Table of Contents

1. Introduction
2. Linear Kernels
3. Nonlinear KRR
4. Interpretable Kernels
5. Approximated KRR
6. Conclusions

1. Introduction

Dominant method for nonlinear prediction in SVMs: Kernels

- Highly nonlinear decision planes are possible.
- Nonlinearity in the original (primal) problem.
- Nonlinearity possible for any linear model (e.g., kernel ridge regression, kernel logistic regression, etc.)
- Kernels are only possible for loss functions with ridge penalty.
- So far: no interpretation in terms of the original variables.
- Contribution this paper:
  - nonlinear kernels can be interpreted as a linear combination of the original variables.
  - This is a contribution to explainable AI.

1. Introduction

Dominant method for nonlinear prediction in SVMs: Kernels

- Highly nonlinear decision planes are possible.
- Nonlinearity in the original (primal) problem.
- Nonlinearity possible for any linear model (e.g., kernel ridge regression, kernel logistic regression, etc.)
- Kernels are only possible for loss functions with ridge penalty.
- So far: no interpretation in terms of the original variables.
- Contribution this paper:
  - nonlinear kernels can be interpreted as a linear combination of the original variables.
  - This is a contribution to explainable AI.

1. Introduction

Dominant method for nonlinear prediction in SVMs: Kernels

- Highly nonlinear decision planes are possible.
- Nonlinearity in the original (primal) problem.
- Nonlinearity possible for any linear model (e.g., kernel ridge regression, kernel logistic regression, etc.)
- Kernels are only possible for loss functions with ridge penalty.
- So far: no interpretation in terms of the original variables.
- Contribution this paper:
  - nonlinear kernels can be interpreted as a linear combination of the original variables.
  - This is a contribution to explainable AI.

1. Introduction

Dominant method for nonlinear prediction in SVMs: Kernels

- Highly nonlinear decision planes are possible.
- Nonlinearity in the original (primal) problem.
- Nonlinearity possible for any linear model (e.g., kernel ridge regression, kernel logistic regression, etc.)
- Kernels are only possible for loss functions with ridge penalty.
- So far: no interpretation in terms of the original variables.
- Contribution this paper:
  - nonlinear kernels can be interpreted as a linear combination of the original variables.
  - This is a contribution to explainable AI.

1. Introduction

Dominant method for nonlinear prediction in SVMs: Kernels

- Highly nonlinear decision planes are possible.
- Nonlinearity in the original (primal) problem.
- Nonlinearity possible for any linear model (e.g., kernel ridge regression, kernel logistic regression, etc.)
- Kernels are only possible for loss functions with ridge penalty.
- So far: no interpretation in terms of the original variables.
- Contribution this paper:
  - nonlinear kernels can be interpreted as a linear combination of the original variables.
  - This is a contribution to explainable AI.

1. Introduction

Dominant method for nonlinear prediction in SVMs: Kernels

- Highly nonlinear decision planes are possible.
- Nonlinearity in the original (primal) problem.
- Nonlinearity possible for any linear model (e.g., kernel ridge regression, kernel logistic regression, etc.)
- Kernels are only possible for loss functions with ridge penalty.
- So far: no interpretation in terms of the original variables.
- Contribution this paper:
    - nonlinear kernels can be interpreted as a linear combination of the original variables.
    - This is a contribution to explainable AI.

1. Introduction

Dominant method for nonlinear prediction in SVMs: Kernels

- Highly nonlinear decision planes are possible.
- Nonlinearity in the original (primal) problem.
- Nonlinearity possible for any linear model (e.g., kernel ridge regression, kernel logistic regression, etc.)
- Kernels are only possible for loss functions with ridge penalty.
- So far: no interpretation in terms of the original variables.
- Contribution this paper:
  - nonlinear kernels can be interpreted as a linear combination of the original variables.
  - This is a contribution to explainable AI.

1. Introduction

Dominant method for nonlinear prediction in SVMs: Kernels

- Highly nonlinear decision planes are possible.
- Nonlinearity in the original (primal) problem.
- Nonlinearity possible for any linear model (e.g., kernel ridge regression, kernel logistic regression, etc.)
- Kernels are only possible for loss functions with ridge penalty.
- So far: no interpretation in terms of the original variables.
- Contribution this paper:
  - nonlinear kernels can be interpreted as a linear combination of the original variables.
  - This is a contribution to explainable AI.

1. Introduction

Kernels

- Kernels make use of the same trick as polynomial basis expansion or spline transformations.
- Requires a ridge penalty: $\lambda \mathbf{w}^\top \mathbf{w}$, e.g., in kernel ridge regression (KRR) or support vector machines (SVM).
- Maps $\mathbf{x}_i$ (row i of $\mathbf{X}$) to ϕ_i in some high dimensional space.
- Fit the model linearly in the high dimensional space.
- Then, at most n parameters need to be optimized through a dual approach.

1. Introduction

Kernels

- Kernels make use of the same trick as polynomial basis expansion or spline transformations.
- Requires a ridge penalty: $\lambda \mathbf{w}^\top \mathbf{w}$, e.g., in kernel ridge regression (KRR) or support vector machines (SVM).
- Maps $\mathbf{x}_i$ (row i of $\mathbf{X}$) to ϕ_i in some high dimensional space.
- Fit the model linearly in the high dimensional space.
- Then, at most n parameters need to be optimized through a dual approach.

1. Introduction

Kernels

- Kernels make use of the same trick as polynomial basis expansion or spline transformations.
- Requires a ridge penalty: $\lambda \mathbf{w}^\top \mathbf{w}$, e.g., in kernel ridge regression (KRR) or support vector machines (SVM).
- Maps $\mathbf{x}_i$ (row i of $\mathbf{X}$) to ϕ_i in some high dimensional space.
- Fit the model linearly in the high dimensional space.
- Then, at most n parameters need to be optimized through a dual approach.

1. Introduction

Kernels

- Kernels make use of the same trick as polynomial basis expansion or spline transformations.
- Requires a ridge penalty: $\lambda \mathbf{w}^\top \mathbf{w}$, e.g., in kernel ridge regression (KRR) or support vector machines (SVM).
- Maps $\mathbf{x}_i$ (row i of $\mathbf{X}$) to ϕ_i in some high dimensional space.
- Fit the model linearly in the high dimensional space.
- Then, at most n parameters need to be optimized through a dual approach.

1. Introduction

Kernels

- Kernels make use of the same trick as polynomial basis expansion or spline transformations.
- Requires a ridge penalty: $\lambda \mathbf{w}^\top \mathbf{w}$, e.g., in kernel ridge regression (KRR) or support vector machines (SVM).
- Maps $\mathbf{x}_i$ (row i of $\mathbf{X}$) to ϕ_i in some high dimensional space.
- Fit the model linearly in the high dimensional space.
- Then, at most n parameters need to be optimized through a dual approach.

Table of Contents

1. Introduction
2. Linear Kernels
3. Nonlinear KRR
4. Interpretable Kernels
5. Approximated KRR
6. Conclusions

2. Linear Kernel

Ridge regression

- Loss function ridge regression:

$$L_{\text{ridge}}(w_0, \mathbf{w}) \quad = \quad \|\mathbf{y} - (w_0\mathbf{1} + \mathbf{X}\mathbf{w})\|^2 + \lambda \mathbf{w}^\top \mathbf{w}$$

- The vector of predicted values is: $\hat{\mathbf{y}} = \mathbf{q} = w_0\mathbf{1} + \mathbf{X}\mathbf{w}$
- The intercept w_0 complicates things; therefore, we set $\tilde{\mathbf{q}} = \mathbf{X}\mathbf{w}$ so that

$$\mathbf{q} = w_0\mathbf{1} + \mathbf{X}\mathbf{w} = w_0\mathbf{1} + \tilde{\mathbf{q}}$$

2. Linear Kernel

Ridge regression

- Loss function ridge regression:

$$L_{\text{ridge}}(w_0, \mathbf{w}) \quad = \quad \|\mathbf{y} - (w_0\mathbf{1} + \mathbf{X}\mathbf{w})\|^2 + \lambda\mathbf{w}^\top\mathbf{w}$$

- The vector of predicted values is: $\hat{\mathbf{y}} = \mathbf{q} = w_0\mathbf{1} + \mathbf{X}\mathbf{w}$
- The intercept w_0 complicates things; therefore, we set $\tilde{\mathbf{q}} = \mathbf{X}\mathbf{w}$ so that

$$\mathbf{q} = w_0\mathbf{1} + \mathbf{X}\mathbf{w} = w_0\mathbf{1} + \tilde{\mathbf{q}}$$

2. Linear Kernel

Ridge regression

- Loss function ridge regression:

$$L_{\text{ridge}}(w_0, \mathbf{w}) \quad = \quad \|\mathbf{y} - (w_0\mathbf{1} + \mathbf{X}\mathbf{w})\|^2 + \lambda\mathbf{w}^\top\mathbf{w}$$

- The vector of predicted values is: $\hat{\mathbf{y}} = \mathbf{q} = w_0\mathbf{1} + \mathbf{X}\mathbf{w}$
- The intercept w_0 complicates things; therefore, we set $\tilde{\mathbf{q}} = \mathbf{X}\mathbf{w}$ so that

$$\mathbf{q} = w_0\mathbf{1} + \mathbf{X}\mathbf{w} = w_0\mathbf{1} + \tilde{\mathbf{q}}$$

