



**Universitat
Pompeu Fabra**
Barcelona

Department
of Economics and Business

Economics Working Paper Series

Working Paper No. 1915

Interpretable Kernels

Patrick J. F. Groenen and Michael J. Greenacre

September 2025

Interpretable Kernels

Patrick J.F. Groenen¹ and Michael Greenacre²

¹Econometric Institute, Erasmus University Rotterdam, The Netherlands

²Universitat Pompeu Fabra, Barcelona, Spain

September 4, 2025

Abstract

The use of kernels for nonlinear prediction is widespread in machine learning. They have been popularized in support vector machines and used in kernel ridge regression, amongst others. Kernel methods share three aspects. First, instead of the original matrix of predictor variables or features, each observation is mapped into an enlarged feature space. Second, a ridge penalty term is used to shrink the coefficients on the features in the enlarged feature space. Third, the solution is not obtained in this enlarged feature space, but through solving a dual problem in the observation space. A major drawback in the present use of kernels is that the interpretation in terms of the original features is lost. In this paper, we argue that in the case of a wide matrix of features, where there are more features than observations, the kernel solution can be re-expressed in terms of a linear combination of the original matrix of features and a ridge penalty that involves a special metric. Consequently, the exact same predicted values can be obtained as a weighted linear combination of the features in the usual manner and thus can be interpreted. In the case where the number of features is less than the number of observations, we discuss a least-squares approximation of the kernel matrix that still allows the interpretation in terms of a linear combination. It is shown that these results hold for any function of a linear combination that minimizes the coefficients and has a ridge penalty on these coefficients, such as in kernel logistic regression and kernel Poisson regression. This work makes a contribution to interpretable artificial intelligence.

1 Introduction

The last two decades have shown a dramatic improvement of prediction accuracy by artificial intelligence methods through machine learning. The availability of big data combined with highly nonlinear models has led to improved prediction capabilities. However, in many applications, such nonlinear predictions cannot

be used due to regulatory requirements. For example, a bank cannot use directly the results from a deep neural net “black box” algorithm for credit scoring because the regulator requires the bank to be able to explain their credit scoring rules. Therefore, explainable artificial intelligence (abbreviated as XAI) has received much attention. Ali et al. [2023] and Dwivedi et al. [2023] give overviews of XAI, classifying the area into different types, where the present contribution would be included in the category of “model explainability”. Furthermore, our approach falls in what Rudin [2019] calls models that are “inherently interpretable” rather than explaining black box models.

This paper focuses on kernel methods that are used for nonlinear predictions, where our main contribution lies in re-expressing the kernel penalty term such that a linear interpretation in the features becomes possible. The use of kernels has become a popular and attractive tool to allow for nonlinear predictions. It has gained popularity in machine learning through the method of support vector machines and can be applied in many other techniques that use linear prediction. We refer to Hastie et al. [2008], Section 5.8, who give a simplified introduction to kernel methods based on the explanation in Wahba [1990]. We provide below a linear algebra interpretation.

The main idea is to map each row \mathbf{x}_i^\top of the $n \times p$ matrix \mathbf{X} of features into the vector ϕ_i^\top of an enlarged feature space represented by the $n \times r$ matrix Φ , where $r \gg p$. A well-known example is the so-called second degree nonhomogeneous polynomial kernel that is equivalent to choosing ϕ_i^\top as the vector of all main and two-variable interaction effects, which is a special case of the features in polynomial regression. Then, the linear prediction is done in the enlarged feature space of the ϕ_i^\top by $\phi_i^\top \beta$, which is equivalent to a nonlinear prediction in the original space of the \mathbf{x}_i^\top . As the dimensionality r of the space spanned by ϕ_i^\top increases towards or beyond the number of observations (rows) n , then some form of shrinkage is needed to avoid overfitting.

For very large r , the estimation of explicit coefficients β of the linear combination $\eta = \Phi\beta$ becomes infeasible. This situation is one where the so-called kernel trick can be used: that is, instead of computing the solution for β , the optimization can be done directly over the predictions in η , making use of the $n \times n$ kernel matrix $\mathbf{K} = \Phi\Phi^\top$. Mappings from \mathbf{X} to Φ that satisfy the so-called Mercer conditions [Mercer, 1909, Campbell, 2002] have very simple expressions for $k_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ as a function of \mathbf{x}_i and \mathbf{x}_j only (which we will also denote by k_{ij} to simplify notation). For example, the radial basis function (RBF) kernel (or Gaussian kernel) has $k_{ij} = \exp(-s\|\mathbf{x}_i - \mathbf{x}_j\|^2)$ (with $s > 0$ a scaling hyper-parameter) and the nonhomogeneous polynomial kernel of degree d has $k_{ij} = (1 + \mathbf{x}_i^\top \mathbf{x}_j)^d$.

One of the main disadvantages of the use of kernels is that only the resulting predictions $\eta = \Phi\beta$ are found, without knowing β . Thus, for many kernels no model interpretation is given in terms of the derived features in Φ or in the original features in \mathbf{X} . This paper offers a general solution to this problem. Here, it is shown that if $p \geq n$ the kernel solution can also be expressed in terms of the original features, that is, $\eta = \Phi\beta = \mathbf{X}\gamma$, so that the coefficients in γ can be interpreted similarly as in regression. In this case, both linear combinations

and both kernel penalties are exactly the same. For $p < n$, we propose to use a least-squares approximation of the kernel penalty that preserves the prediction by $\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\gamma}$ and thus the linear interpretation through the coefficients in $\boldsymbol{\gamma}$. A simple diagnostic is proposed, the kernel accounted for (KAF), that gives the proportion of the kernel penalty that is accounted for by the approximation.

Next to the explainability of the kernel, a second important conclusion of this paper is that the nonlinear prediction through kernels critically depends on the kernel penalty and is equivalent to a linear prediction with a quadratic penalty on the coefficients, with contributions to the penalty not only on the size of the individual coefficients (as for the ridge penalty) but also on their combinations.

