Journal of Data Science,17(4). P. 674 - 695,2019
DOI:10.6339/JDS.201910_17(4).0003

WEIGHTED ORTHOGONAL COMPONENTS REGRESSION ANALYSIS

Xiaogang Su∗[1], Yaa Wonkye[2], Pei Wang[3], and Xiangrong Yin[3]

[1]Department of Mathematical Sciences, University of Texas, El Paso, TX 79968

[2]Department of Mathematics and Statistics, Bowling Green State University, Bowling Green, OH 43403

[3]Department of Statistics, University of Kentucky, Lexington, KY 40536

ABSTRACT

In the linear regression setting, we propose a general framework, termed weighted orthogonal components regression (WOCR), which encompasses many known methods as special cases, including ridge regression and principal components regression. WOCR makes use of the monotonicity inherent in orthogonal components to parameterize the weight function. The formulation allows for efficient determination of tuning parameters and hence is computationally advantageous. Moreover, WOCR offers insights for deriving new better variants. Specifically, we advocate assigning weights to components based on their correlations with the response, which may lead to enhanced predictive performance. Both simulated studies and real data examples are provided to assess and illustrate the advantages of the proposed methods.

Keywords: AIC; BIC; GCV; Principal components regression; Ridge regression.

1. Introduction

Consider the typical multiple linear regression setting where the available data L: $= \{(y_i, \mathbf{x}_i): \; i = 1, \ldots, n\}$ consist of n i.i.d. copies of the continuous response y and the predictor vector $\mathbf{x} \in \mathbb{R}^p$. Without loss of generality (WLOG), we assume y_i's are centered and x_{ij}'s are standardized throughout the article. Thus the intercept term is presumed to be 0 in linear models, which have the general form $\mathbf{y} = \mathbf{X}\beta + \boldsymbol{\varepsilon}$ with $\mathbf{y} = (y_i)$ and random error vector $\boldsymbol{\varepsilon} \sim \; (0, \sigma^2 \mathbf{I}_n)$. For the sake of convenience, we sometimes omit the subscript i. When the $n \times p$ design matrix $\mathbf{X}$ is of full column rank p, the ordinary least squares (OLS) estimator $\boldsymbol{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$, as well as its corresponding predicted value $\hat{y}(\mathbf{x}') = \mathbf{x}'^T\boldsymbol{\beta}$ at a new observation $\mathbf{x}'$, enjoys many attractive properties.

However, OLS becomes problematic when $\mathbf{X}$ is rank-deficient, in which case the Gram matrix $\mathbf{X}^T\mathbf{X}$ is singular. This may happen either because of multicollinearity when the predictors are highly correlated or because of high dimensionality when $p \quad n$. A wealth of proposals have been made to combat the problem. Besides others, we are particularly concerned with a group of techniques that include ridge regression (RR; Hoerl and Kennard,1970), principal components regression (PCR; Massy,1965), partial least squares regression (PLSR;Wold,1966&1978), and continuum regression (CR; Stone and Brooks,1990). One common feature of these approaches lies in the fact that they first extract orthogonal or uncorrelated components that are linear combinations of $\mathbf{X}$ and then regress the response directly on the orthogonal components. The number of orthogonal components doesn't exceed n and p, hence reducing the dimensionality. This is the key how these types of methods approach high-dimensional or multicollinear data.

In this article, we introduce a general framework, termed weighted orthogonal components regression (WOCR), which puts the aforementioned methods into a unified class. Compared to the original predictors in $\mathbf{X}$, there is a natural ordering in the orthogonal components. This information allows us to parameterize the weight function in WOCR with low-dimensional parameters, which are essentially the tuning parameters, and estimate the tuning parameters via optimization. The WOCR formulation also facilitates a convenient comparison of the available methods and suggests their new natural variants by introducing more intuitive weight functions.

We restrict our attention to PCR and RR models. The remainder of the article is organized as follows. In Section2, the general framework of WOCR is introduced. Section3 exemplifies the applications of WOCR with RR and PCR. More specifically, we demonstrate how WOCR formulation can be used to estimate the tuning parameter in RR and select the number of principal components in PCR, and then introduce their better variants on the basis of WOCR. Section4 presents numerical results from simulated studies that are designed to illustrate and assess WOCR and make comparisons with others. We also provide real data illustrations in Section5. Section6 concludes with a brief discussion, including the implication of WOCR on PLSR and CR models.

2. Weighted Orthogonal Components Regression (WOCR)

Denote m = rank($\mathbf{X}$) so that $m \leq (p \wedge n)$. Let $\{\mathbf{u}_1, \ldots, \mathbf{u}_m\}$ be the orthogonal components extracted in some principled way, satisfying that $\mathbf{u}_j^T \mathbf{u}_{j'} = 0$ if $j \neq j'$ and 1 otherwise. Here

$\{\mathbf{u}_j\}_{j=1}^m$ forms an orthonormal basis of the column space of **X**, C(**X**) = {**Xa**: for some **a** $\in$ R^p}. Since $\mathbf{u}_j \in C(\mathbf{X})$, suppose $\mathbf{u}_j = \mathbf{X}\mathbf{a}_j$ for j = 1, . . . , m. The condition $\mathbf{u}_j^T\mathbf{u}_{j'} = 0$ implies that $\mathbf{a}_j^T \boldsymbol{X}^T\boldsymbol{X}\boldsymbol{a}_{j'} = 0$, i.e., vectors $\mathbf{a}_j$ and $\mathbf{a}_{j'}$ are $\mathbf{X}^T\mathbf{X}$-orthogonal, which implies that $\mathbf{a}_j$ and $\mathbf{a}_{j'}$ are orthogonal if, furthermore, either $\mathbf{a}_j$ or $\mathbf{a}_{j'}$ is an eigenvector of $\mathbf{X}^T\mathbf{X}$ associated with a non-zero eigenvalue. Letting $\mathbf{U}_{n\times m} = [\mathbf{u}_1, \ldots, \mathbf{u}_m]$ and $\mathbf{A}_{p\times m} = [\mathbf{a}_1, \ldots, \mathbf{a}_m]$, we have $\mathbf{U} = \mathbf{XA}$ in matrix form with $\mathbf{U}^T\mathbf{U} = \mathbf{I}_m$, but it is not necessarily true that $\mathbf{U}\mathbf{U}^T = \mathbf{I}_n$. The construction of matrix A may (e.g., in RR and PCR) or may not (e.g., in PLSR and CR) depend on the response y; again, our discussion will be restricted to the former scenario. It is worth noting that extracting m components reduces the original n × p problem into an n × m problem, hence achieving an automatic dimension reduction.