2. Linear Kernel

A dual approach for KRR:

- Basic idea of the dual approach:
  If $p \gg n$ (and $\mathbf{X}$ has rank n), then switch to the minimization over $\mathbf{q}$ (n parameters) instead of w_0 and $\mathbf{w}$ ($p + 1$ parameters)

2. Linear Kernel

Towards a dual approach:

- Example of an $\mathbf{X}$ with $n < p$: $n = 2, p = 3$

$$\mathbf{X} = \begin{bmatrix} -.25 & .75 & .50 \\ .50 & .50 & .50 \end{bmatrix}$$

- Choose (e.g.)

$$\mathbf{w} = \begin{bmatrix} .25 \\ -.50 \\ .50 \end{bmatrix}$$

- Then, the $n \times 1 = 2 \times 1$ vector $\tilde{\mathbf{q}}$ must be in the linear space spanned by $\mathbf{x}_1$ and $\mathbf{x}_2$

$$\tilde{\mathbf{q}} = \mathbf{X}\mathbf{w} = \mathbf{X}\mathbf{w}_1 = \begin{bmatrix} -.1875 \\ .1250 \end{bmatrix}$$

2. Linear Kernel

Towards a dual approach:

- Example of an $\mathbf{X}$ with $n < p$: $n = 2, p = 3$

$$\mathbf{X} = \begin{bmatrix} -.25 & .75 & .50 \\ .50 & .50 & .50 \end{bmatrix}$$

- Choose (e.g.)

$$\mathbf{w} = \begin{bmatrix} .25 \\ -.50 \\ .50 \end{bmatrix}$$

- Then, the $n \times 1 = 2 \times 1$ vector $\tilde{\mathbf{q}}$ must be in the linear space spanned by $\mathbf{x}_1$ and $\mathbf{x}_2$

$$\tilde{\mathbf{q}} = \mathbf{X}\mathbf{w} = \mathbf{X}\mathbf{w}_1 = \begin{bmatrix} -.1875 \\ .1250 \end{bmatrix}$$

2. Linear Kernel

Towards a dual approach:

- Example of an $\mathbf{X}$ with $n < p$: $n = 2, p = 3$

$$\mathbf{X} = \begin{bmatrix} -.25 & .75 & .50 \\ .50 & .50 & .50 \end{bmatrix}$$

- Choose (e.g.)

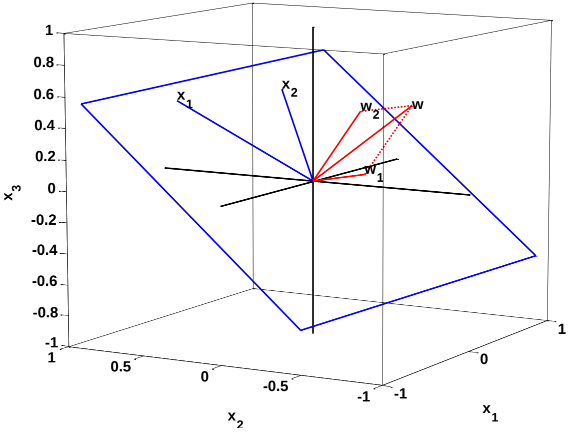
$$\mathbf{w} = \begin{bmatrix} .25 \\ -.50 \\ .50 \end{bmatrix}$$

- Then, the $n \times 1 = 2 \times 1$ vector $\tilde{\mathbf{q}}$ must be in the linear space spanned by $\mathbf{x}_1$ and $\mathbf{x}_2$

$$\tilde{\mathbf{q}} = \mathbf{X}\mathbf{w} = \mathbf{X}\mathbf{w}_1 = \begin{bmatrix} -.1875 \\ .1250 \end{bmatrix}$$

2. Linear Kernel

Towards a dual approach:



2. Linear Kernel

Steps to arrive at a dual ridge regression formulation:

1. Decompose $\mathbf{w} = \mathbf{w}_1 + \mathbf{w}_2$ with a part that is in the linear space of $\mathbf{X}$ ($\mathbf{w}_1$) and a part that is orthogonal to the linear space of $\mathbf{X}$ ($\mathbf{w}_2$).
2. $\tilde{\mathbf{q}}$ depends only on $\mathbf{w}_1$ and not on $\mathbf{w}_2$.
3. Penalty term has $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \mathbf{w}_1^\top \mathbf{w}_1$ because $\mathbf{w}_2^\top \mathbf{w}_2 = 0$.
4. Penalty term equals $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \tilde{\mathbf{q}}^\top (\mathbf{X}\mathbf{X}^\top)^{-1} \tilde{\mathbf{q}}$ where the $n \times n$ matrix $\mathbf{X}\mathbf{X}^\top$ has elements $\mathbf{x}_i^\top \mathbf{x}_{i'}$. Proof A.1
5. Without loss of generality, we may optimize directly over the n parameters $\tilde{q}_i$ without any restriction.
6. $L_{\text{ridge}}(w_0, \tilde{\mathbf{q}})$ is now only a function of w_0 and $\tilde{q}_i$.

2. Linear Kernel

Steps to arrive at a dual ridge regression formulation:

1. Decompose $\mathbf{w} = \mathbf{w}_1 + \mathbf{w}_2$ with a part that is in the linear space of $\mathbf{X}$ ($\mathbf{w}_1$) and a part that is orthogonal to the linear space of $\mathbf{X}$ ($\mathbf{w}_2$).
2. $\tilde{\mathbf{q}}$ depends only on $\mathbf{w}_1$ and not on $\mathbf{w}_2$.
3. Penalty term has $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \mathbf{w}_1^\top \mathbf{w}_1$ because $\mathbf{w}_2^\top \mathbf{w}_2 = 0$.
4. Penalty term equals $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \tilde{\mathbf{q}}^\top (\mathbf{X}\mathbf{X}^\top)^{-1} \tilde{\mathbf{q}}$ where the $n \times n$ matrix $\mathbf{X}\mathbf{X}^\top$ has elements $\mathbf{x}_i^\top \mathbf{x}_{i'}$. Proof A.1
5. Without loss of generality, we may optimize directly over the n parameters $\tilde{q}_i$ without any restriction.
6. $L_{\text{ridge}}(w_0, \tilde{\mathbf{q}})$ is now only a function of w_0 and $\tilde{q}_i$.

2. Linear Kernel

Steps to arrive at a dual ridge regression formulation:

1. Decompose $\mathbf{w} = \mathbf{w}_1 + \mathbf{w}_2$ with a part that is in the linear space of $\mathbf{X}$ ($\mathbf{w}_1$) and a part that is orthogonal to the linear space of $\mathbf{X}$ ($\mathbf{w}_2$).
2. $\tilde{\mathbf{q}}$ depends only on $\mathbf{w}_1$ and not on $\mathbf{w}_2$.
3. Penalty term has $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \mathbf{w}_1^\top \mathbf{w}_1$ because $\mathbf{w}_2^\top \mathbf{w}_2 = 0$.
4. Penalty term equals $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \tilde{\mathbf{q}}^\top (\mathbf{X}\mathbf{X}^\top)^{-1} \tilde{\mathbf{q}}$ where the $n \times n$ matrix $\mathbf{X}\mathbf{X}^\top$ has elements $\mathbf{x}_i^\top \mathbf{x}_{i'}$. Proof A.1
5. Without loss of generality, we may optimize directly over the n parameters $\tilde{q}_i$ without any restriction.
6. $L_{\text{ridge}}(w_0, \tilde{\mathbf{q}})$ is now only a function of w_0 and $\tilde{q}_i$.

2. Linear Kernel

Steps to arrive at a dual ridge regression formulation:

1. Decompose $\mathbf{w} = \mathbf{w}_1 + \mathbf{w}_2$ with a part that is in the linear space of $\mathbf{X}$ ($\mathbf{w}_1$) and a part that is orthogonal to the linear space of $\mathbf{X}$ ($\mathbf{w}_2$).
2. $\tilde{\mathbf{q}}$ depends only on $\mathbf{w}_1$ and not on $\mathbf{w}_2$.
3. Penalty term has $\lambda\mathbf{w}^\top\mathbf{w} = \lambda\mathbf{w}_1^\top\mathbf{w}_1$ because $\mathbf{w}_2^\top\mathbf{w}_2 = 0$.
4. Penalty term equals $\lambda\mathbf{w}^\top\mathbf{w} = \lambda\tilde{\mathbf{q}}^\top(\mathbf{X}\mathbf{X}^\top)^{-1}\tilde{\mathbf{q}}$ where the $n \times n$ matrix $\mathbf{X}\mathbf{X}^\top$ has elements $\mathbf{x}_i^\top\mathbf{x}_{i'}$. Proof A.1
5. Without loss of generality, we may optimize directly over the n parameters $\tilde{q}_i$ without any restriction.
6. $L_{\text{ridge}}(w_0, \tilde{\mathbf{q}})$ is now only a function of w_0 and $\tilde{q}_i$.