A third contribution is the realization that the linear interpretation proposed in this paper can be applied to any prediction model that uses a kernel in combination with a quadratic kernel penalty. Examples are support vector machines, kernel ridge regression, kernel logistic regression, kernel Poisson regression, amongst others. With this generality, we contribute to explainable AI.

The present paper can be seen as a contribution following the call of Crawford et al. [2018] who provided general approximation for kernels and what they call effect size analogs through projections in Bayesian Approximate Kernel Regression. They write: “It should be clear that a variety of projection procedures can be specified corresponding to various priors and loss functions, and a systematic study elucidating which projections are efficient and robust is of great interest.” We follow a linear algebra derivation leading to coefficients that are exactly the same in cases of a wide matrix of predictors, and approximate otherwise.

Section 2 starts with the general theory of a kernel penalty and its approximation, showing how coefficients for the original variables can be obtained. Section 3 deals with computational aspects, where it is shown how solutions to the kernel approach can be obtained using standard software. Section 4 gives two applications, one in a linear regression context and the other in logistic regression. Section 5 concludes with a summary and discussion.

2 Approximate kernel ridge regression and estimation of interpretable regression coefficients

2.1 Kernel ridge regression and the dual problem

In general, kernel methods can be written as the minimization of the sum of a loss function, $f(\boldsymbol{\eta})$, where $\boldsymbol{\eta} = \boldsymbol{\Phi}\boldsymbol{\beta}$, and a ridge penalty on the coefficients $\boldsymbol{\beta}$ ($r \times 1$), that is,

$$\min_{\boldsymbol{\beta}} f(\boldsymbol{\eta}) + \lambda \|\boldsymbol{\beta}\|^2, \quad (1)$$

where $\|\beta\|^2 = \beta^\top \beta$ and $\lambda > 0$ is a fixed penalty strength parameter that can be determined through k -fold cross-validation. The simplest example is ridge regression with $f(\eta) = \|\mathbf{y} - \eta\|^2$, where \mathbf{y} is an observed $n \times 1$ vector of a response variable, Φ is chosen as the observed predictor matrix \mathbf{X} , and so $\eta = \Phi\beta$. This scheme includes generalized linear models where the function f is the deviance and there is a link function $g(\eta)$ between the linear model and the response mean.

Our approach starts by passing from the original minimization problem in terms of coefficients β in the linear combinations $\eta = \Phi\beta$ of the high-dimensional predictors, to the dual problem of estimating η itself where the penalty is imposed on η .

Hence, $\|\beta\|^2$ is rewritten as a quadratic function of η , by solving for β as follows:

$$\begin{aligned}\eta &= \Phi\beta \\ \Phi^\top \eta &= \Phi^\top \Phi\beta \\ (\Phi^\top \Phi)^- \Phi^\top \eta &= \beta,\end{aligned}\tag{2}$$

where $()^-$ refers to the regular matrix inverse, if it exists, otherwise the Moore-Penrose generalized inverse. Then,

$$\begin{aligned}\|\beta\|^2 &= \beta^\top \beta \\ &= \eta^\top \Phi (\Phi^\top \Phi)^- (\Phi^\top \Phi)^- \Phi^\top \eta \\ &= \eta^\top (\Phi \Phi^\top)^- \eta \\ &= \eta^\top \mathbf{K}^- \eta.\end{aligned}\tag{3}$$

Notice that the step from the second to the third lines in (3) above can be deduced using the singular value decomposition (SVD)¹ of Φ . The dual problem is thus, in terms of η , the minimization of

$$f(\eta) + \lambda \eta^\top \mathbf{K}^- \eta\tag{4}$$

over η .

We now show how to obtain estimates of interpretable coefficients γ ($p \times 1$) in linear combinations $\mathbf{X}\gamma$, of the original predictors, which are equal to or approximate those in $\Phi\beta$, depending on whether $p \geq n$ or $p < n$ respectively. This is possible thanks to approximating the kernel, and the estimation will be as successful as the closeness of the approximation.

2.2 Approximating the kernel

The approximation to the kernel matrix \mathbf{K} amounts to a double projection onto the rows and columns of \mathbf{X} . There are two slightly different outcomes depending

¹SVD: $\Phi = \mathbf{U}\Sigma\mathbf{V}^\top$, $\mathbf{U}(n \times k)$, $\mathbf{V}(r \times k)$, $\Sigma(k \times k)$, where Φ is of rank k , $\mathbf{U}^\top \mathbf{U} = \mathbf{V}^\top \mathbf{V} = \mathbf{I}$ and Σ is a diagonal matrix of positive singular values. Then $(\Phi^\top \Phi)^- = (\mathbf{V}\Sigma\mathbf{U}^\top\mathbf{U}\Sigma\mathbf{V}^\top)^- = (\mathbf{V}\Sigma^2\mathbf{V}^\top)^- = \mathbf{V}\Sigma^{-2}\mathbf{V}^\top$, where Σ^{-2} denotes a diagonal matrix of the inverses of the squares of the singular values. Similarly, $(\Phi\Phi^\top)^- = \mathbf{U}\Sigma^{-2}\mathbf{U}^\top$. The result $\Phi(\Phi^\top \Phi)^- (\Phi^\top \Phi)^- \Phi^\top = (\Phi\Phi^\top)^-$ follows.