2.1 Model Specification

The general form of a WOCR model can be conveniently expressed in terms of its fitted vector

$$\hat{\boldsymbol{y}} = \sum_{j=1}^{m} w_j \langle \boldsymbol{y}, \boldsymbol{u}_j\rangle \boldsymbol{u}_j = \sum_{j=1}^{m} w_j \gamma_j \boldsymbol{u}_j, \tag{1}$$

where $\gamma_j = \langle \mathbf{y}, \mathbf{u}_j \rangle = \mathbf{y}^T \mathbf{u}_j$ is the regression coefficient and $0 \le w_j \le 1$ is the weight for the j-th orthogonal component $\mathbf{u}_j$. We shall reserve the notation $\tilde{\mathbf{y}}$ for WOCR fitted vector. Denoting $\mathbf{W}_{m\times m} = \text{diag}(w_j)$, (1) becomes

$$\tilde{\mathbf{y}} = \mathbf{U}\mathbf{W}\mathbf{U}^T\mathbf{y} = \mathbf{XAWU}^T\mathbf{y} \tag{2}$$

in matrix form, recalling that $\mathbf{U} = \mathbf{XA}$. We will see that RR, PCR, and many others are all special cases of the above WOCR specification, with different choices of $\{\mathbf{u}_j, w_j\}$. For example, if $w_j = 1$ or $\mathbf{W} = \mathbf{I}_m$, then (2) amounts to the OLS fitting, since $\tilde{\mathbf{y}} = \mathbf{U}\mathbf{U}^T\mathbf{y}$ is the projection of $\mathbf{y}$ on C(**U**) in this case and C (**U**) = C (**X**).

This WOCR formulation allows us to conveniently study its general properties. It follows immediately from (2) that the associated hat matrix **H** is

$$\mathbf{H} = \mathbf{U}\mathbf{W}\mathbf{U}^T = \mathbf{XAWU}^T. \tag{3}$$

The resultant sum of square errors (SSE) is given by SSE $= \mathbf{y}^T(\mathbf{I}_n - \mathbf{H})^2\mathbf{y}$. Note that **H** is not an idempotent or projection matrix in general, neither is $(\mathbf{I} - \mathbf{H})$. Instead,

$$(\mathbf{I} - \mathbf{H})^2 = \mathbf{I} - 2\mathbf{H} + \mathbf{H}^2 = \mathbf{I} - \mathbf{U}(2\mathbf{W} - \mathbf{W}^2)\mathbf{U}^T.$$

The diagonal matrix $(2\mathbf{W} - \mathbf{W}^2)$ has diagonal element $\{1 - (1 - w_j)^2\}$. Therefore,

$$\begin{aligned} \text{SSE} &= \mathbf{y}^T\mathbf{y} - \mathbf{y}^T\mathbf{U}\text{diag}\{1 - (1 - w_j)^2\}\mathbf{U}^T\mathbf{y} \\ &= \|\boldsymbol{y}\|^2 - \sum_{j=1}^{m} \left(2w_j^2 - w_j^2\right)\gamma_j^2 \end{aligned} \tag{4}$$

From (2), the WOCR estimate of β is

$$\tilde{\boldsymbol{\beta}} = \mathbf{AWU}^T\mathbf{y} \tag{5}$$

It follows that, given a new data matrix **X**, the predicted vector $\tilde{\mathbf{y}}$ can be obtained as

$$\tilde{\boldsymbol{y}}' = \boldsymbol{X}'\tilde{\boldsymbol{\beta}} = \boldsymbol{X}'\boldsymbol{A}\boldsymbol{W}\boldsymbol{U}^T\boldsymbol{y} \tag{6}$$

Although not further pursued here, many other quantities and properties of WOCR can be derived accordingly with the generic form, including $E\left\|\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\|^2$ as studied in Hoerl and Kennard (1970) and Hwang and Nettleton (2003).

2.2 Parameterizing the Weights

The next important component in specifying WOCR is to parameterize the weights in $\mathbf{W}$ in a principled way. The key motivation stems from the observation that, compared to the original regressors in $\mathbf{X}$, the orthogonal components in $\mathbf{U}$ are naturally ordered in terms of some measure. This ordering may be attributed to the observed variation in $\mathbf{X}$ that each $\mathbf{u}_j$ is intended to account for. Another natural ordering is based on the coefficients $|\gamma_j|$. Because of orthogonality, the regression coefficient $\gamma_j = <\mathbf{y}, \mathbf{u}_j>$ remains the same for $\mathbf{u}_j$ in both the simple regression and multiple regression settings.

This motivates us to parameterize the weights w_j based on the ordering measures. It is intuitive to assign more weights to more important components. To do so, w_j can be specified as a function monotone in the ordering measure and parameterized with a low-dimensional vector $\boldsymbol{\lambda}$. Two such examples are given in Figure1. Among many other choices, the usage of sigmoid functions will be advocated in this article because they provide a smooth approximation to the 0-1 hard-thresholding indicator function that is useful for the component selection purpose and they are also flexible enough to adjust for achieving improved prediction accuracy. In general, we denote $w_j = w_j(\boldsymbol{\lambda})$. The vector $\boldsymbol{\lambda}$ in the weight function are essentially the tuning parameters. This parameterization expands these conventional modeling methods by providing several natural WOCR variants that are more attractive, as illustrated in the next section.

Determining the tuning parameters λ is yet another daunting task. In common practice, one fits the model at a number of fixed λ values and then resorts to cross-validation or a model selection criterion to select the best tuning parameter. This can be computationally intensive, especially with massive data. When a model selection criterion is used, WOCR provides a computationally efficient way of determining the tuning parameter λ. The key idea is to plug the specification (1) in a model selection criterion and optimize with respect to λ. Depending on the scenarios, commonly used model selection criteria include the Akaike information criterion (AIC; Akaike, 1974), the generalized cross- validation (GCV; Golub, Heath, and Wahba, 1979), and the Bayesian information criterion (BIC; Schwarz,1978). The terms involved in these model selection criteria are essentially SSE and the degrees of freedom (DF). A general form of SSE is given by (4). For DF, we follow the generalized definition by Efron (2004):

$$\mathrm{DF}(\lambda) = \mathrm{E}\{\mathrm{tr}(d\hat{\mathbf{y}}/d\mathbf{y})\} \tag{7}$$

If neither the components $\mathbf{U}$ nor the weights w_j depend on $\mathbf{y}$, then DF, often termed as the effective degrees of freedom (EDF) in this scenario, can be computed as

$$\mathrm{EDF} = \mathrm{tr}(\mathbf{H}) = \mathrm{tr}(\mathbf{U}\mathbf{W}\mathbf{U}^T) = \mathrm{tr}(\mathbf{W}\mathbf{U}^T\mathbf{U}) = \mathrm{tr}(\mathbf{W}) = \sum_{j=1}^{m} w_j^2. \tag{8}$$

When either components $\mathbf{U}$ or the weights w_j depends on $\mathbf{y}$, the computation of DF is more difficult and will be treated on a case-by-case basis.