2. Linear Kernel

Steps to arrive at a dual ridge regression formulation:

1. Decompose $\mathbf{w} = \mathbf{w}_1 + \mathbf{w}_2$ with a part that is in the linear space of $\mathbf{X}$ ($\mathbf{w}_1$) and a part that is orthogonal to the linear space of $\mathbf{X}$ ($\mathbf{w}_2$).
2. $\tilde{\mathbf{q}}$ depends only on $\mathbf{w}_1$ and not on $\mathbf{w}_2$.
3. Penalty term has $\lambda\mathbf{w}^\top\mathbf{w} = \lambda\mathbf{w}_1^\top\mathbf{w}_1$ because $\mathbf{w}_2^\top\mathbf{w}_2 = 0$.
4. Penalty term equals $\lambda\mathbf{w}^\top\mathbf{w} = \lambda\tilde{\mathbf{q}}^\top(\mathbf{X}\mathbf{X}^\top)^{-1}\tilde{\mathbf{q}}$ where the $n \times n$ matrix $\mathbf{X}\mathbf{X}^\top$ has elements $\mathbf{x}_i^\top\mathbf{x}_{i'}$. Proof A.1
5. Without loss of generality, we may optimize directly over the n parameters $\tilde{q}_i$ without any restriction.
6. $L_{\text{ridge}}(w_0, \tilde{\mathbf{q}})$ is now only a function of w_0 and $\tilde{q}_i$.

2. Linear Kernel

Steps to arrive at a dual ridge regression formulation:

1. Decompose $\mathbf{w} = \mathbf{w}_1 + \mathbf{w}_2$ with a part that is in the linear space of $\mathbf{X}$ ($\mathbf{w}_1$) and a part that is orthogonal to the linear space of $\mathbf{X}$ ($\mathbf{w}_2$).
2. $\tilde{\mathbf{q}}$ depends only on $\mathbf{w}_1$ and not on $\mathbf{w}_2$.
3. Penalty term has $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \mathbf{w}_1^\top \mathbf{w}_1$ because $\mathbf{w}_2^\top \mathbf{w}_2 = 0$.
4. Penalty term equals $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \tilde{\mathbf{q}}^\top (\mathbf{X}\mathbf{X}^\top)^{-1} \tilde{\mathbf{q}}$ where the $n \times n$ matrix $\mathbf{X}\mathbf{X}^\top$ has elements $\mathbf{x}_i^\top \mathbf{x}_{i'}$. Proof A.1
5. Without loss of generality, we may optimize directly over the n parameters $\tilde{q}_i$ without any restriction.
6. $L_{\text{ridge}}(w_0, \tilde{\mathbf{q}})$ is now only a function of w_0 and $\tilde{q}_i$.

2. Linear Kernel

- The loss of linear KRR L_{ridge} is now only a function of w_0 and $\tilde{q}_i$:

$$L_{\text{ridge}}(w_0, \tilde{\mathbf{q}}) = \underset{\substack{\uparrow \\ \boxed{\text{Regression term}}}}{\|\mathbf{y} - (w_0\mathbf{1} + \tilde{\mathbf{q}})\|^2} \quad \underset{\substack{\uparrow \\ \boxed{\text{Penalty term}}}}{+\lambda\tilde{\mathbf{q}}^\top(\mathbf{X}\mathbf{X}^\top)^{-1}\tilde{\mathbf{q}}}$$

Table of Contents

1. Introduction
2. Linear Kernels
3. Nonlinear KRR
4. Interpretable Kernels
5. Approximated KRR
6. Conclusions

3. Nonlinear Kernels

Kernels for nonlinear prediction:

- Kernels make use of same dual trick for $p \gg n$.
- Replace the all the variables in $\mathbf{X}$ by their $n \times k$ kernel basis $\Phi(\mathbf{X})$ or Φ for short.
- The equivalent of matrix $\mathbf{X}\mathbf{X}^\top$ becomes the $n \times n$ kernel matrix $\mathbf{K} = \Phi\Phi^\top$ with elements $k_{ii'} = \phi_i^\top \phi_{i'}$
- Kernel trick: choose smart Φ such that k_{ij} can be directly computed from rows $\mathbf{x}_i$ and $\mathbf{x}_{i'}$.
- Kernel ridge regression loss equals:

$$L_{\text{KRR}}(w_0, \tilde{\mathbf{q}}) = \|\mathbf{y} - (w_0\mathbf{1} + \tilde{\mathbf{q}})\|^2 + \lambda \tilde{\mathbf{q}}^\top \mathbf{K}^{-1} \tilde{\mathbf{q}}$$

- For out-of-sample prediction, see Appendix A.2

3. Nonlinear Kernels

Kernels for nonlinear prediction:

- Kernels make use of same dual trick for $p \gg n$.
- Replace the all the variables in $\mathbf{X}$ by their $n \times k$ kernel basis $\mathbf{\Phi}(\mathbf{X})$ or $\mathbf{\Phi}$ for short.
- The equivalent of matrix $\mathbf{X}\mathbf{X}^\top$ becomes the $n \times n$ kernel matrix $\mathbf{K} = \mathbf{\Phi}\mathbf{\Phi}^\top$ with elements $k_{ii'} = \phi_i^\top \phi_{i'}$
- Kernel trick: choose smart $\mathbf{\Phi}$ such that k_{ij} can be directly computed from rows $\mathbf{x}_i$ and $\mathbf{x}_{i'}$.
- Kernel ridge regression loss equals:

$$L_{\text{KRR}}(w_0, \tilde{\mathbf{q}}) = \|\mathbf{y} - (w_0\mathbf{1} + \tilde{\mathbf{q}})\|^2 + \lambda \tilde{\mathbf{q}}^\top \mathbf{K}^{-1} \tilde{\mathbf{q}}$$

- For out-of-sample prediction, see Appendix A.2

3. Nonlinear Kernels

Kernels for nonlinear prediction:

- Kernels make use of same dual trick for $p \gg n$.
- Replace the all the variables in $\mathbf{X}$ by their $n \times k$ kernel basis $\mathbf{\Phi}(\mathbf{X})$ or $\mathbf{\Phi}$ for short.
- The equivalent of matrix $\mathbf{X}\mathbf{X}^\top$ becomes the $n \times n$ kernel matrix $\mathbf{K} = \mathbf{\Phi}\mathbf{\Phi}^\top$ with elements $k_{ii'} = \phi_i^\top \phi_{i'}$
- Kernel trick: choose smart $\mathbf{\Phi}$ such that k_{ij} can be directly computed from rows $\mathbf{x}_i$ and $\mathbf{x}_{i'}$.
- Kernel ridge regression loss equals:

$$L_{\text{KRR}}(w_0, \tilde{\mathbf{q}}) = \|\mathbf{y} - (w_0\mathbf{1} + \tilde{\mathbf{q}})\|^2 + \lambda \tilde{\mathbf{q}}^\top \mathbf{K}^{-1} \tilde{\mathbf{q}}$$

- For out-of-sample prediction, see Appendix A.2

3. Nonlinear Kernels

Kernels for nonlinear prediction:

- Kernels make use of same dual trick for $p \gg n$.
- Replace the all the variables in $\mathbf{X}$ by their $n \times k$ kernel basis $\mathbf{\Phi}(\mathbf{X})$ or $\mathbf{\Phi}$ for short.
- The equivalent of matrix $\mathbf{X}\mathbf{X}^\top$ becomes the $n \times n$ kernel matrix $\mathbf{K} = \mathbf{\Phi}\mathbf{\Phi}^\top$ with elements $k_{ii'} = \phi_i^\top \phi_{i'}$
- Kernel trick: choose smart $\mathbf{\Phi}$ such that k_{ij} can be directly computed from rows $\mathbf{x}_i$ and $\mathbf{x}_{i'}$.
- Kernel ridge regression loss equals:

$$L_{\text{KRR}}(w_0, \tilde{\mathbf{q}}) = \|\mathbf{y} - (w_0\mathbf{1} + \tilde{\mathbf{q}})\|^2 + \lambda \tilde{\mathbf{q}}^\top \mathbf{K}^{-1} \tilde{\mathbf{q}}$$

- For out-of-sample prediction, see Appendix A.2

3. Nonlinear Kernels

Kernels for nonlinear prediction:

- Kernels make use of same dual trick for $p \gg n$.
- Replace the all the variables in $\mathbf{X}$ by their $n \times k$ kernel basis $\mathbf{\Phi}(\mathbf{X})$ or $\mathbf{\Phi}$ for short.
- The equivalent of matrix $\mathbf{X}\mathbf{X}^\top$ becomes the $n \times n$ kernel matrix $\mathbf{K} = \mathbf{\Phi}\mathbf{\Phi}^\top$ with elements $k_{ii'} = \phi_i^\top \phi_{i'}$
- Kernel trick: choose smart $\mathbf{\Phi}$ such that k_{ij} can be directly computed from rows $\mathbf{x}_i$ and $\mathbf{x}_{i'}$.
- Kernel ridge regression loss equals:

$$L_{\text{KRR}}(w_0, \tilde{\mathbf{q}}) = \|\mathbf{y} - (w_0\mathbf{1} + \tilde{\mathbf{q}})\|^2 + \lambda \tilde{\mathbf{q}}^\top \mathbf{K}^{-1} \tilde{\mathbf{q}}$$

- For out-of-sample prediction, see Appendix A.2

3. Nonlinear Kernels

Kernels for nonlinear prediction:

- Kernels make use of same dual trick for $p \gg n$.
- Replace the all the variables in $\mathbf{X}$ by their $n \times k$ kernel basis $\mathbf{\Phi}(\mathbf{X})$ or $\mathbf{\Phi}$ for short.
- The equivalent of matrix $\mathbf{X}\mathbf{X}^\top$ becomes the $n \times n$ kernel matrix $\mathbf{K} = \mathbf{\Phi}\mathbf{\Phi}^\top$ with elements $k_{ii'} = \phi_i^\top \phi_{i'}$
- Kernel trick: choose smart $\mathbf{\Phi}$ such that k_{ij} can be directly computed from rows $\mathbf{x}_i$ and $\mathbf{x}_{i'}$.
- Kernel ridge regression loss equals:

$$L_{\text{KRR}}(w_0, \tilde{\mathbf{q}}) = \|\mathbf{y} - (w_0\mathbf{1} + \tilde{\mathbf{q}})\|^2 + \lambda \tilde{\mathbf{q}}^\top \mathbf{K}^{-1} \tilde{\mathbf{q}}$$

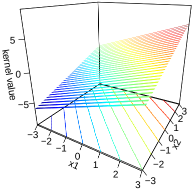
- For out-of-sample prediction, see Appendix A.2

3. Nonlinear Kernels

Three examples of kernels:

linear

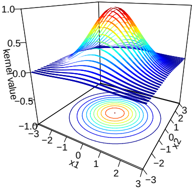
$$k_{ii'} = \mathbf{x}_i^\top \mathbf{x}_{i'}$$



radial basis function function (RBF)

$$k_{ii'} = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_{i'}\|^2}$$

with fixed $\gamma > 0$



inhomogeneous polynomial

$$k_{ii'} = (1 + \mathbf{x}_i^\top \mathbf{x}_{i'})^d$$

with fixed $d > 0$

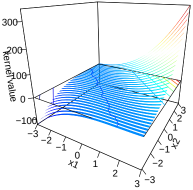


Table of Contents

1. Introduction

2. Linear Kernels

3. Nonlinear KRR

4. Interpretable Kernels

5. Approximated KRR

6. Conclusions

4. Interpretable Kernels

Kernels

- Kernels in regression or SVM can be used for nonlinear prediction.
- Often combined with quadratic ridge penalty against overfitting.
- Problem so far:
  - No interpretation in original predictor variables in the $n \times p$ matrix $\mathbf{X}$.
  - Use of kernels in, e.g., regression and SVM is a black-box method.

4. Interpretable Kernels

Kernels

- Kernels in regression or SVM can be used for nonlinear prediction.
- Often combined with quadratic ridge penalty against overfitting.
- Problem so far:
    - No interpretation in original predictor variables in the $n \times p$ matrix $\mathbf{X}$.
    - Use of kernels in, e.g., regression and SVM is a black-box method.

4. Interpretable Kernels

Kernels

- Kernels in regression or SVM can be used for nonlinear prediction.
- Often combined with quadratic ridge penalty against overfitting.
- Problem so far:
  - No interpretation in original predictor variables in the $n \times p$ matrix $\mathbf{X}$.
  - Use of kernels in, e.g., regression and SVM is a black-box method.

4. Interpretable Kernels

Kernels

- Kernels in regression or SVM can be used for nonlinear prediction.
- Often combined with quadratic ridge penalty against overfitting.
- Problem so far:
  - No interpretation in original predictor variables in the $n \times p$ matrix $\mathbf{X}$.
  - Use of kernels in, e.g., regression and SVM is a black-box method.

4. Interpretable Kernels

Kernels

- Kernels in regression or SVM can be used for nonlinear prediction.
- Often combined with quadratic ridge penalty against overfitting.
- Problem so far:
  - No interpretation in original predictor variables in the $n \times p$ matrix $\mathbf{X}$.
  - Use of kernels in, e.g., regression and SVM is a black-box method.

4. Interpretable Kernels

Contribution this paper

- Introduce approximated kernel ridge regression (AKRR) where kernel matrix is approximated by $\mathbf{X}$
- Express the kernel predictions as linear combination in $\mathbf{X}$:
  - If $n-1 \leq p$ (and rank($\mathbf{X}$) $= n-1$) then approximation is exact equivalence.
  - If $n-1 > p$ (or rank($\mathbf{X}$) $< n-1$) then the kernel solution can be linearly approximated.
  - Provide a solution for the interpretation of nonlinear prediction through kernels.
- Contributes to explainable artificial intelligence (AI).

4. Interpretable Kernels

Contribution this paper

- Introduce approximated kernel ridge regression (AKRR) where kernel matrix is approximated by $\mathbf{X}$
- Express the kernel predictions as linear combination in $\mathbf{X}$:
  - If $n-1 \leq p$ (and $\text{rank}(\mathbf{X}) = n-1$) then approximation is exact equivalence.
  - If $n-1 > p$ (or $\text{rank}(\mathbf{X}) < n-1$) then the kernel solution can be linearly approximated.
  - Provide a solution for the interpretation of nonlinear prediction through kernels.
- Contributes to explainable artificial intelligence (AI).

4. Interpretable Kernels

Contribution this paper

- Introduce approximated kernel ridge regression (AKRR) where kernel matrix is approximated by $\mathbf{X}$
- Express the kernel predictions as linear combination in $\mathbf{X}$:
  - If $n-1 \leq p$ (and $\text{rank}(\mathbf{X}) = n-1$) then approximation is exact equivalence.
  - If $n-1 > p$ (or $\text{rank}(\mathbf{X}) < n-1$) then the kernel solution can be linearly approximated.
  - Provide a solution for the interpretation of nonlinear prediction through kernels.
- Contributes to explainable artificial intelligence (AI).

4. Interpretable Kernels

Contribution this paper

- Introduce approximated kernel ridge regression (AKRR) where kernel matrix is approximated by $\mathbf{X}$
- Express the kernel predictions as linear combination in $\mathbf{X}$:
  - If $n - 1 \leq p$ (and $\text{rank}(\mathbf{X}) = n - 1$) then approximation is exact equivalence.
  - If $n - 1 > p$ (or $\text{rank}(\mathbf{X}) < n - 1$) then the kernel solution can be linearly approximated.
  - Provide a solution for the interpretation of nonlinear prediction through kernels.
- Contributes to explainable artificial intelligence (AI).

4. Interpretable Kernels

Contribution this paper

- Introduce approximated kernel ridge regression (AKRR) where kernel matrix is approximated by $\mathbf{X}$
- Express the kernel predictions as linear combination in $\mathbf{X}$:
  - If $n-1 \leq p$ (and $\text{rank}(\mathbf{X}) = n-1$) then approximation is exact equivalence.
  - If $n-1 > p$ (or $\text{rank}(\mathbf{X}) < n-1$) then the kernel solution can be linearly approximated.
  - Provide a solution for the interpretation of nonlinear prediction through kernels.
- Contributes to explainable artificial intelligence (AI).

4. Interpretable Kernels

Contribution this paper

- Introduce approximated kernel ridge regression (AKRR) where kernel matrix is approximated by $\mathbf{X}$
- Express the kernel predictions as linear combination in $\mathbf{X}$:
  - If $n-1 \leq p$ (and $\text{rank}(\mathbf{X}) = n-1$) then approximation is exact equivalence.
  - If $n-1 > p$ (or $\text{rank}(\mathbf{X}) < n-1$) then the kernel solution can be linearly approximated.
  - Provide a solution for the interpretation of nonlinear prediction through kernels.
- Contributes to explainable artificial intelligence (AI).

4. Interpretable Kernels

Main result (for KRR and $n-1 \leq p$):

$$\begin{aligned} L_{\text{KRR}}(w_0, \tilde{\mathbf{q}}) &= \|\mathbf{y} - (w_0\mathbf{1} + \tilde{\mathbf{q}})\|^2 + \lambda \tilde{\mathbf{q}}^\top \mathbf{K}^{-1}\tilde{\mathbf{q}} \\ &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^{\mathsf{T}}\mathbf{A}\gamma = L_{\text{AKRR}}(\gamma) \end{aligned}$$

with

- $\mathbf{A} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)\left(\mathbf{X}^{\mathsf{T}}\mathbf{K}\mathbf{X}\right)^{-1}\left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)$
- γ: p vector with weights.