on whether $p < n$ or $p \geq n$. Let us suppose that Φ can be approximated by \mathbf{XB} , where \mathbf{B} is a matrix of regression coefficients. Since $\mathbf{K} = \Phi\Phi^\top$ is then approximated by $\mathbf{XBB}^\top\mathbf{X}^\top$, we are rather concerned with estimating \mathbf{BB}^\top , which can be denoted by the $p \times p$ matrix $\mathbf{A} = \mathbf{BB}^\top$. The least-squares approximation of the kernel matrix \mathbf{K} by \mathbf{XAX}^\top implies the following minimization objective:

$$\min_{\mathbf{A}} \|\mathbf{K} - \mathbf{XAX}^\top\|^2. \quad (5)$$

To minimize (5) we take its partial derivatives with respect to \mathbf{A} and equate that to zero, in order to obtain an estimate $\hat{\mathbf{A}}$ of \mathbf{A} , that is,

$$2\mathbf{X}^\top\mathbf{X}\hat{\mathbf{A}}\mathbf{X}^\top\mathbf{X} - 2\mathbf{X}^\top\mathbf{K}\mathbf{X} = \mathbf{0}$$

and solve for $\hat{\mathbf{A}}$

$$\hat{\mathbf{A}} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{K}\mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1} \quad (6)$$

so that

$$\hat{\mathbf{K}} = \mathbf{X}\hat{\mathbf{A}}\mathbf{X}^\top = \mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{K}\mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top$$

is the approximate kernel matrix. Note that $\mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top$ is a projector matrix so that $\hat{\mathbf{K}}$ can be interpreted as the projection of the row and column spaces of \mathbf{K} onto the space spanned by \mathbf{X} .

The amount of loss incurred by the penalty approximation can be obtained by inserting $\hat{\mathbf{A}}$ from (6) into (5), that is,

$$\begin{aligned} \|\mathbf{K} - \hat{\mathbf{K}}\|^2 &= \|\mathbf{K} - \mathbf{X}\hat{\mathbf{A}}\mathbf{X}^\top\|^2 \\ &= \|\mathbf{K} - \mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{K}\mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\|^2 \\ &= \|\mathbf{K} - \mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{K}(\mathbf{I} - \mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top)\|^2 \end{aligned} \quad (7)$$

from which it can be seen that the loss is equal to the part of \mathbf{K} that is not in the space of \mathbf{X} .

From (7), we define the *kernel accounted for* (KAF) as the proportion of $\|\mathbf{K}\|^2$ in the space of \mathbf{X} :

$$\text{KAF} = \frac{\|\hat{\mathbf{K}}\|^2}{\|\mathbf{K}\|^2}. \quad (8)$$

Note that if $p \geq n$ and \mathbf{X} is of full rank then, from (6) and again using the SVD, the approximation $\hat{\mathbf{K}}$ is perfect and the objective criterion in (5) is zero and $\text{KAF} = 1$; this is the exact case. If $p < n$, then $\text{KAF} < 1$; this is the approximate case.

2.3 Estimating the interpretable coefficients

Finally, we arrive at estimating p interpretable coefficients on the original p predictors. The results are different depending on whether $p \geq n$ or $p < n$.

In the former exact case, Φ can be written as \mathbf{XB} , where \mathbf{B} is of full rank n . Thus, the linear combination $\eta = \Phi\beta$ is equal to $\mathbf{XB}\beta$, which can be written as $\mathbf{X}\gamma$, with coefficients on the original features (columns of \mathbf{X}) equal to $\gamma = \mathbf{B}\beta$. Solving for β , $\beta = (\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{B}^\top \gamma$ (as obtained in (2)), and the penalty in (4) in this exact case can be expressed in terms of β as

$$\beta^\top \beta = \gamma^\top \mathbf{B}(\mathbf{B}^\top \mathbf{B})^{-1} (\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{B}^\top \gamma = \gamma^\top (\mathbf{B}\mathbf{B}^\top)^{-1} \gamma = \gamma^\top \hat{\mathbf{A}}^{-1} \gamma, \quad (9)$$

(see the simplification using the SVD in Footnote 1).

In the approximate case when $p < n$, Φ is only approximated by \mathbf{XB} , and the result in (9) reduces to the approximate penalty $\beta^\top \beta \approx \gamma^\top \hat{\mathbf{A}}^{-1} \gamma$. In the exact case, we have the equality of the kernel solution η , which is numerically the same as the solution of $\mathbf{X}\gamma$ with penalty $\lambda \gamma^\top \hat{\mathbf{A}}^{-1} \gamma$, that is,

$$f(\eta) + \lambda \eta^\top \mathbf{K}^{-1} \eta = f(\mathbf{X}\gamma) + \lambda \gamma^\top \hat{\mathbf{A}}^{-1} \gamma. \quad (10)$$

Therefore, in this exact case, it is possible to obtain the same kernel solution by a traditional linear combination where the coefficients γ can be interpreted as in traditional regression techniques. In the approximate case, the kernel solution is approximate, with its quality measured by the KAF.

2.4 Dealing with the regression intercept

The derivations above assume that if there is an intercept, then the intercept is modelled through a column of ones in \mathbf{X} and the intercept itself is penalized in the ridge penalty. In many cases, one may want to avoid the regularization of the intercept, or simply not estimate the intercept at all. As many kernels implicitly estimate the intercept, some extra steps are required to avoid estimation of the intercept in the linear combination. Let $\mathbf{J} = \mathbf{I} - n^{-1} \mathbf{1}\mathbf{1}^\top$ be the $n \times n$ centring matrix. Then, $\mathbf{J}\mathbf{X}$ is the column centred version of \mathbf{X} and, similarly, $\mathbf{J}\Phi$ is the column centred high-dimensional space with $\mathbf{J}\Phi\Phi^\top \mathbf{J} = \mathbf{J}\mathbf{K}\mathbf{J} = \mathbf{K}_c$, the double centred version of \mathbf{K} . We use the subscript c to denote the (double) centred versions of a matrix or vector. The double centred approximation \mathbf{K}_c through the minimization of $\|\mathbf{K}_c - \mathbf{X}_c \mathbf{A} \mathbf{X}_c^\top\|^2$ over \mathbf{A} with $\mathbf{X}_c = \mathbf{J}\mathbf{X}$ is achieved by

$$\hat{\mathbf{A}}_c = (\mathbf{X}_c^\top \mathbf{X}_c)^{-1} \mathbf{X}_c^\top \mathbf{K}_c \mathbf{X}_c (\mathbf{X}_c^\top \mathbf{X}_c)^{-1}. \quad (11)$$