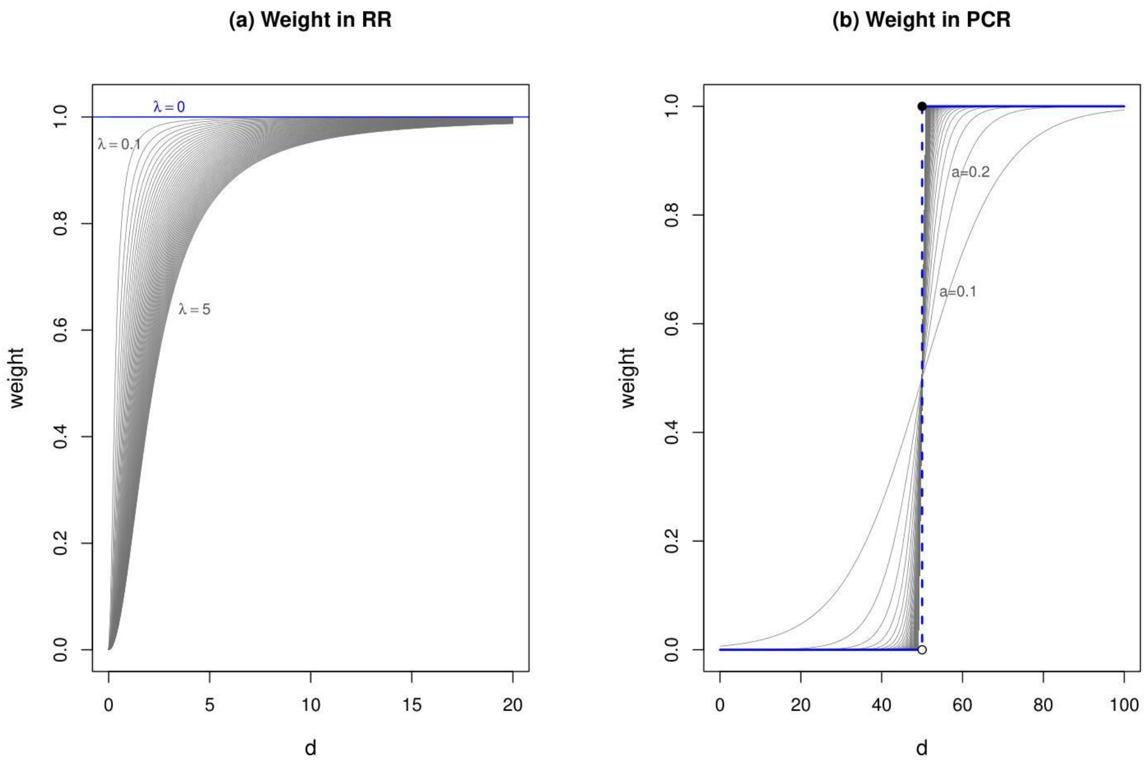

Figure 1: Plot of the weights used in ridge regression (RR) and principal components regression (PCR) as a function of the singular values d_j of X: (a) $w(d) = d^2/(d^2 + \lambda)$ in RR for $\lambda = 0.0, 0.1, 0.2, \ldots, 5.0$; (b) the discrete threshold $w(d) = I(x \geq c)$ in PCR with $c = 50.0$, approximated with the expit weight $w(d) = \text{expit}\{a\,(d - c)\}$ for $a = \{0.1, 0.2, \ldots, 50.0\}$.

The specific forms of GCV, AIC, and BIC can be obtained accordingly. We treat the model selection as an objective function for λ. The best tuning parameter $\hat{\lambda}$ can then be estimated by optimization. Since λ is of low dimension, the optimization can be solved efficiently. This saves the computational cost in selecting the tuning parameter for a great deal.

3. WOCR Models

We show how several conventional models relate to WOCR with different weight specifications and different ways of constructing the orthogonal components $\mathbf{U} = \mathbf{XA}$ and how the WOCR formulation can help improve and expand them. In this section, we first discuss how WOCR helps determine the optimal tuning parameter λ in ridge regression and make inference accordingly. Next, we show that WOCR facilitates an efficient computational method for selecting the number of components in PCR. The key idea is to approximate the 0-1 threshold function with a smooth sigmoid weight function. Several natural variants of RR and PCR that are advantageous in predictive modeling are then derived within the WOCR framework.

3.1 Pre-Tuned Ridge Regression

The ridge regression (Hoerl and Kennard, 1970) can be formulated as a penalized least squares (LS) optimization problem

$$\min \|\boldsymbol{y} - \boldsymbol{X\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|^2$$

with tuning parameter λ. The solution yields the ridge estimator

$$\widehat{\boldsymbol{\beta}}_R = (\boldsymbol{X}^T\boldsymbol{X} + \lambda \boldsymbol{I}_P)^{-1}\boldsymbol{X}\boldsymbol{y}.$$

The singular value decomposition (SVD) of data matrix X offers a useful insight into RR (see, e.g., Hastie, Tibshirani, and Friedman, 2009). Suppose that the SVD of $\mathbf{X}$ is given by

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T = \sum_{j=1}^{m} d_j \boldsymbol{u}_j \boldsymbol{v}_j, \tag{9}$$

where both $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_m] \in \mathrm{R}^{n\times m}$ and $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_m] \in \mathrm{R}^{p\times m}$ have orthonormal column vectors that form an orthonormal basis for the column space $\mathsf{C}(\mathbf{X})$ and the row space $\mathsf{C}(\mathbf{X}^T)$ of $\mathbf{X}$, respectively, and matrix $\mathbf{D} = \mathrm{diag}\,(d_j)$ with singular values satisfying $d_1 \geq d_2 \geq \cdots \geq d_m > 0$. Noticing that $\mathbf{X}^T\mathbf{X} = \mathbf{V}\mathbf{D}^2\mathbf{V}^T$, the column vectors of $\mathbf{V}$ yield the principal directions. Since $\mathbf{X}\mathbf{v}_j = d_j\mathbf{u}_j$, it can be seen that $\mathbf{u}_j$ is the j-th normalized principal component.

The fitted vector in RR conforms well to the general form (1) of WOCR, as established by the following proposition. The proof is deferred to the appendix.

Proposition 3.1. Regardless of the magnitude of $\{n, p, m\}$, the fitted vector $\widehat{\boldsymbol{y}} = \boldsymbol{x}\widehat{\boldsymbol{\beta}}_{\boldsymbol{R}}$ in ridge regression can be written as

$$\widehat{\boldsymbol{y}}_{\boldsymbol{R}} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X} + \lambda \boldsymbol{I}_P)^{-1}\boldsymbol{X}\boldsymbol{y} = \boldsymbol{U}\boldsymbol{W}\boldsymbol{U}^{\boldsymbol{T}}\boldsymbol{y} = \sum_{j=1}^{m} d_j^2/(d_j^2 + \lambda)\,\langle y, u_j\rangle u_j u_j, \tag{10}$$

with $\mathbf{W}=\mathrm{diag}\{w_j\}$and $w_j = d_j^2/(d_j^2 + \lambda)$ for j=1,…m.

One natural ordering of the principal components $\mathbf{u}_j$s is based on their associated singular values d_j. Hence, the weight function $w_j = w(d_j; \lambda) = d_j^2/(d_j^2 + \lambda)$ is monotone in d_j and parameterized with one single parameter λ. See Figure 1(a) for a graphical illustration of this weight function. In view of $\mathbf{XV} = \mathbf{UD}$, matrix $\mathbf{A}$ in WOCR is given as $\mathbf{A} = \mathbf{V}\mathbf{D}^{-1}$.