Table of Contents

1. Introduction

2. Linear Kernels

3. Nonlinear KRR

4. Interpretable Kernels

5. Approximated KRR

6. Conclusions

5. Approximated KRR

Linearly Approximated KRR:

- Two steps:
  1. Approximate the kernel space Φ by $\mathbf{XB}$ through (classical) multidimensional scaling (MDS) through strain loss:

$$L_{\text{Strain}}(\mathbf{B}) \quad = \quad \|\mathbf{K} - \mathbf{XBB}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\|^2$$

  2. Do a ridge regression with predictors $\mathbf{X}$ and adapted ridge penalty

5. Approximated KRR

Linearly Approximated KRR:

- Two steps:
  1. Approximate the kernel space $\boldsymbol{\Phi}$ by $\mathbf{XB}$ through (classical) multidimensional scaling (MDS) through strain loss:

$$L_{\text{Strain}}(\mathbf{B}) \quad = \quad \|\mathbf{K} - \mathbf{XBB}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\|^2$$

  2. Do a ridge regression with predictors $\mathbf{X}$ and adapted ridge penalty

5. Approximated KRR

Linearly Approximated KRR:

- Two steps:
  1. Approximate the kernel space $\mathbf{\Phi}$ by $\mathbf{XB}$ through (classical) multidimensional scaling (MDS) through strain loss:

$$L_{\text{Strain}}(\mathbf{B}) \quad = \quad \|\mathbf{K} - \mathbf{XB}\mathbf{B}^\top\mathbf{X}^\top\|^2$$

  2. Do a ridge regression with predictors $\mathbf{X}$ and adapted ridge penalty

5. Step 1 of Approximated KRR

Step 1 Linearly Approximated KRR: Minimize $L_{\text{Strain}}(\mathbf{B}) = \|\mathbf{K} - \mathbf{X}\mathbf{B}\mathbf{B}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\|^2$

- Solution:
  - Computing the eigendecomposition

$$(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-1/2}\mathbf{X}^{\mathsf{T}}\mathbf{K}\mathbf{X}(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-1/2} = \mathbf{Q}\boldsymbol{\Gamma}\mathbf{Q}^{\mathsf{T}} = (\mathbf{Q}\boldsymbol{\Gamma}^{1/2})(\mathbf{Q}\boldsymbol{\Gamma}^{1/2})^{\mathsf{T}}$$

  - and weight matrix $\mathbf{B}$

$$\mathbf{B}^* \quad = \quad (\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-1/2}\mathbf{Q}\boldsymbol{\Gamma}^{1/2}$$

    with $(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{\alpha} = \mathbf{P}\boldsymbol{\Sigma}^{\alpha}\mathbf{P}^{\mathsf{T}}$ and σ_{ii}^{α} the power α of the nonzero eigenvalues of $\mathbf{X}^{\mathsf{T}}\mathbf{X}$ and $\sigma_{ii}^{\alpha} = 0$ otherwise.

- If $n - 1 < p$ then $L_{\text{Strain}}(\mathbf{B}^*) = 0$ (perfect reconstruction of $\mathbf{K}$).
- If $n - 1 > p$, then $L_{\text{Strain}}(\mathbf{B}^*) > 0$ (approximated reconstruction of $\mathbf{K}$)

5. Step 1 of Approximated KRR

Step 1 Linearly Approximated KRR: Minimize $L_{\text{Strain}}(\mathbf{B}) = \|\mathbf{K} - \mathbf{X}\mathbf{B}\mathbf{B}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\|^2$

- Solution:
  - Computing the eigendecomposition

$$(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-1/2}\mathbf{X}^{\mathsf{T}}\mathbf{K}\mathbf{X}(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-1/2} = \mathbf{Q}\boldsymbol{\Gamma}\mathbf{Q}^{\mathsf{T}} = (\mathbf{Q}\boldsymbol{\Gamma}^{1/2})(\mathbf{Q}\boldsymbol{\Gamma}^{1/2})^{\mathsf{T}}$$

  - and weight matrix $\mathbf{B}$

$$\mathbf{B}^* = (\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-1/2}\mathbf{Q}\boldsymbol{\Gamma}^{1/2}$$

    with $(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{\alpha} = \mathbf{P}\boldsymbol{\Sigma}^{\alpha}\mathbf{P}^{\mathsf{T}}$ and σ_{ii}^{α} the power α of the nonzero eigenvalues of $\mathbf{X}^{\mathsf{T}}\mathbf{X}$ and $\sigma_{ii}^{\alpha} = 0$ otherwise.

- If $n - 1 < p$ then $L_{\text{Strain}}(\mathbf{B}^*) = 0$ (perfect reconstruction of $\mathbf{K}$).
- If $n - 1 > p$, then $L_{\text{Strain}}(\mathbf{B}^*) > 0$ (approximated reconstruction of $\mathbf{K}$)

5. Step 1 of Approximated KRR

Step 1 Linearly Approximated KRR: Minimize $L_{\text{Strain}}(\mathbf{B}) = \|\mathbf{K} - \mathbf{X}\mathbf{B}\mathbf{B}^\mathsf{T}\mathbf{X}^\mathsf{T}\|^2$

- Solution:
  - Computing the eigendecomposition

$$(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1/2}\mathbf{X}^\mathsf{T}\mathbf{K}\mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1/2} = \mathbf{Q}\boldsymbol{\Gamma}\mathbf{Q}^\mathsf{T} = (\mathbf{Q}\boldsymbol{\Gamma}^{1/2})(\mathbf{Q}\boldsymbol{\Gamma}^{1/2})^\mathsf{T}$$

  - and weight matrix $\mathbf{B}$

$$\mathbf{B}^* = (\mathbf{X}^\mathsf{T}\mathbf{X})^{-1/2}\mathbf{Q}\boldsymbol{\Gamma}^{1/2}$$

with $(\mathbf{X}^\mathsf{T}\mathbf{X})^\alpha = \mathbf{P}\boldsymbol{\Sigma}^\alpha\mathbf{P}^\mathsf{T}$ and σ_{ii}^α the power α of the nonzero eigenvalues of $\mathbf{X}^\mathsf{T}\mathbf{X}$ and $\sigma_{ii}^\alpha = 0$ otherwise.

- If $n-1 < p$ then $L_{\text{Strain}}(\mathbf{B}^*) = 0$ (perfect reconstruction of $\mathbf{K}$).
- If $n-1 > p$, then $L_{\text{Strain}}(\mathbf{B}^*) > 0$ (approximated reconstruction of $\mathbf{K}$)

5. Step 1 of Approximated KRR

Step 1 Linearly Approximated KRR: Minimize $L_{\text{Strain}}(\mathbf{B}) = \|\mathbf{K} - \mathbf{X}\mathbf{B}\mathbf{B}^\mathsf{T}\mathbf{X}^\mathsf{T}\|^2$

- Solution:
  - Computing the eigendecomposition

$$(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1/2}\mathbf{X}^\mathsf{T}\mathbf{K}\mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1/2} = \mathbf{Q}\boldsymbol{\Gamma}\mathbf{Q}^\mathsf{T} = (\mathbf{Q}\boldsymbol{\Gamma}^{1/2})(\mathbf{Q}\boldsymbol{\Gamma}^{1/2})^\mathsf{T}$$

  - and weight matrix $\mathbf{B}$

$$\mathbf{B}^* = (\mathbf{X}^\mathsf{T}\mathbf{X})^{-1/2}\mathbf{Q}\boldsymbol{\Gamma}^{1/2}$$

with $(\mathbf{X}^\mathsf{T}\mathbf{X})^\alpha = \mathbf{P}\boldsymbol{\Sigma}^\alpha\mathbf{P}^\mathsf{T}$ and σ_{ii}^α the power α of the nonzero eigenvalues of $\mathbf{X}^\mathsf{T}\mathbf{X}$ and $\sigma_{ii}^\alpha = 0$ otherwise.

- If $n - 1 < p$ then $L_{\text{Strain}}(\mathbf{B}^*) = 0$ (perfect reconstruction of $\mathbf{K}$).
- If $n - 1 > p$, then $L_{\text{Strain}}(\mathbf{B}^*) > 0$ (approximated reconstruction of $\mathbf{K}$)

5. Step 1 of Approximated KRR

Step 1 Linearly Approximated KRR: Minimize $L_{\text{Strain}}(\mathbf{B}) = \|\mathbf{K} - \mathbf{X}\mathbf{B}\mathbf{B}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\|^2$

- Solution:
  - Computing the eigendecomposition

$$(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-1/2}\mathbf{X}^{\mathsf{T}}\mathbf{K}\mathbf{X}(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-1/2} = \mathbf{Q}\boldsymbol{\Gamma}\mathbf{Q}^{\mathsf{T}} = (\mathbf{Q}\boldsymbol{\Gamma}^{1/2})(\mathbf{Q}\boldsymbol{\Gamma}^{1/2})^{\mathsf{T}}$$

  - and weight matrix $\mathbf{B}$

$$\mathbf{B}^* \quad = \quad (\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-1/2}\mathbf{Q}\boldsymbol{\Gamma}^{1/2}$$

  with $(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{\alpha} = \mathbf{P}\boldsymbol{\Sigma}^{\alpha}\mathbf{P}^{\mathsf{T}}$ and σ_{ii}^{α} the power α of the nonzero eigenvalues of $\mathbf{X}^{\mathsf{T}}\mathbf{X}$ and $\sigma_{ii}^{\alpha} = 0$ otherwise.