The equivalent of (10) with an unpenalized intercept α and centred $\eta_c = \mathbf{J}\eta$ then becomes

$$f(\alpha \mathbf{1} + \eta_c) + \lambda \eta_c^\top \mathbf{K}_c^{-1} \eta_c = f(\alpha \mathbf{1} + \mathbf{X}_c \gamma) + \lambda \gamma^\top \hat{\mathbf{A}}_c^{-1} \gamma. \quad (12)$$

2.5 Re-expressing the kernel penalty as a regular ridge penalty

As for the exact case, it is also possible to re-express the kernel approach as a linear combination $\mathbf{Z}\delta$ with ridge penalty term $\lambda \delta^\top \delta$. The advantage of this

re-expression is that the kernel solution results can be computed using standard software (e.g., the `glmnet` package [Friedman et al., 2010, Tay et al., 2023] in R [R Core Team, 2022]) which is based on a linear combination using features in \mathbf{Z} , coefficients $\boldsymbol{\delta}$, and a ridge penalty. Below, the pre- and post-processing calculations are given.

Let $\mathbf{K} = \mathbf{Q}\mathbf{D}\mathbf{Q}^\top$ be the eigendecomposition of \mathbf{K} , with \mathbf{D} the diagonal matrix of positive eigenvalues so that \mathbf{K} is assumed to be positive definite and \mathbf{Q} the matrix of corresponding eigenvectors with $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}$, and also $\mathbf{Q}\mathbf{Q}^\top = \mathbf{I}$ since \mathbf{Q} is square. Then,

$$\lambda\boldsymbol{\eta}^\top\mathbf{K}^{-1}\boldsymbol{\eta} = \lambda(\boldsymbol{\eta}^\top\mathbf{Q}\mathbf{D}^{-1/2})(\mathbf{D}^{-1/2}\mathbf{Q}^\top\boldsymbol{\eta}) = \lambda\boldsymbol{\delta}^\top\boldsymbol{\delta} \quad (13)$$

using

$$\boldsymbol{\delta} = \mathbf{D}^{-1/2}\mathbf{Q}^\top\boldsymbol{\eta}$$

so that

$$\boldsymbol{\eta} = \mathbf{Q}\mathbf{D}^{1/2}\boldsymbol{\delta} = \mathbf{Z}\boldsymbol{\delta}.$$

Note that $\mathbf{Z} = \mathbf{Q}\mathbf{D}^{1/2}$ also implies that $\mathbf{Z} = (\mathbf{Q}\mathbf{D}\mathbf{Q})^\top\mathbf{Q}\mathbf{D}^{-1/2} = \mathbf{K}\mathbf{Q}\mathbf{D}^{-1/2}$.

In the case of approximate kernels, use the approximate kernel $\hat{\mathbf{K}} = \mathbf{X}\hat{\mathbf{A}}\mathbf{X}^\top$ instead of \mathbf{K} . Then

$$\begin{aligned} \boldsymbol{\eta} &= \mathbf{Z}\boldsymbol{\delta} \\ &= \hat{\mathbf{K}}\mathbf{Q}\mathbf{D}^{-1/2}\boldsymbol{\delta} \\ &= (\mathbf{X}\hat{\mathbf{A}}\mathbf{X}^\top)\mathbf{Q}\mathbf{D}^{-1/2}\boldsymbol{\delta} \\ &= \mathbf{X}(\hat{\mathbf{A}}\mathbf{X}^\top\mathbf{Q}\mathbf{D}^{-1/2}\boldsymbol{\delta}) \\ &= \mathbf{X}\boldsymbol{\gamma} \end{aligned}$$

so that $\boldsymbol{\gamma} = \hat{\mathbf{A}}\mathbf{X}^\top\mathbf{Q}\mathbf{D}^{-1/2}\boldsymbol{\delta}$. In this way, we obtain coefficient estimates that can be interpreted as quantifying effect sizes of the predictor variables on the response.

2.6 Test set predictions

Often there is the need to provide predicted values for unseen data in the test set \mathbf{X}_{test} with n_{test} rows. With kernels, the prediction becomes

$$\begin{aligned} \boldsymbol{\eta}_{\text{test}} &= \boldsymbol{\Phi}_{\text{test}}\boldsymbol{\beta} \\ &= \boldsymbol{\Phi}_{\text{test}}(\boldsymbol{\Phi}^\top\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^\top\boldsymbol{\eta} \end{aligned}$$

Using the SVD of $\boldsymbol{\Phi}$ from Footnote 1, $\boldsymbol{\eta}_{\text{test}}$ becomes

$$\begin{aligned} \boldsymbol{\eta}_{\text{test}} &= \boldsymbol{\Phi}_{\text{test}}(\mathbf{V}\boldsymbol{\Sigma}^{-2}\mathbf{V}^\top)(\mathbf{V}\boldsymbol{\Sigma}\mathbf{U}^\top)\boldsymbol{\eta} \\ &= \boldsymbol{\Phi}_{\text{test}}\mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{U}^\top\boldsymbol{\eta} \\ &= \boldsymbol{\Phi}_{\text{test}}(\mathbf{V}\boldsymbol{\Sigma}\mathbf{U}^\top)(\mathbf{U}\boldsymbol{\Sigma}^{-2}\mathbf{U}^\top)\boldsymbol{\eta} \\ &= \boldsymbol{\Phi}_{\text{test}}\boldsymbol{\Phi}^\top(\boldsymbol{\Phi}\boldsymbol{\Phi}^\top)^{-1}\boldsymbol{\eta} \\ &= \mathbf{K}_{\text{test}}\mathbf{K}^{-1}\boldsymbol{\eta}, \end{aligned}$$

where $\mathbf{K}_{\text{test}} = \Phi_{\text{test}} \Phi_{\text{test}}^\top$ is the between-training-test-block kernel matrix. $\mathbf{K}_{\text{test}} = \Phi_{\text{test}} \Phi_{\text{test}}^\top$ and has elements $k_{ij}(\mathbf{x}_{\text{test},i}, \mathbf{x}_j)$ where $\mathbf{x}_{\text{test},i}^\top$ denotes row i of \mathbf{X}_{test} . The prediction using kernels can be obtained by using $\mathbf{Z}_{\text{test}} = \mathbf{K}_{\text{test}} \mathbf{Q} \mathbf{D}^{-1/2}$ and the linear combination for the test data becomes

$$\begin{aligned}\boldsymbol{\eta}_{\text{test}} &= \mathbf{Z}_{\text{test}} \boldsymbol{\delta} \\ &= \mathbf{K}_{\text{test}} \mathbf{K}^{-1} \mathbf{X} \boldsymbol{\gamma}.\end{aligned}$$