Since RR is most useful for predictive modeling without considering component selection, GCV is an advisable criterion for selecting the best tuning parameter $\hat{\lambda}$. With our WOCR approach, we first plugging (10) into GCV to form an objective function for λ and then optimize it with respect to λ ,On the basis of (4) and (8), the specific form of GCV(λ) is given up to some irrelevant constant,

$$GCV(\lambda) \propto \frac{SSE}{(n-EDF)^2} = \frac{\|y\|^2 - \Sigma_{j=1}^m (w_j^2 - 2w_j)\gamma_j^2}{\left(n - \Sigma_{j=1}^m w_j\right)^2}. \tag{11}$$

GCV has a wide applicability even in the ultra-high dimensions. Alternatively, AIC can be used instead. If $\lim_{n\to\infty} m/n = 0$, GCV is asymptotically equivalent to AIC(λ) $\propto$ n ln(SSE) + 2·EDF.

The best tuning parameter in RR can be estimated as $\hat{\lambda} = \mathrm{argmin}_\lambda$ GCV(λ). Bringing $\hat{\lambda}$ back to (10) yields the final RR estimator. Since the tuning parameter is determined beforehand, we call this method 'pre-tuning'. We denote this pre-tuned RR method as RR(d; λ), where the first argument d indicates the ordering on which basis the components are sorted and the second argument indicates the tuning parameter λ. We shall use this as a generic notation for other new WOCR models. As we shall demonstrate with simulation in Section4.1, RR(d; λ) provides nearly identical fitting results to RR; however, pre-tuning dramatically improves the computational efficiency, especially when dealing with massive data.

Remark 1. One statistically thorny issue with regularization is selection of the tuning parameter. First of all, this is a one-dimensional optimization, yet done in an inefficient way in the current practice by selecting a grid of values and evaluating the objective function at each value. The pretuned version helps amend this deficiency. Secondly, although the tuning parameter λ is often selected adaptively and hence is a statistic, no statistical inference is made for the tuning parameter unless within the Bayesian setting. The above pre-tuning method yields a convenient way of making inference on λ. Since the objective function GCV(λ) is smooth in λ, the statistical properties of $\hat{\lambda}$ follow well through standard M-estimation arguments. However, this is not part of the theme in this paper, thus we shall not pursue further.

3.2 Pre-Tuned PCR

PCR regresses the response on the first k ($1 \leq k \leq m$) principal components as given by the SVD of X in (9). The fitted vector in PCR can be rewritten as

$$\hat{\boldsymbol{y}}_{PCR} = \sum_{j=1}^{k} \langle \boldsymbol{y}, \boldsymbol{u}_j \rangle \boldsymbol{u}_j = \sum_{j=1}^{m} \delta_j \gamma_j \boldsymbol{u}_j,$$

where $\gamma_j = \langle \boldsymbol{y}, \boldsymbol{u}_j \rangle$ and $\delta_j = I(j \leq k)$ for $j = 1, \ldots, m$. Clearly, PCR can be put in the WOCR form with $w_j = \delta_j$. Conventionally, the ordering of principal components is aligned with the singular values $\{d_j\}$; thus we may rewrite $\delta_j = \delta(d_j; c) = I(d_j \geq c)$ with a threshold value $c = d_k$ if k is known. Either the number of components k or the threshold c is the tuning parameter. Selecting the optimal k by examining many PCR models with leading components is a discrete process.

To facilitate pre-tuning, we replace the indicator weight $\delta(x; c) = I(x \geq c)$ with a smooth sigmoid function. While many other choices are available, it is convenient to use the logistic or expit function $\pi(x) = \text{expit}(x) = \{1 + \exp(-x)\}^{-1}$ so that

$$w_j = \pi(d_j; a, c) = \text{expit}\{a(d_j - c)\} \quad (12)$$

Figure 1(b) plots expit$\{a(x - c)\}$ with $c = 50.0$ for different choices of $a > 0$. It can be seen that a larger a value yields a better approximation to the indicator function $I(x \geq 0)$, while a smaller a yields a smoother function which is favorable for optimization. In order to emulate PCR, the parameter a can be fixed *a priori* at a relatively large value. Our numerical studies show that the performance of the method is quite robust with respect to the choice of a. On that basis, we recommend fixing a in the range of $[1, 100]$.

Since PCR involves selection of the optimal number of PCs, BIC, given by BIC($\boldsymbol{\lambda}$) $\propto$ n ln(SSE)+ ln(n) $\cdot$ DF, is selection-consistent (Yang,2005) and often has a superior empirical performance in variable selection. The hat matrix $\mathbf{H}$ in PCR is idempotent, so is $\mathbf{I}_n - \mathbf{H}$. Thus the SSE can be reduced a little bit as $\mathbf{y}^T(\mathbf{I}_n - \mathbf{H})\mathbf{y}$, which then can be approximated by substituting $\delta(d_j; c)$ with $\pi(d_j; a, c)$. The DF can be approximately in a similar way as DF $= k = \sum_j \delta(d_j; c) \approx \sum_j \pi(d_j; ac)$. This results in the following form for BIC

$$BIC(c) \propto nln(\|y\|^2 - \sum_{j=1}^{m} w_j \gamma_j^2) + ln(n) \sum_{j=1}^{m} w_j\ , \quad (13)$$

which is treated as an objective function of c. We estimate the best cutoff point $\hat{c}$ by optimizing BIC(c) with respect to c. This is a one-dimensional smooth optimization problem with a search range $c \in [d_1, d_m]$. Once $\hat{c}$ is available, we use it as a threshold to select the components and fit a regular PCR. We denote this pre-tuned PCR approach as PCR(d; a). Compared to the discrete selection in PCR, PCR(d; a) is computationally more efficient.

3.3 WOCR Variants of RR and PCR Models

Not only can many existing models be cast into the WOCR framework, but it also suggests new favorable variants. We explore some of them. One immediate variant of PCR is to leave both a and c free in (13). More specifically, we first obtain $(\hat{a}, \hat{c}) = \mathrm{argmin}_{a,c}\, \mathrm{BIC}(a, c)$ and then compute the WOCR fitted vector in (1) with weight $w_j = \exp\{\hat{a}(d_j - \hat{c})\}$ for $j = 1, \ldots, m$. This will give PCR more flexibility and adaptivity and hence may lead to improved predictive power. In this approach, selecting components is no longer a concern; thus GCV or AIC can be used as the objective function instead. We denote this approach as PCR(d; a, c).

The principal components are constructed independently from the response. Artemiou and Li (2009) and Ni (2011) argued that the response tends to be more correlated with the leading principal components. However, this is usually not the case in reality; see, e.g., Jollife (1982) and Hadi and Ling (1998) for real-life data illustrations. Nevertheless, there has not been a principled way to deal with this issue in PCR. WOCR can provide a convenient solution: one simply bases the ordering of $\mathbf{u}_j$ on the regression coefficients γ_j and defines the weights w_j via a monotone function of $|\gamma_j|$ or, preferably, γ^2. Doing so will induce dependence on the response to the weights. As a result, the associated DF has to be computed differently, as established in Proposition3.2.

Proposition 3.2. *Suppose that the WOCR model (1) has orthogonal components $\mathbf{u}_j$ constructed independently of $\mathbf{y}$ and weights $w_j = w(\gamma^2; \boldsymbol{\lambda})$, where $w(\cdot)$ is a smooth monotonically increasing function and $\boldsymbol{\lambda}$ is the parameter vector. Its degrees of freedom (DF) can be estimated as*

$$\widehat{DF} = \sum_{j=1}^{m} \left(2\gamma_j^2 \dot{w}_j + w_j\right) \tag{14}$$

Where $\dot{w}_j = dw(\gamma_j^2; \lambda)/d(\gamma_j^2)$.