- If $n - 1 < p$ then $L_{\text{Strain}}(\mathbf{B}^*) = 0$ (perfect reconstruction of $\mathbf{K}$).
- If $n - 1 > p$, then $L_{\text{Strain}}(\mathbf{B}^*) > 0$ (approximated reconstruction of $\mathbf{K}$)

5. Step 2 of Approximated KRR

Step 2: We define Approximated KRR (AKRR):

- Do ridge regression with **XB** as predictor variables:

$$L_{\mathrm{AKRR}}(\boldsymbol{\beta}) \quad = \quad \|\mathbf{y} - \mathbf{XB}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|^2$$

- Let $\gamma = \mathbf{B}\beta$ so that

$$\begin{aligned} \mathbf{B}^\top\gamma &= \mathbf{B}^\top\mathbf{B}\beta \\ \beta &= \left(\mathbf{B}^\top\mathbf{B}\right)^{-1}\mathbf{B}^\top\gamma \\ \beta^\top\beta &= \gamma^\top\mathbf{B}\left(\mathbf{B}^\top\mathbf{B}\right)^{-2}\mathbf{B}^\top\gamma = \gamma^\top\left(\mathbf{B}^\top\mathbf{B}\right)^{-1}\gamma. \end{aligned}$$

- Then

$$\begin{aligned} L_{\mathrm{AKRR}}(\beta) &= L_{\mathrm{AKRR}}(\gamma) \\ &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^\top\left(\mathbf{B}^\top\mathbf{B}\right)^{-1}\gamma \\ &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^\top(\mathbf{X}^\top\mathbf{X})\left(\mathbf{X}^\top\mathbf{K}\mathbf{X}\right)^{-1}(\mathbf{X}^\top\mathbf{X})\gamma \\ &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^\top\mathbf{A}\gamma. \end{aligned}$$

5. Step 2 of Approximated KRR

Step 2: We define Approximated KRR (AKRR):

- Do ridge regression with $\mathbf{XB}$ as predictor variables:

$$L_{\mathrm{AKRR}}(\boldsymbol{\beta}) = \|\mathbf{y} - \mathbf{XB}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|^2$$

- Let $\boldsymbol{\gamma} = \mathbf{B}\boldsymbol{\beta}$ so that

$$\begin{aligned} \mathbf{B}^\mathsf{T}\boldsymbol{\gamma} &= \mathbf{B}^\mathsf{T}\mathbf{B}\boldsymbol{\beta} \\ \boldsymbol{\beta} &= \left(\mathbf{B}^\mathsf{T}\mathbf{B}\right)^{-1}\mathbf{B}^\mathsf{T}\boldsymbol{\gamma} \\ \boldsymbol{\beta}^\mathsf{T}\boldsymbol{\beta} &= \boldsymbol{\gamma}^\mathsf{T}\mathbf{B}\left(\mathbf{B}^\mathsf{T}\mathbf{B}\right)^{-2}\mathbf{B}^\mathsf{T}\boldsymbol{\gamma} = \boldsymbol{\gamma}^\mathsf{T}\left(\mathbf{B}^\mathsf{T}\mathbf{B}\right)^{-1}\boldsymbol{\gamma}. \end{aligned}$$

- Then

$$\begin{aligned} L_{\mathrm{AKRR}}(\boldsymbol{\beta}) &= L_{\mathrm{AKRR}}(\boldsymbol{\gamma}) \\ &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\gamma}\|^2 + \lambda\boldsymbol{\gamma}^\mathsf{T}\left(\mathbf{B}^\mathsf{T}\mathbf{B}\right)^{-1}\boldsymbol{\gamma} \\ &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\gamma}\|^2 + \lambda\boldsymbol{\gamma}^\mathsf{T}(\mathbf{X}^\mathsf{T}\mathbf{X})\left(\mathbf{X}^\mathsf{T}\mathbf{KX}\right)^{-1}(\mathbf{X}^\mathsf{T}\mathbf{X})\boldsymbol{\gamma} \\ &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\gamma}\|^2 + \lambda\boldsymbol{\gamma}^\mathsf{T}\mathbf{A}\boldsymbol{\gamma}. \end{aligned}$$

5. Step 2 of Approximated KRR

Step 2: We define Approximated KRR (AKRR):

- Do ridge regression with **XB** as predictor variables:

$$L_{\mathrm{AKRR}}(\boldsymbol{\beta}) \quad = \quad \|\mathbf{y} - \mathbf{XB}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|^2$$

- Let $\boldsymbol{\gamma} = \mathbf{B}\boldsymbol{\beta}$ so that

$$\begin{aligned} \mathbf{B}^{\mathsf{T}}\boldsymbol{\gamma} &= \mathbf{B}^{\mathsf{T}}\mathbf{B}\boldsymbol{\beta} \\ \boldsymbol{\beta} &= \left(\mathbf{B}^{\mathsf{T}}\mathbf{B}\right)^{-1}\mathbf{B}^{\mathsf{T}}\boldsymbol{\gamma} \\ \boldsymbol{\beta}^{\mathsf{T}}\boldsymbol{\beta} &= \boldsymbol{\gamma}^{\mathsf{T}}\mathbf{B}\left(\mathbf{B}^{\mathsf{T}}\mathbf{B}\right)^{-2}\mathbf{B}^{\mathsf{T}}\boldsymbol{\gamma} = \boldsymbol{\gamma}^{\mathsf{T}}\left(\mathbf{B}^{\mathsf{T}}\mathbf{B}\right)^{-1}\boldsymbol{\gamma}. \end{aligned}$$

- Then

$$\begin{aligned} L_{\mathrm{AKRR}}(\boldsymbol{\beta}) &= L_{\mathrm{AKRR}}(\boldsymbol{\gamma}) \\ &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\gamma}\|^2 + \lambda\boldsymbol{\gamma}^{\mathsf{T}}\left(\mathbf{B}^{\mathsf{T}}\mathbf{B}\right)^{-1}\boldsymbol{\gamma} \\ &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\gamma}\|^2 + \lambda\boldsymbol{\gamma}^{\mathsf{T}}(\mathbf{X}^{\mathsf{T}}\mathbf{X})\left(\mathbf{X}^{\mathsf{T}}\mathbf{K}\mathbf{X}\right)^{-1}(\mathbf{X}^{\mathsf{T}}\mathbf{X})\boldsymbol{\gamma} \\ &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\gamma}\|^2 + \lambda\boldsymbol{\gamma}^{\mathsf{T}}\mathbf{A}\boldsymbol{\gamma}. \end{aligned}$$

5. Approximated KRR

Properties of Approximated KRR:

- Loss AKRR:

$$\begin{aligned} L_{\text{AKRR}}(\gamma) &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^{\mathsf{T}}\left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)\left(\mathbf{X}^{\mathsf{T}}\mathbf{K}\mathbf{X}\right)^{-1}\left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)\gamma \\ &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^{\mathsf{T}}\mathbf{A}\gamma \end{aligned}$$

- AKRR yields exactly the same predictions as KRR if $n - 1 < p$ and $\text{rank}(\mathbf{X}) = n - 1$ (no approximation).
- AKRR approximates KRR otherwise.
- Interpretation of AKKR in terms of weights γ as in (ridge) regression.
- t-tests for γ can be derived as in ridge regression.

5. Approximated KRR

Properties of Approximated KRR:

- Loss AKRR:

$$\begin{aligned} L_{\text{AKRR}}(\gamma) &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^{\mathsf{T}}(\mathbf{X}^{\mathsf{T}}\mathbf{X})\left(\mathbf{X}^{\mathsf{T}}\mathbf{K}\mathbf{X}\right)^{-1}(\mathbf{X}^{\mathsf{T}}\mathbf{X})\gamma \\ &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^{\mathsf{T}}\mathbf{A}\gamma \end{aligned}$$

- AKRR yields exactly the same predictions as KRR if $n - 1 < p$ and rank($\mathbf{X}$) $= n - 1$ (no approximation).
- AKRR approximates KRR otherwise.
- Interpretation of AKKR in terms of weights γ as in (ridge) regression.
- t-tests for γ can be derived as in ridge regression.

5. Approximated KRR

Properties of Approximated KRR:

- Loss AKRR:

$$\begin{aligned} L_{\text{AKRR}}(\gamma) &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^\mathsf{T}(\mathbf{X}^\mathsf{T}\mathbf{X})\left(\mathbf{X}^\mathsf{T}\mathbf{K}\mathbf{X}\right)^{-1}(\mathbf{X}^\mathsf{T}\mathbf{X})\gamma \\ &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^\mathsf{T}\mathbf{A}\gamma \end{aligned}$$

- AKRR yields exactly the same predictions as KRR if $n - 1 < p$ and rank($\mathbf{X}$) $= n - 1$ (no approximation).
- AKRR approximates KRR otherwise.
- Interpretation of AKKR in terms of weights γ as in (ridge) regression.
- t-tests for γ can be derived as in ridge regression.

5. Approximated KRR

Properties of Approximated KRR:

- Loss AKRR:

$$\begin{aligned} L_{\text{AKRR}}(\gamma) &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^{\mathsf{T}}(\mathbf{X}^{\mathsf{T}}\mathbf{X})\left(\mathbf{X}^{\mathsf{T}}\mathbf{K}\mathbf{X}\right)^{-1}(\mathbf{X}^{\mathsf{T}}\mathbf{X})\gamma \\ &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^{\mathsf{T}}\mathbf{A}\gamma \end{aligned}$$

- AKRR yields exactly the same predictions as KRR if $n - 1 < p$ and rank$(\mathbf{X}) = n - 1$ (no approximation).
- AKRR approximates KRR otherwise.
- Interpretation of AKKR in terms of weights γ as in (ridge) regression.
- t-tests for γ can be derived as in ridge regression.

5. Approximated KRR

Properties of Approximated KRR:

- Loss AKRR:

$$
\begin{aligned}
L_{\mathrm{AKRR}}(\gamma) &= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^{\mathsf{T}}(\mathbf{X}^{\mathsf{T}}\mathbf{X})\left(\mathbf{X}^{\mathsf{T}}\mathbf{K}\mathbf{X}\right)^{-1}(\mathbf{X}^{\mathsf{T}}\mathbf{X})\gamma \\
&= \|\mathbf{y} - \mathbf{X}\gamma\|^2 + \lambda\gamma^{\mathsf{T}}\mathbf{A}\gamma
\end{aligned}
$$

- AKRR yields exactly the same predictions as KRR if $n-1 < p$ and $\mathrm{rank}(\mathbf{X}) = n-1$ (no approximation).
- AKRR approximates KRR otherwise.
- Interpretation of AKKR in terms of weights γ as in (ridge) regression.
- t-tests for γ can be derived as in ridge regression.