Note that this prediction is *not* a linear combination of \mathbf{X}_{test} as it maps \mathbf{X}_{test} first to the high dimensional space of Φ_{test} .

Consider the alternative prediction for the test data using the approximate kernel approach as a linear combination from the training set coefficients $\boldsymbol{\gamma}$, that is,

$$\boldsymbol{\eta}_{\text{appr.test}} = \mathbf{X}_{\text{test}} \boldsymbol{\gamma}.$$

For the test data, this prediction comes at the cost that the reconstruction of \mathbf{K}_{test} is not optimal, not even in the case of a KAF of 1 for the training data. This can be seen as follows:

$$\begin{aligned}\boldsymbol{\eta}_{\text{appr.test}} &= \mathbf{X}_{\text{test}} \boldsymbol{\gamma} \\ &= \mathbf{X}_{\text{test}} \hat{\mathbf{A}} \mathbf{X}^\top \mathbf{Q} \mathbf{D}^{-1/2} \boldsymbol{\delta} \\ &= \mathbf{X}_{\text{test}} \hat{\mathbf{A}} \mathbf{X}^\top (\mathbf{X} \hat{\mathbf{A}} \mathbf{X}^\top)^{-1} \boldsymbol{\eta} \\ &= [\mathbf{X}_{\text{test}} (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{K} \mathbf{X} (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top] \hat{\mathbf{K}}^{-1} \boldsymbol{\eta}.\end{aligned}$$

This means that \mathbf{K}_{test} is approximated by

$$\hat{\mathbf{K}}_{\text{test}} = \mathbf{X}_{\text{test}} (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{K} \mathbf{X} (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top,$$

the approximate test kernel matrix $\hat{\mathbf{K}}_{\text{test}}$. Note that this approximation is based on \mathbf{K} only and does not make use of \mathbf{K}_{test} .

3 Computational aspects

Using the results of the previous section, a kernel solution using both exact and approximated kernels can be obtained by any standard software such as `glmnet`, using the following calculations.

1. Compute the kernel matrix \mathbf{K} , using the chosen kernel, for example the RBF kernel.
2. Compute the eigendecomposition $\mathbf{K} = \mathbf{Q} \mathbf{D} \mathbf{Q}^\top$.
3. Compute $\mathbf{Z} = \mathbf{Q} \mathbf{D}^{1/2}$.
4. Estimate $\boldsymbol{\delta}$ through standard software that has a ridge penalty using \mathbf{Z} as features.

5. Compute the linear combination as $\boldsymbol{\eta} = \mathbf{Z}\boldsymbol{\delta}$.
6. Compute the feature coefficients as $\boldsymbol{\gamma} = \widehat{\mathbf{A}}\mathbf{X}^\top\mathbf{Q}\mathbf{D}^{-1/2}\boldsymbol{\delta}$.
7. If there are n_{test} hold-out test data the linear combination for the test data becomes $\boldsymbol{\eta}_{\text{appr.test}} = \mathbf{X}_{\text{test}}\boldsymbol{\gamma}$.

In the case that one wants no penalty on the intercept by applying centring to \mathbf{K} , \mathbf{X} , and thus to $\boldsymbol{\eta}$ as in Section 2.4, one can simply read the formulas above in the computational steps with \mathbf{K} , \mathbf{X} , and thus to $\boldsymbol{\eta}$ being substituted by \mathbf{K}_c , \mathbf{X}_c , and $\boldsymbol{\eta}_c$ respectively. In case an inverse is needed, we replace the inverse by the Moore-Penrose inverse, for example, if \mathbf{K}_c^{-1} is needed, we use $\mathbf{K}_c^{-1} = \mathbf{K}_c^- = \mathbf{Q}\mathbf{D}^-\mathbf{Q}^\top$ where $\mathbf{K}_c = \mathbf{Q}\mathbf{D}\mathbf{Q}^\top$ is the eigen decomposition of \mathbf{K}_c , \mathbf{D} is the diagonal matrix with nonnegative eigenvalues, and \mathbf{D}^- is the diagonal matrix with diagonal elements d_{ii}^{-1} if $d_{ii} > 0$ and 0 otherwise.

4 Applications

4.1 Chemometric data set ‘apples’

To illustrate the use of approximated kernels, we use a data set by Zude et al. [2006] on a chemometric study on a sample of apples. The explanatory variables are spectroscopic data on 256 wavelengths, denoted here simply as 1 to 256, and the response variable is soluble solids content (SSC) in units of Brix (symbol $^\circ\text{Bx}$, a measure of the dissolved solids in a liquid, representing the strength of the solution as percentage by mass). The original data set has 642 apples, but we used a reduced sample of size $n = 179$. This ensures that there are many more variables than samples in the training set, in order to illustrate the benefit of the approximate RBF approach. We also use the compositional data analysis (CoDA) approach [Greenacre, 2018, 2021] since the relative values of the chemometric data are regarded as relevant, not their absolute values. The additive logratio transformation is applied to the data [Greenacre et al., 2021], thereby reducing the number of predictor variables from 256 to 255. In a separate exercise, it was confirmed that the logratio-transformed variables perform better than the original spectroscopic variables in predicting Brix.