Clearly both PCR and RR can be benefited from this reformulation. As a variant of RR, the weight now becomes $w_j = w(\gamma^2; \lambda) = \gamma^2/(\gamma^2 + \lambda)$ and hence $\dot{w}_j = \lambda/(\gamma^2 + \lambda)^2$. It follows that the estimated DF is

$$\widehat{DF} = \sum_{j=1}^{m} \left(\gamma_j^4 + 3\lambda\gamma_j^2\right)/\left(\gamma_j^2 + \lambda\right)^2$$

The best tuning parameter $\hat{\lambda}$ can be obtained by minimizing GCV. Using similar notations as earlier, we denote this RR variant as RR(γ; λ). It is worth noting that RR(γ; λ) is, in fact, not a ridge regression model. Its solution can no longer be nicely motivated by a regularized or constrained least squares optimization problem as in the original RR. But what really matters in these methods is the predictive power. By directly formulating the fitted values $\hat{\mathbf{y}}$, the WOCR model (1) facilitates a direct and flexible model specification that focuses on prediction.

Table 1: WOCR Variants of ridge regression (RR) and principal components regression (PCR) models, both based on the normalized principal components $\{\mathbf{u}_j : j = 1, \ldots p\}$.

Model	Component Ordering	Weights	Tuning Parameter	Suggested WOCR Objective Function
$\mathrm{RR}(d;\lambda)$	d_j	$w_j = d_j^2/(d_j^2+\lambda)$	λ	$\mathrm{GCV}(\lambda)$
$\mathrm{RR}(\gamma;\lambda)$	γ_j^2	$w_j = \gamma_j^2/(\gamma_j^2+\lambda)$	λ	$\mathrm{GCV}(\lambda)$
$\mathrm{PCR}(d;c)$	d_j	$w_j = \mathrm{expit}\{a(d_j - c)\}$ with fixed a	c	$\mathrm{BIC}(c)$
$\mathrm{PCR}(d;a,c)$	d_j	$w_j = \mathrm{expit}\{a(d_j - c)\}$	a, c	$\mathrm{GCV}(a,c)$
$\mathrm{PCR}(\gamma;c)$	γ_j^2	$w_j = \mathrm{expit}\{a(\gamma_j^2 - c)\}$ with fixed a	c	$\mathrm{BIC}(c)$
$\mathrm{PCR}(\gamma;a,c)$	γ_j^2	$w_j = \mathrm{expit}\{a(\gamma_j^2 - c)\}$	a, c	$\mathrm{GCV}(a,c)$

For PCR, the weight becomes $w_j = \pi(\gamma_j^2; a, c)$, hence $\dot{w}_j = a w_j(1-w_j)$ and

$$\widehat{DF} = \sum_{j=1}^{m} w_j\left(2a\gamma_j^2 + 1 - 2a w_j \gamma_j^2\right)$$

Depending on whether or not we want to select components, we may fix $a > 0$ at a larger value or leave it free. This results in two PCR variants, which we denote as PCR(γ; c) and PCR(γ; a, c), respectively.

Table1summarizes the WOCR models that we have discussed so far. Among them, RR(d; λ) and PCR(d; c) resemble the conventional RR and PCR, yet with pre-tuning. Depending on the analytic purpose, we also suggest a preferable objective function for each WOCR model. In general, we have recommended using GCV for predictive purposes, in which scenarios AIC can be used as an alternative. AIC is equivalent to GCV if $\lim_{n\to\infty} p/n = 0$, both being selection-efficient in the sense prescribed by Shibata (1981). On the other hand, if selecting components is desired, using BIC is recommended.

Remark 2. It is worth noting that the WOCR model PCR(γ; c) has a close connection with the MIC (Minimum approximated Information Criterion) sparse estimation method of Su (2015) and Su et al. (2016,2018). MIC yields sparse estimation in the ordinary regression setting by solving a p-dimensional smooth optimization problem

$$\min_{\gamma} \; n \ln\|\boldsymbol{y} - \boldsymbol{XW\gamma}\|^2 + \log(\mathrm{n})\,\mathrm{tr}(\mathbf{W}),$$

where $\mathbf{W}$=diag(w_j) with diagonal elements w_j=tanh($a\gamma_j^2$) approximating the indictor function I($\gamma^2 \neq$ c). Comparatively, PCR(γ;c) solves a one-dimensional optimization problem

$$\min_{c} \; n \ln\|y - \boldsymbol{UW\gamma}\|^2 + \log(\mathrm{n})\, tr(\boldsymbol{W}),$$

where $\mathbf{W}$=diag(w_j) with diagonal elements w_j=expit(a($\gamma_j^2 - c$)) approximating I($\gamma_j^2 \geq c$)

The substantial simplification in PCR(γ; c) is because of the orthogonality of the design matrix $\mathbf{U}$. Hence the coefficient estimates $\boldsymbol{\gamma}$ in multiple regression are the same as those in simple regression and can be computed ahead. Furthermore, the orthogonal regressors $\mathbf{u}_j$, i.e., the columns of $\mathbf{U}$, are naturally ordered by γ^2. This allows us to formulate a one-parameter smooth approximation to the indicator function I($\gamma^2 \geq$ c), which achieves selection of $\mathbf{u}_j$ in this PCR variant.

3.4 Implementation: R Package WOCR

The proposed WOCR method is implemented in an R package **WOCR**. The current version is hosted on GitHub at https://github.com/xgsu/WOCR. The main function `WOCR()` has an argument `model=` with options in RR.d.lambda, RR.gamma.lambda, PCR.d.c, PCR.gamma.c, PCR.d.a.c, and PCR.gamma.a.c, which correspond to the six WOCR variants listed in Table 1. Among them, RR($d;\lambda$), RR(γ; λ), PCR(d; c), and PCR(γ; c) each involve a one-dimensional smooth optimization. This can be solved via the Brent (1973) method, available in the R function `optim()`. Owing to the nonconvex nature, dividing the search range of the decision variable can be helpful. The other two methods, PCR(d; a, c) and PCR(γ; a, c), each involve a two-dimensional smooth nonconvex optimization. Mullen (2014) provides a comprehensive comparison of many global optimization algorithms currently available in R (R Core Team,2018). We have followed her suggestion to choose the generalized simulated annealing method (Tsallis and Stariolo, 1996), available from the R package **GenSA** (Xiang et al., 2013). More details of the implementation can be found in the help file of the **WOCR** package.

[1] Simulation Studies

This section presents some of the simulation studies that we have conducted to investigate the performance of WOCR models and its comparison with other methods.