5. Approximated KRR

Quality of approximation of penalty:

- Loss optimal approximation kernel penalty:

$$\begin{aligned} L_{\text{Strain}}(\mathbf{B}^*) &= \|\mathbf{K} - \mathbf{X}\mathbf{B}^*\mathbf{B}^{*\mathsf{T}}\mathbf{X}^\mathsf{T}\|^2 \\ &= \|\mathbf{K} - \mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T}\mathbf{K}\mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T}\|^2 \\ &= \mathrm{tr}(\mathbf{I} - \mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T})\mathbf{K}(\mathbf{I} - \mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T}) \end{aligned}$$

- This loss is equal to the part of $\mathbf{K}$ that is not in the space of $\mathbf{X}$.
- Penalty accounted for (PAF) is the proportion of $\|\mathbf{K}\|^2$ in the space of $\mathbf{X}$:

$$\text{PAF} = \frac{\mathrm{tr}(\mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T})\mathbf{K}(\mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T})}{\|\mathbf{K}\|^2}$$

5. Approximated KRR

Quality of approximation of penalty:

- Loss optimal approximation kernel penalty:

$$\begin{aligned} L_{\text{Strain}}(\mathbf{B}^*) &= \|\mathbf{K} - \mathbf{X}\mathbf{B}^*\mathbf{B}^{*\mathsf{T}}\mathbf{X}^\mathsf{T}\|^2 \\ &= \|\mathbf{K} - \mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T}\mathbf{K}\mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T}\|^2 \\ &= \mathrm{tr}(\mathbf{I} - \mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T})\mathbf{K}(\mathbf{I} - \mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T}) \end{aligned}$$

- This loss is equal to the part of **K** that is not in the space of **X**.
- Penalty accounted for (PAF) is the proportion of $\|\mathbf{K}\|^2$ in the space of **X**:

$$\text{PAF} = \frac{\mathrm{tr}(\mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T})\mathbf{K}(\mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^-\mathbf{X}^\mathsf{T})}{\|\mathbf{K}\|^2}$$

5. Approximated KRR

Quality of approximation of penalty:

- Loss optimal approximation kernel penalty:

$$\begin{aligned} L_{\text{Strain}}(\mathbf{B}^*) &= \|\mathbf{K} - \mathbf{X}\mathbf{B}^*{\mathbf{B}^*}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\|^2 \\ &= \|\mathbf{K} - \mathbf{X}(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-}\mathbf{X}^{\mathsf{T}}\mathbf{K}\mathbf{X}(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-}\mathbf{X}^{\mathsf{T}}\|^2 \\ &= \mathrm{tr}(\mathbf{I} - \mathbf{X}(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-}\mathbf{X}^{\mathsf{T}})\mathbf{K}(\mathbf{I} - \mathbf{X}(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-}\mathbf{X}^{\mathsf{T}}) \end{aligned}$$

- This loss is equal to the part of **K** that is not in the space of **X**.
- Penalty accounted for (PAF) is the proportion of $\|\mathbf{K}\|^2$ in the space of **X**:

$$\mathrm{PAF} = \frac{\mathrm{tr}(\mathbf{X}(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-}\mathbf{X}^{\mathsf{T}})\mathbf{K}(\mathbf{X}(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{-}\mathbf{X}^{\mathsf{T}})}{\|\mathbf{K}\|^2}$$

Table of Contents

1. Introduction
2. Linear Kernels
3. Nonlinear KRR
4. Interpretable Kernels
5. Approximated KRR
6. Conclusions

Conclusions

- Nonlinearity in orginal variables can be done by nonlinear kernels.
- Any kernel method can be interpreted as a linear combination in original predictors with a quadratic ridge penalty in specific metric of the weights.
- Approximation is exact if $p > n$ and has PAF $= 1$.
- Can be used for SVM, Kernel Ridge Regression, Kernel Logistic Regression, etc.
- Could be used in any software that allows ridge weighted quadratic penalties such as `glmnet` (with some pre- and post-processing by linear algebra).

Conclusions

- Nonlinearity in orginal variables can be done by nonlinear kernels.
- Any kernel method can be interpreted as a linear combination in original predictors with a quadratic ridge penalty in specific metric of the weights.
- Approximation is exact if $p > n$ and has PAF $= 1$.
- Can be used for SVM, Kernel Ridge Regression, Kernel Logistic Regression, etc.
- Could be used in any software that allows ridge weighted quadratic penalties such as `glmnet` (with some pre- and post-processing by linear algebra).

Conclusions

- Nonlinearity in orginal variables can be done by nonlinear kernels.
- Any kernel method can be interpreted as a linear combination in original predictors with a quadratic ridge penalty in specific metric of the weights.
- Approximation is exact if $p > n$ and has PAF $= 1$.
- Can be used for SVM, Kernel Ridge Regression, Kernel Logistic Regression, etc.
- Could be used in any software that allows ridge weighted quadratic penalties such as `glmnet` (with some pre- and post-processing by linear algebra).

Conclusions

- Nonlinearity in orginal variables can be done by nonlinear kernels.
- Any kernel method can be interpreted as a linear combination in original predictors with a quadratic ridge penalty in specific metric of the weights.
- Approximation is exact if $p > n$ and has PAF $= 1$.
- Can be used for SVM, Kernel Ridge Regression, Kernel Logistic Regression, etc.
- Could be used in any software that allows ridge weighted quadratic penalties such as `glmnet` (with some pre- and post-processing by linear algebra).

Conclusions

- Nonlinearity in orginal variables can be done by nonlinear kernels.
- Any kernel method can be interpreted as a linear combination in original predictors with a quadratic ridge penalty in specific metric of the weights.
- Approximation is exact if $p > n$ and has PAF $= 1$.
- Can be used for SVM, Kernel Ridge Regression, Kernel Logistic Regression, etc.
- Could be used in any software that allows ridge weighted quadratic penalties such as `glmnet` (with some pre- and post-processing by linear algebra).

A.1 Proof that $\lambda\mathbf{w}^\top\mathbf{w} = \lambda\tilde{\mathbf{q}}^\top(\mathbf{X}\mathbf{X}^\top)^{-1}\tilde{\mathbf{q}}$

Proof that $\lambda\mathbf{w}^\top\mathbf{w} = \lambda\tilde{\mathbf{q}}^\top(\mathbf{X}\mathbf{X}^\top)^{-1}\tilde{\mathbf{q}}$:

- Let the SVD of $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$. Then

$$\begin{aligned}\tilde{\mathbf{q}} &= \mathbf{X}\mathbf{w} = \mathbf{U}\mathbf{D}\mathbf{V}^\top\mathbf{w}\\ \mathbf{U}^\top\tilde{\mathbf{q}} &= \mathbf{D}\mathbf{V}^\top\mathbf{w}\\ \mathbf{D}^{-1}\mathbf{U}^\top\tilde{\mathbf{q}} &= \mathbf{V}^\top\mathbf{w}\end{aligned}$$

- The penalty term can be written as

$$\begin{aligned}\lambda\mathbf{w}^\top\mathbf{w} &= \lambda\mathbf{w}^\top(\mathbf{I} - \mathbf{V}\mathbf{V}^\top + \mathbf{V}\mathbf{V}^\top)\mathbf{w}\\ &= \lambda\mathbf{w}^\top(\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathbf{w} + \lambda\mathbf{w}^\top\mathbf{V}\mathbf{V}^\top\mathbf{w}\end{aligned}$$

  The part of $\mathbf{w}$ in the space of $\mathbf{X}$ is $\mathbf{w}_1 = \mathbf{V}\mathbf{V}^\top\mathbf{w}$ and the part outside is $\mathbf{w}_2 = (\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathbf{w}$
- Thus, $\mathbf{w}^\top(\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathbf{w} = \mathbf{w}_2^\top\mathbf{w}_2 = 0$ and

$$\begin{aligned}\lambda\mathbf{w}^\top\mathbf{w} &= \lambda\mathbf{w}^\top\mathbf{V}\mathbf{V}^\top\mathbf{w} = \lambda\mathbf{w}_1^\top\mathbf{w}_1\\ &= \lambda\tilde{\mathbf{q}}\mathbf{U}\mathbf{D}^{-2}\mathbf{U}^\top\tilde{\mathbf{q}} = \lambda\tilde{\mathbf{q}}\mathbf{U}\mathbf{D}^{-1}\mathbf{V}^\top\mathbf{V}\mathbf{D}^{-1}\mathbf{U}^\top\tilde{\mathbf{q}} = \lambda\tilde{\mathbf{q}}(\mathbf{X}\mathbf{X}^\top)^{-1}\tilde{\mathbf{q}}\end{aligned}$$

Back

A.1 Proof that $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \tilde{\mathbf{q}}^\top (\mathbf{X}\mathbf{X}^\top)^{-1} \tilde{\mathbf{q}}$