Having established the predictor set, we compare two methods: (a) kernel ridge regression with the radial basis function kernel (RBF) using scaling parameter $1/p$, and (b) approximated kernel ridge regression using again the RBF kernel (Approx RBF). All computations were done in R. The optimal λ for each of these shrinkage methods was determined by 10-fold cross validation. The same folds were used over the two methods. For each method, 100 replications were done to see how stable the methods were over many random samples. The performance measure was the root mean squared prediction error (RMSE) and overall performance was summarized by boxplots of the 100 replications (Fig. 1) for the test set results, and a table of medians over the 100 replications, for both training and test sets (Table 1). Additionally in Table 1, for each replication, the RMSE of the two methods was ranked as lowest RMSE (1) or highest (2), and

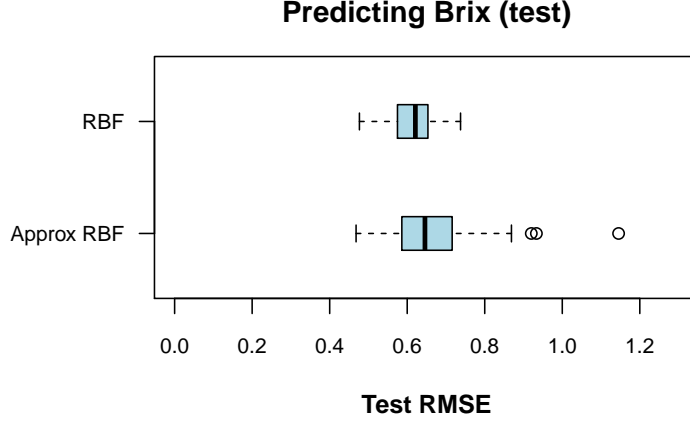


Figure 1: Boxplots of test RMSEs over 100 random splits of the data in 2/3 training data and 1/3 test data.

Table 1: Median RMSE and mean ranks for the training and test sets in the prediction Brix over 100 random samples of $n = 119$ from the total sample $N = 179$ and the remaining $n_{\text{test}} = 60$ as test samples, for KRR with the RBF kernel and the approximated RBF kernel. Lower values are better.

	Median RMSE		Mean rank	
	Training	Test	Training	Test
RBF	0.323	0.621	1.54	1.38
Approx RBF	0.323	0.646	1.46	1.62

their average ranks over the replications was taken, the average being preferred here because the individual values are integers.

The kernel ridge regression performs the best, as expected, with mean rank on the test set equal to 1.38, but does not produce any estimates of effect sizes. The median coefficient estimates in Fig. 2 show more variation in the middle spectral bands than at the extremes, showing some clear positive and negative peaks in the middle bands that predict the response variable Brix. Notice that the RMSEs for the RBF and the Approx RBF are identical on the training set, because the number of variables exceeds the number of samples.

4.2 Microbiome data set ‘Crohn’

Data set ‘Crohn’ is available in the R package `coda4microbiome` [Calle et al., 2023] in a modified form, with zero counts replaced (see Greenacre [2024]). The data are published in their original form with zeros in the selbal R package and

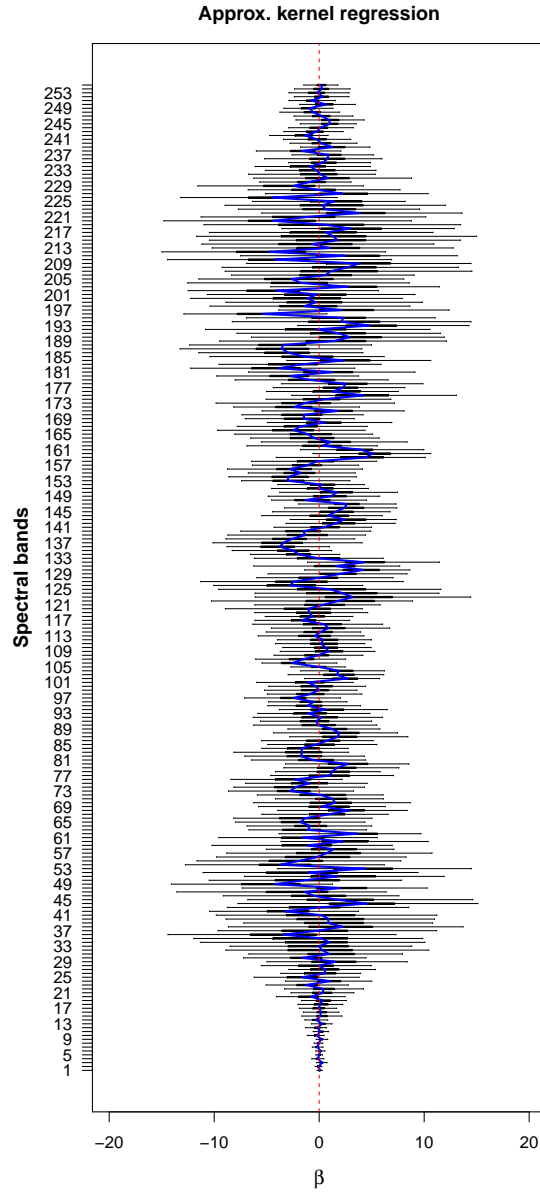


Figure 2: Boxplots of coefficients for regression and approximated kernel regression, to predict Brix. Boxplots are obtained by randomly splitting the data a 100 times in 2/3 training data and 1/3 test data. The blue lines connect the medians of the coefficients over the replications.