4.1 Comparing Ridge Regression with RR*(d; λ)*

We first compare the conventional ridge regression with its pretuned version, i.e., RR(d; λ). The data are generated as follows. We simulate the design matrix $\mathbf{X} \in \mathrm{R}^{n\times p}$ from a multivariate normal distribution $N(\mathbf{o}, \mathbf{\Sigma})$ with $\mathbf{\Sigma} = (\sigma_{jj'})$ and $\sigma_{jj'} = \rho^{|j-j'|}$ for $j, j' = 1, ..., p$. Apply SVD to extract matrix $\mathbf{U}$ and $\mathbf{D}$. Then we form the mean response as

$$\text{Model A: } \mathbf{y} = \sum \mathbf{b_j u_j} + \boldsymbol{\varepsilon} \text{ with } m = p \wedge n \text{ and } \boldsymbol{\varepsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_n), \tag{15}$$

where $\mathbf{b} = (b_j) = [m, m - 1, . . . , 1]^T/10$. For each simulated data set, we apply RR (as implemented by the R function `lm.ridge`) and RR(d; λ), both selecting λ with minimum GCV.

To compare, we consider the mean square error (MSE) for prediction. To this end, a test data set of 500 observations is generated in advance. The fitted RR and RR(d; λ) from each simulation run will be applied to the test set and the MSE is obtained accordingly. The 'best' tuning parameter $\hat{\lambda}$ is also recorded. We only report the results for the setting $\rho = 0.5$, $\sigma^2 = 1$, $p = 100$, $\mathbf{b} = (b_j) = (p, p - 1, \ldots, 1)^T/10$. Two sample sizes $n \in \{50, 500\}$ are considered. For each model configuration, a total of 200 simulation runs are considered.

In the simulation, we found how to specify the search points could be a problem in the current practice of ridge regression. Initially, we found the ridge regression gave inferior performance com- pared to RR(d; λ) in many scenarios. However, after adjusting its search range, the results became

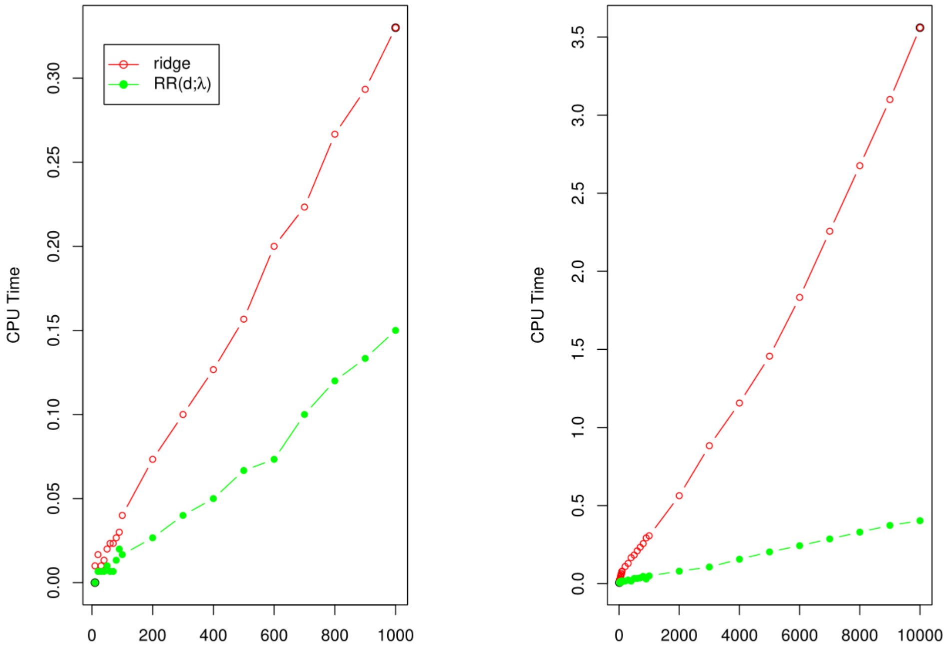

Figure 2: CPU Computing time comparison between ridge regression (RR) and its WOCR variant RR(d; λ) in selecting the best λ via minimum GCV: (a) with varying p and fixed n = 100 and (b) with varying p and fixed n = 100.

nearly identical to what RR(d; λ) had. This point will be further illustrated in Section4.3. It is also worth noting that the minimum GCV tends to select a very small λ in the ultra-high dimensional case with $p > n$.

To demonstrate the computational advantages of RR(d;λ) over RR, we generated data from the same model A in (15). We first fix $n = 100$ and let p vary in $\{10, 20, \ldots, 100, 200, \ldots 1000\}$. And then we fix $p = 100$ and let n vary in $\{10, 20, \ldots, 100, 200, \ldots, 1000, 2000, \ldots, 10000\}$. For each setting, we recorded the CPU computing time for RR and RR(d; λ) averaged from three simulation runs. We have set the search range for λ as $\{0.1, 0.2, \ldots, 100\}$. The results are plotted in Figure2(a) and2(b). It can be seen that RR(d; λ) is much faster than RR, especially when either p or n gets large.

4.2 Comparing PCR Variants

Next we compare PCR with its WOCR variants. Data are generated from Model A in (15) with two sets of dimensions: $\{n = 1,000, p = 200\}$ and $\{n = 200, p = 1,000\}$, of which the latter illustrates a $p > n$ scenario. We consider two choices of coefficients $\mathbf{b} = (b_j) \in \mathrm{R}^p$as follow

$$\text{Model A1: } \mathrm{b_j} = \begin{cases} 10 & j = 1,\ldots,5 \\ -10 & j = 6,\ldots,10 \\ 0 & j = 11,\ldots,p \end{cases}$$

$$\text{Model A2: } \mathrm{b_j} = \begin{cases} 0 & j = 1,\ldots,50 \\ 10 & j = 51,\ldots,55 \\ -10 & j = 56,\ldots,60 \\ 0 & j = 61,\ldots,p \end{cases} \qquad (16)$$

In Model A1, the underlying assumption in PCR that the leading principal components associated with larger singular values d_j are more important in predicting y is met, while this is not case in Model A2. In both models, the true number of important components is 10.

For PCR, we included four methods for selecting the optimal number of components: 10-fold cross validation (CV) as implemented in R Package **pls** (Mevik and Wehrens, 2007), AIC, BIC, and LASSO (Tibshirani, 1996), following the suggestion of a referee. Owing to high dimensions, best subset selection with AIC or BIC is not available. Thus we have restricted their selection process to conform to the PCR assumption by considering subsets of leading components only. As a result, only LASSO is able to select components on basis of $|\gamma_j|$. With orthogonal components, the LASSO solution has an explicit form $\hat{\gamma}_j^{LASSO} = sgn(\gamma_j)(|\gamma_j| - \lambda)_+$ with a tuning parameter $\lambda \geq 0$ and $x_+ = 0 \vee x$. The 10-fold CV as implemented in R Package **ncvreg** (Breheny, 2018) is used to determine the optimal tuning parameter λ in LASSO. For comparison, all four WOCR variants for PCR in Table1are included. In PCR(d; c) and PCR(γ; c), the fixed parameter a = 50 is used as default.

A total of 200 simulation runs are used for each model configuration. For a generated dataset, the number of selected components is recorded for every method. To investigate predictive performance, a test sample of size 500 is generated beforehand and the resultant mean square error (MSE) is obtained for each method in each simulation run. Table 2 reports the averaged MSE (± SE) and the median number of selected components out of 200 simulation runs.