Proof that $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \tilde{\mathbf{q}}^\top (\mathbf{X}\mathbf{X}^\top)^{-1} \tilde{\mathbf{q}}$:

- Let the SVD of $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$. Then

$$\begin{aligned}
\tilde{\mathbf{q}} &= \mathbf{X}\mathbf{w} = \mathbf{U}\mathbf{D}\mathbf{V}^\top \mathbf{w} \\
\mathbf{U}^\top \tilde{\mathbf{q}} &= \mathbf{D}\mathbf{V}^\top \mathbf{w} \\
\mathbf{D}^{-1}\mathbf{U}^\top \tilde{\mathbf{q}} &= \mathbf{V}^\top \mathbf{w}
\end{aligned}$$

- The penalty term can be written as

$$\begin{aligned}
\lambda \mathbf{w}^\top \mathbf{w} &= \lambda \mathbf{w}^\top (\mathbf{I} - \mathbf{V}\mathbf{V}^\top + \mathbf{V}\mathbf{V}^\top)\mathbf{w} \\
&= \lambda \mathbf{w}^\top (\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathbf{w} + \lambda \mathbf{w}^\top \mathbf{V}\mathbf{V}^\top \mathbf{w}
\end{aligned}$$

  The part of $\mathbf{w}$ in the space of $\mathbf{X}$ is $\mathbf{w}_1 = \mathbf{V}\mathbf{V}^\top \mathbf{w}$ and the part outside is $\mathbf{w}_2 = (\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathbf{w}$

- Thus, $\mathbf{w}^\top (\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathbf{w} = \mathbf{w}_2^\top \mathbf{w}_2 = 0$ and

$$\begin{aligned}
\lambda \mathbf{w}^\top \mathbf{w} &= \lambda \mathbf{w}^\top \mathbf{V}\mathbf{V}^\top \mathbf{w} = \lambda \mathbf{w}_1^\top \mathbf{w}_1 \\
&= \lambda \tilde{\mathbf{q}}\mathbf{U}\mathbf{D}^{-2}\mathbf{U}^\top \tilde{\mathbf{q}} = \lambda \tilde{\mathbf{q}}\mathbf{U}\mathbf{D}^{-1}\mathbf{V}^\top \mathbf{V}\mathbf{D}^{-1}\mathbf{U}^\top \tilde{\mathbf{q}} = \lambda \tilde{\mathbf{q}}(\mathbf{X}\mathbf{X}^\top)^{-1}\tilde{\mathbf{q}}
\end{aligned}$$

Back

A.1 Proof that $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \tilde{\mathbf{q}}^\top (\mathbf{X}\mathbf{X}^\top)^{-1} \tilde{\mathbf{q}}$

Proof that $\lambda \mathbf{w}^\top \mathbf{w} = \lambda \tilde{\mathbf{q}}^\top (\mathbf{X}\mathbf{X}^\top)^{-1} \tilde{\mathbf{q}}$:

- Let the SVD of $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$. Then

$$\begin{aligned} \tilde{\mathbf{q}} &= \mathbf{X}\mathbf{w} = \mathbf{U}\mathbf{D}\mathbf{V}^\top \mathbf{w} \\ \mathbf{U}^\top \tilde{\mathbf{q}} &= \mathbf{D}\mathbf{V}^\top \mathbf{w} \\ \mathbf{D}^{-1}\mathbf{U}^\top \tilde{\mathbf{q}} &= \mathbf{V}^\top \mathbf{w} \end{aligned}$$

- The penalty term can be written as

$$\begin{aligned} \lambda \mathbf{w}^\top \mathbf{w} &= \lambda \mathbf{w}^\top (\mathbf{I} - \mathbf{V}\mathbf{V}^\top + \mathbf{V}\mathbf{V}^\top)\mathbf{w} \\ &= \lambda \mathbf{w}^\top (\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathbf{w} + \lambda \mathbf{w}^\top \mathbf{V}\mathbf{V}^\top \mathbf{w} \end{aligned}$$

  The part of $\mathbf{w}$ in the space of $\mathbf{X}$ is $\mathbf{w}_1 = \mathbf{V}\mathbf{V}^\top \mathbf{w}$ and the part outside is $\mathbf{w}_2 = (\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathbf{w}$

- Thus, $\mathbf{w}^\top (\mathbf{I} - \mathbf{V}\mathbf{V}^\top)\mathbf{w} = \mathbf{w}_2^\top \mathbf{w}_2 = 0$ and

$$\begin{aligned} \lambda \mathbf{w}^\top \mathbf{w} &= \lambda \mathbf{w}^\top \mathbf{V}\mathbf{V}^\top \mathbf{w} = \lambda \mathbf{w}_1^\top \mathbf{w}_1 \\ &= \lambda \tilde{\mathbf{q}} \mathbf{U}\mathbf{D}^{-2}\mathbf{U}^\top \tilde{\mathbf{q}} = \lambda \tilde{\mathbf{q}} \mathbf{U}\mathbf{D}^{-1}\mathbf{V}^\top \mathbf{V}\mathbf{D}^{-1}\mathbf{U}^\top \tilde{\mathbf{q}} = \lambda \tilde{\mathbf{q}} (\mathbf{X}\mathbf{X}^\top)^{-1} \tilde{\mathbf{q}} \end{aligned}$$

Back

A.2 Prediction for Nonlinear KRR

Final step needed with kernels for predicting the test data $\mathbf{X}_u$:

- Let the SVD of $\boldsymbol{\Phi} = \mathbf{UDV}^\top$. Then $\boldsymbol{\Phi}^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top)^{-1}\boldsymbol{\Phi} = \mathbf{VV}^\top$ because

$$\begin{aligned}
\boldsymbol{\Phi}^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top)^{-1}\boldsymbol{\Phi} &= \mathbf{VDU}^\top(\mathbf{UDV}^\top\mathbf{VDU}^\top)^{-1}\mathbf{UDV}^\top \\
&= \mathbf{VDU}^\top\mathbf{UD}^{-2}\mathbf{U}^\top\mathbf{UDV}^\top \\
&= \mathbf{VDD}^{-2}\mathbf{DV}^\top = \mathbf{VV}^\top
\end{aligned}$$

- Then the predicted $\mathbf{q}_u$ for the test set $\mathbf{X}_u$ is

$$\begin{aligned}
\mathbf{q}_u = w_0\mathbf{1} + \boldsymbol{\Phi}_u\mathbf{w} &= w_0\mathbf{1} + \boldsymbol{\Phi}_u\mathbf{VV}^\top\mathbf{w} \\
&= w_0\mathbf{1} + \boldsymbol{\Phi}_u\boldsymbol{\Phi}^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top)^{-1}\boldsymbol{\Phi}\mathbf{w} \\
&= w_0\mathbf{1} + (\boldsymbol{\Phi}_u\boldsymbol{\Phi}^\top)(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top)^{-1}(\boldsymbol{\Phi}\mathbf{w}) \\
&= w_0\mathbf{1} + \mathbf{K}_u\mathbf{K}^{-1}\tilde{\mathbf{q}}
\end{aligned}$$

with $\mathbf{K}_u$ is the $n_u \times n$ kernel matrix with elements k_{ij} where i stands for row i of $\mathbf{X}_u$ and j for row j of $\mathbf{X}$. Back

A.2 Prediction for Nonlinear KRR

Final step needed with kernels for predicting the test data $\mathbf{X}_u$:

- Let the SVD of $\mathbf{\Phi} = \mathbf{UDV}^\top$. Then $\mathbf{\Phi}^\top(\mathbf{\Phi\Phi}^\top)^{-1}\mathbf{\Phi} = \mathbf{VV}^\top$ because

$$\begin{aligned}\mathbf{\Phi}^\top(\mathbf{\Phi\Phi}^\top)^{-1}\mathbf{\Phi} &= \mathbf{VDU}^\top(\mathbf{UDV}^\top\mathbf{VDU}^\top)^{-1}\mathbf{UDV}^\top \\ &= \mathbf{VDU}^\top\mathbf{UD}^{-2}\mathbf{U}^\top\mathbf{UDV}^\top \\ &= \mathbf{VDD}^{-2}\mathbf{DV}^\top = \mathbf{VV}^\top\end{aligned}$$

- Then the predicted $\mathbf{q}_u$ for the test set $\mathbf{X}_u$ is

$$\begin{aligned}\mathbf{q}_u = w_0\mathbf{1} + \mathbf{\Phi}_u\mathbf{w} &= w_0\mathbf{1} + \mathbf{\Phi}_u\mathbf{VV}^\top\mathbf{w} \\ &= w_0\mathbf{1} + \mathbf{\Phi}_u\mathbf{\Phi}^\top(\mathbf{\Phi\Phi}^\top)^{-1}\mathbf{\Phi w} \\ &= w_0\mathbf{1} + (\mathbf{\Phi}_u\mathbf{\Phi}^\top)(\mathbf{\Phi\Phi}^\top)^{-1}(\mathbf{\Phi w}) \\ &= w_0\mathbf{1} + \mathbf{K}_u\mathbf{K}^{-1}\tilde{\mathbf{q}}\end{aligned}$$

  with $\mathbf{K}_u$ is the $n_u \times n$ kernel matrix with elements k_{ij} where i stands for row i of $\mathbf{X}_u$ and j for row j of $\mathbf{X}$. Back