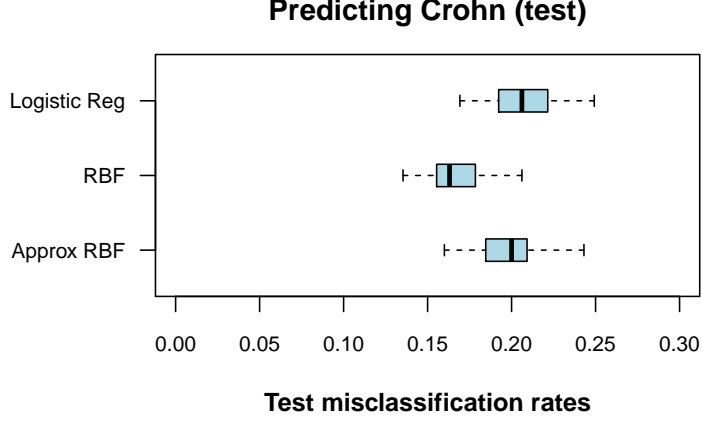


Figure 3: Boxplots of test misclassification rates over 100 random splits of the data in 2/3 training data and 1/3 test data.

analysed by Rivera-Pinto et al. [2018] – this original data set is analysed here. The data form a matrix of counts of bacterial species aggregated into $p = 48$ genera on $n = 975$ human samples. In addition, each sample has been classified as having the digestive ailment called Crohn’s disease (662 samples) or not (313 samples).

These data are compositional, and their total count in each sample is irrelevant, and are thus expressed as proportions of their respective sample totals: $y_{ij} = x_{ij} / \sum_{j=1}^p x_{ij}$. Furthermore, Greenacre [2024] has shown that a power transformation of these compositional data, including the zeros, with a power of 0.28 (i.e., practically a fourth-root), is optimal for predicting the disease: $z_{ij} = y_{ij}^{0.28}$. These transformed values z_{ij} are used as the features in the following. In this case where $p < n$, we include logistic regression as a third method for comparison with RBF and Approx RBF.

We did 100 random splits of the data in 2/3 training data and 1/3 test data, again stratified to respect the proportions of Crohn and non-Crohn in the whole data set. Figure 3 shows the misclassification rate (i.e., proportion of incorrect predictions) in the test set for each of the three methods. On the test data, we see that the RBF kernel still performs best, then the approximated kernel approach followed by regular logistic regression. Ranking per replication the best performing method on the test data by 1 and the worst by 3, the average rank on the test data is 1.00 for the RBF kernel (i.e., it is the best on all replications), 2.38 for the approximated kernel approach, and 2.62 for logistic regression (Table 2).

For the Crohn data, the median KAF over the 100 replications was 0.749, indicating that the approximated RBF kernel accounts for approximately three quarters of the RBF kernel penalty.

The boxplots of Figure 4 summarize the estimated coefficients for the 100

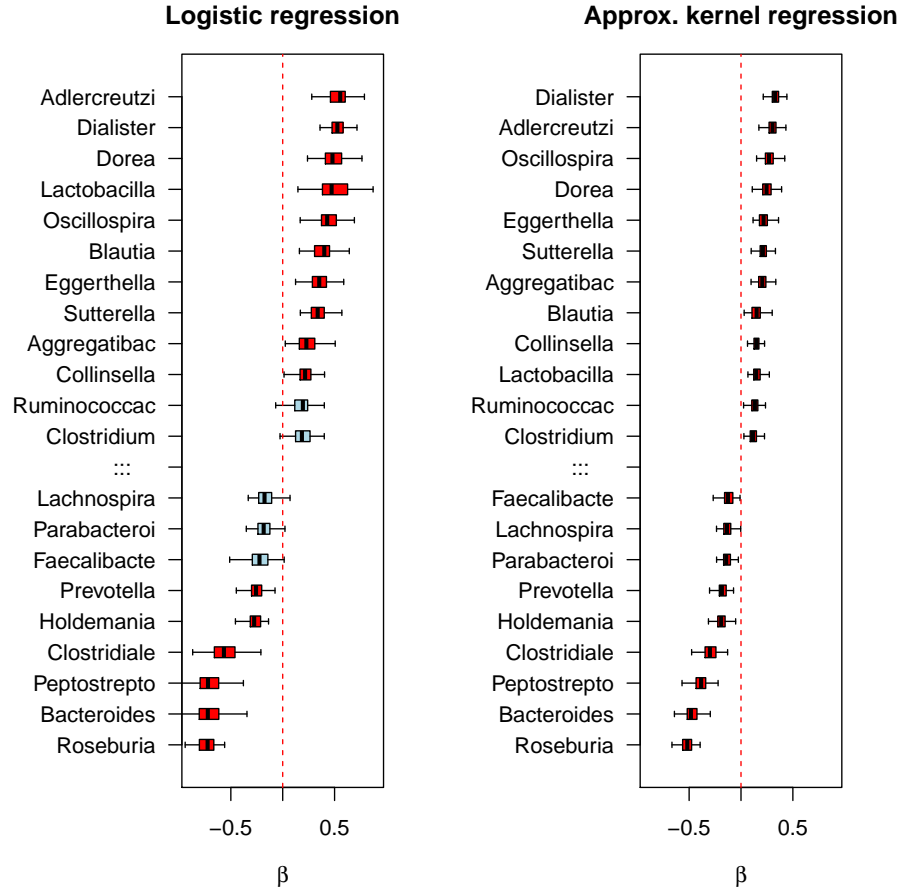


Figure 4: Boxplots of coefficients for logistic regression and approximated kernel logistic regression, to predict presence or absence of Crohn’s disease. Boxplots are obtained by randomly splitting the data a 100 times in 2/3 training data and 1/3 test data. Red boxes indicate that zero is not in the range of the replicate values. The figure only reports variables (bacteria) that have a red box for the approximated kernel. Some bacteria names have been shortened.

Table 2: Median misclassification rates and mean ranks for the training and test sets in the prediction of Crohn’s disease over 100 random samples of $n = 650$ from the total sample $N = 975$ and the remaining $n_{\text{test}} = 325$ as test samples, for the three methods. Lower values are better.