In terms of component selection, PCR(γ; c) clearly stands out prominently. It is not surprising that PCR with AIC or BIC, and PCR(d; c) may work well in Model A1, but fail badly in Model A_2, where the assumption underlying PCR is unmet. BIC is better than AIC in selecting components for PCR in most settings. The 10-fold cross validation tends to overfit by selecting more components than necessary. In terms of prediction accuracy, PCR(γ; a, c) wins out substantially, owing to its adaptability to γ_j and flexibility in weighting other components with parameters (a, c). LASSO also does well, but it performance has been compromised by its biased solution with soft thresholding. The PCR(γ; c) method comes next; it seems to suffer from its rigid selection of components. We have also experimented with the different a choices for PCR(d; c) and PCR(γ; c); results are not reported here. For any reasonably large value of a $\in$ [1, 100], the performances of PCR(d; c) and PCR(γ; c) are quite stable with some minor variations. On this basis, we recommend simply fixing a = 50 for standardized predictors.

Table 2: Comparison of PCR and its WOCR variants. Data are generated from Model A. Performance measures include the averaged mean square error (MSE) for prediction (average ± standard error), and the median number of selected components by each method, out of 200 simulation runs for each configuration.

			$n = 1,000$ and $p = 200$		$n = 200$ and $p = 1,000$	
Model	Method	Variant	#Components	MSE	#Components	MSE
A1	PCR	CV	24	3.8123 ± 0.0367	141.5	4.1301 ± 0.0241
		AIC	10	3.7934 ± 0.0361	169	4.1368 ± 0.0250
		BIC	10	3.7910 ± 0.0360	10	3.9199 ± 0.0208
		LASSO	37	3.5972 ± 0.0312	37	3.7204 ± 0.0201
	WOCR	PCR(d; c)	11	3.7926 ± 0.0360	12	3.9260 ± 0.0210
		PCR(d; a, c)	200	3.7920 ± 0.0361	200	3.9737 ± 0.0221
		PCR(γ; c)	10	3.7918 ± 0.0360	10	3.9227 ± 0.0210
		PCR(γ; a, c)	200	3.4134 ± 0.0300	200	3.5947 ± 0.0251
A2	PCR	CV	77.5	4.0300 ± 0.0298	169	4.5172 ± 0.0218
		AIC	60	4.0178 ± 0.0295	169	4.5190 ± 0.0219
		BIC	60	4.0138 ± 0.0293	60	4.3280 ± 0.0205
		LASSO	39	3.7127 ± 0.0246	41.5	3.9824 ± 0.0186
	WOCR	PCR(d; c)	62	4.0145 ± 0.0293	64	4.3395 ± 0.0206
		PCR(d; a, c)	200	4.0153 ± 0.0294	200	4.3656 ± 0.0218
		PCR(γ; c)	10	3.9633 ± 0.0289	10	4.2692 ± 0.0201
		PCR(γ; a, c)	200	3.5420 ± 0.0299	200	3.7896 ± 0.0298

4.3 Predictive Performance Comparisons

To assess the predictive performance of all WOCR models, we generate data of size $n = 500$ from two nonlinear models in Friedman(1991), as given below:

$$\text{Model B: } y = 0.1\exp(4x_1) + 4\,\text{expit}\{20(x_2 - 0.5)\} + 3x_3 + 2x_4 + x_5 + \varepsilon; \qquad (17)$$

$$\text{Model C: } y = 10\sin(\pi x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + x_5 + \varepsilon \quad . \qquad (18)$$

The covariates of dimension p are independently generated from the uniform[0,1] distribution and the random error term ε follows $N(0, 1)$. In both models, only the first five predictors are involved in the mean response function. Two choices of $p \in \{5, 50\}$ are considered. For each simulated data set, ridge regression, PCR, and six WOCR variants in Table1are trained with the default or recommended settings. In particular, we fix the scale parameter $a = 50$ in PCR(d; c) and PCR(γ; c). To apply ridge regression, we have used $\lambda \in \{0.01, 0.02, \ldots, 200\}$. To assess the predictive performance, a test data set of 500 observations is generated and each trained model is applied to make predictions. The results are integrated over 200 simulation runs. Table3presents the prediction MSE (mean and SE) and the median number of selected components by each method.

First of all, the ridge regression appears to provide the worst MSE results in several scenarios. This is again because of the deficiencies involved in the current practice of ridge regression. Ridge estimators are computed for discrete set of λ values within a user-specified range, which may not even include the true global GCV minimum point.

Comparatively, RR(d; λ) provides a computationally efficient and reliable way of finding the 'best' tuning parameter. We could have refit the ridge regression according to $\hat{\lambda}$ suggested by RR(d; λ). Another interesting observation is that RR(γ; λ) tends to give more favorable results than RR(d; λ), which comes as no surprise since sorting the components according to $|\gamma_j|$ borrows strength from the association with the response.

Among PCR variants, neither PCR(d; c) nor PCR(γ; c) performs well. On the basis of BIC, they are aimed to find a parsimonious true model when the true model is among the candidate models, which, however, is not the case here. In terms of prediction accuracy, it can be seen that RR(γ; λ), PCR(d; a, c), and PCR(γ; a, c) are highly competitive, all yielding similar performances to PCR. Note that PCR determines the best tuning parameter via 10-fold cross-validation, while PCR(d; a, c) and PCR(γ; a, c) are based on a smooth optimization of GCV and hence are computationally advantageous. In the simulation settings under consideration, PCR has selected all components and hence simply amounts to the OLS fitting.

Table 3: Comparison on predictive accuracy of ridge regression (RR), principal components regression (PCR) with their six WOCR variants. Data (with $n = 500$) were generated from Models B and C. Performance measures include the averaged MSE, the standard errors of MSE, and the median number of selected components by each method, out of 200 simulation runs for each configuration.

			Models							
			RR$(d;\lambda)$	RR$(\gamma;\lambda)$	RR	PCR$(d;c)$	PCR$(d;a,c)$	PCR$(\gamma;c)$	PCR$(\gamma;a,c)$	PCR
Model B	$p=5$	average–MSE	3.130	1.895	38.844	3.048	1.806	2.915	1.807	1.806
		SE–MSE	0.0074	0.0094	1.1494	0.0972	0.0012	0.0577	0.0012	0.0012
		# comps	5	5	5	4	5	2	5	5
	$p=50$	average–MSE	2.485	2.057	6.051	2.499	2.059	2.773	2.075	2.062
		SE–MSE	0.0071	0.0052	0.1654	0.0382	0.0053	0.0813	0.0058	0.0054
		# comps	50	50	50	46	50	29	50	50
Model C	$p=5$	average–MSE	10.335	6.930	192.528	10.356	6.644	9.403	6.644	6.645
		SE–MSE	0.0251	0.0345	8.8263	0.2830	0.0058	0.1559	0.0059	0.0059
		# comps	5	5	5	4	5	2	5	5
	$p=50$	average–MSE	9.058	7.223	54.781	9.257	7.210	9.617	7.226	7.229
		SE–MSE	0.0230	0.0188	3.0581	0.1745	0.0181	0.2677	0.0185	0.0187
		# comps	50	50	50	44	50	28.5	50	50

5. Real Data Examples

For further illustration, we apply WOCR to two well-known data sets, which are `BostonHousing2` and `concrete`. The Boston housing data relate to prediction the median value of owner-occupied homes for 506 census tracts of Boston from the 1970 census. We used the corrected version BostonHousing2 available from R package **mlbench** (Leisch and Dimitriadou, 2012), with dimension $n = 506$ observations and $p = 17$ predictors. The concrete data are available from the UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/datasets/). The goal of this data set is to predict the concrete compressive strength based on a few characteristics of the concrete. The data set has $n = 1,030$ observations and $p = 8$ predictors.