	Median misclass		Mean rank	
	Training	Test	Training	Test
Logistic Reg	0.169	0.206	2.25	2.62
RBF	0.029	0.163	1.00	1.00
Approx RBF	0.173	0.200	2.75	2.38

replications, for the two methods where coefficients can be computed. It can be seen in both methods that the increased presence of bacteria *Dialister* and several others consistently predict higher probability of Crohn’s disease, whereas increased presence of bacteria such as *Roseburia* and *Bacteroides* consistently predict less chance of the disease. The estimated coefficients for the approximated kernel logistic regression are a shrunk version of those for the logistic regression (Fig. 4). Moreover, the dispersions of the estimates over replications in the right hand plot in Fig. 4 are seen to be much lower.

Finally, as a comparison, Greenacre [2024] performed variable selection on the same data set, using the same power transformation in the logistic regression, and chose only 14 out of the 48 predictors as being significant. There the misclassification rates of 0.184 (training set) and 0.214 (test set) were found, using only one replication of the same random training/test set split. These rates are within the bounds of the replications of Fig. 3, but both slightly worse than the median values of 0.169 and 0.206 in Table 2 using all 48 predictors. The 14 selected predictors coincide exactly with the 8 top positive predictors and 6 top negative predictors in the boxplots in the logistic regression results of Fig. 4, all of which have dispersions well separated from the zero line.

5 Discussion and conclusion

The present paper shows through linear algebra that it is possible with a wide predictor matrix to reconstruct exactly predictions obtained by kernel with a ridge penalty, and approximately so if the predictor matrix is tall.

A caveat is that the numeric computations depend on inverses and eigen decompositions of size $n \times n$. The order of operations needed to compute these are $O(n^3)$, so that computationally, our results get harder to compute as n gets into the ten thousands. For those cases, approximate kernels could be used. Section 2.5 reexpressed the kernel penalty as a regular ridge penalty with predictor variables being the principal components of the kernel matrix. Therefore, efficient approximations could be done by first finding a limited number of principal components of the kernel matrix and use the methodology of this paper to

ensure that the approximated kernel matrix is as close as possible to the kernel matrix. We leave this as a topic for further research.

In line with the nonparametric approaches often taken in the machine learning literature, we relied on sampling approaches to establish coefficients to be statistically different from zero. For kernel generalized linear models, it would also be possible to derive p -values for the coefficients based on statistical tests that make use of effective degrees of freedom.

References

- S. Ali, T. Abuhmed, S. El-Sappagh, K. Muhammad, J. M. Alonso-Moral, R. Confalonieri, R. Guidotti, J. Del Ser, N. Díaz-Rodríguez, and F. Herrera. Explainable artificial intelligence (XAI): What we know and what is left to attain trustworthy artificial intelligence. *Information Fusion*, 99:101805, 2023. ISSN 1566-2535. doi: 10.1016/j.inffus.2023.101805.
- M. Calle, M. Pujolassos, and A. Susin. coda4microbiome: Compositional data analysis for microbiome cross-sectional and longitudinal studies. *BMC Bioinformatics*, 24:82, 2023. doi: 10.1186/s12859-023-05205-3.
- C. Campbell. Kernel methods: a survey of current techniques. *Neurocomputing*, 48:63–84, 2002.
- L. Crawford, K. Wood, X. Zhou, and S. Mukherjee. Bayesian approximate kernel regression with variable selection. *Journal of the American Statistical Association*, 113:1710–1721, 2018. doi: 10.1080/01621459.2017.1361830.
- R. Dwivedi, D. Dave, H. Naik, S. Singhal, R. Omer, P. Patel, B. Qian, Z. Wen, T. Shah, G. Morgan, and R. Ranjan. Explainable AI (XAI): Core ideas, techniques, and solutions. 55(9), 2023. doi: 10.1145/3561048.
- J. Friedman, R. Tibshirani, and T. Hastie. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1): 1–22, 2010. doi: 10.18637/jss.v033.i01.
- M. Greenacre. *Compositional Data Analysis in Practice*. Chapman & Hall / CRC Press, Boca Raton, Florida, 2018. doi: 10.1201/9780429455537.
- M. Greenacre. Compositional data analysis. *Annu Rev Stat Appl*, 8:271–99, 2021. doi: 10.1146/annurev-statistics-042720-124436.
- M. Greenacre. The chiPower transformation: a valid alternative to logratio transformations in compositional data analysis. *Advances in Data Analysis and Classification*, 2024. doi: 10.1007/s11634-024-00600-x.
- M. Greenacre, M. Martínez-Alvaro, and A. Blasco. Compositional data analysis of microbiome and any-omics datasets: a revalidation of the additive logratio transformation. *Frontiers in Microbiology*, 2021. doi: 10.3389/fmicb.2021.727398.

- T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning. Second Edition*. Springer, 2008.
- J. Mercer. Functions of positive and negative type and their connection with the theory of integral equations. *Philos.Trans.Roy.Soc.London A*, 209:415–446, 1909.
- R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2022. URL <https://www.R-project.org/>.
- J. Rivera-Pinto, J. J. Egozcue, V. Pawlowsky-Glahn, R. Paredes, M. Noguera-Julian, and M. L. Calle. Balances: a new perspective for microbiome analysis. *mSystems*, 3:e00053–18, 2018. doi: 10.1128/mSystems.00053-18.
- C. Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1:206–215, 2019. ISSN 1369-7412. doi: 10.1038/s42256-019-0048-x.
- J. K. Tay, B. Narasimhan, and T. Hastie. Elastic net regularization paths for all generalized linear models. *Journal of Statistical Software*, 106(1):1–31, 2023. doi: 10.18637/jss.v106.i01.
- G. Wahba. *Spline models for observational data*. SIAM, 1990.
- M. Zude, B. Herold, J.-M. Roger, V. Bellon-Maurel, and S. Landahl. Non-destructive tests on the prediction of apple fruit flesh firmness and soluble solids content on tree and in shelf life. *Journal of Food Engineering*, 77(2):254–260, 2006. ISSN 0260-8774. doi: 10.1016/j.jfoodeng.2005.06.027. Progress on Bioproducts Processing and Food Safety.