Figure 3 plots the singular values d_j and the regression coefficients in absolute value, $|\gamma_j|$, for both data sets. It can be seen that d_j decreases gradually as expected. The bar plot of $|\gamma_j|$, however, shows different patterns. In the BostonHousing2 data, the very first component is highly correlated with the response, while others shows alternate weak correlations. In the concrete data, the third component is most correlated with the response, followed by the 6th and 5th principal components. The first two components are only very weakly correlated. This data set shows a good example where the top components are not necessarily the most relevant components in terms of association with the response.

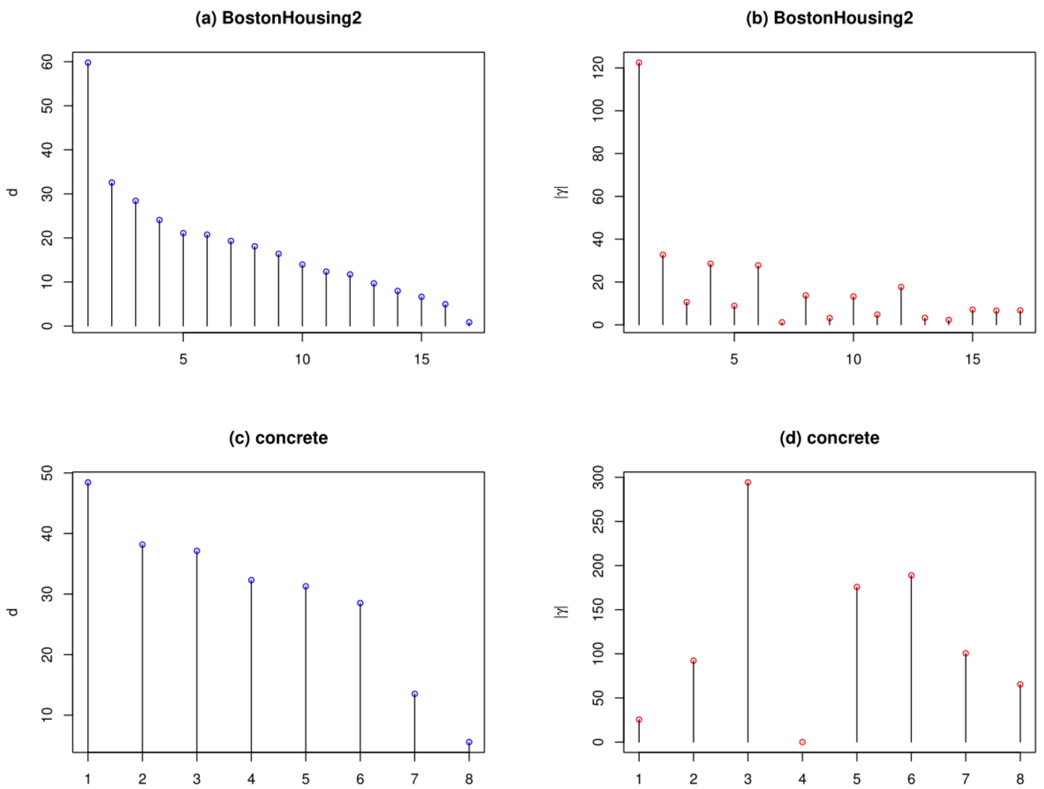

Figure 3: Bar plots of the singular values d_j and the absolute values of coefficients $|\gamma_j|$ for six real data sets.

To compare different models, a unified approach is taken. We randomly partition the data into a training set and a test set with a ratio of approximately 2:1 in sample sizes. The training set is used to construct models and then the constructed models are applied to the test set for prediction. The default settings in Table1are used for each WOCR, while the default 10-fold CV method is used to select the best tuning parameter for ridge regression and PCR. We repeat this entire procedure for 200 runs. The prediction MSE and the number of components for every method is recorded for each run. The results are summarized in Table4.

Table 4: Comparison on predictive accuracy of ridge regression (RR), principal components regression (PCR) with their WOCR variants on two real data sets: BostonHousing2 and concrete. The best performers are highlighted in boldface.

	BostonHousing2				concrete		
Method	average-MSE	SE-MSE	# comps		average-MSE	SE-MSE	# comps
Ridge	15.6630	0.1641	17		110.4587	0.4627	8
RR$(d;\lambda)$	**15.6207**	0.1628	17		110.4581	0.4627	8
RR$(\gamma;\lambda)$	15.6532	0.1652	17		**110.1557**	0.4630	8
PCR	15.9962	0.1671	13.14		110.1968	0.4633	8
PCR$(d;c)$	16.2175	0.1594	9.98		123.6064	0.5406	6
PCR$(d;a,c)$	15.7529	0.1772	17		110.1903	0.4633	8
PCR$(\gamma;c)$	21.6077	0.1793	1		137.0467	2.0446	2.99
PCR$(\gamma;a,c)$	**15.6596**	0.1747	17		**110.1852**	0.4631	8

While most methods provide largely similar results, some details are noteworthy. For ridge regression, RR$(d;\lambda)$ outperforms the original ridge regression slightly but it is much faster in computation time. Comparatively, RR$(\gamma;\lambda)$ improves the prediction accuracy by basing the weights on γ_j's for the concrete data, where the top components are not the most relevant to the response as shown in Figure3. Among the PCR models, both PCR$(d;a,c)$ and PCR$(\gamma;a,c)$ are among top performers in terms of prediction.

Neither PCR$(d;c)$ nor PCR$(\gamma;c)$ performs as well as others in terms of prediction accuracy owing to their different emphasis. Concerning component selection, PCR$(\gamma;c)$ yields simpler models than PCR$(d;c)$ and PCR. This is determined by the nature of each method and the data sets. Referring to Figure 3, PCR$(\gamma;c)$ clearly helps extract parsimonious models with simpler structures.

6. Discussion

We have proposed a new way of constructing predictive models based on orthogonal components extracted from the original data. The approach makes efficient use of the natural monotonicity associated with those orthogonal components. It allows streamlined determination of the tuning parameters. The framework results in several interesting alternative models to RR and PCR. These new WOCR variants make improvement on either predictive performance or selection of the components. Overall speaking, RR$(\gamma;\lambda)$, PCR$(d;a,c)$, and PCR$(\gamma;a,c)$ are highly competitive in terms of predictive performance. PCR(γ; c) better aims for model parsimony by making selection on the basis of association with the response.

WOCR can be implemented with more flexibility. First of all, we have advocated the use of logistic or expit function in regulating the weights. The logistic function expit$\{a(x-c)\}$ is rotationally symmetric about the point (c, 0.5). To have more flexible weights, we may consider a generalized version of the expit function, gexpit(x; a, b, c) = $1/[1 + b\exp\{-a(x - c)\}]$. The range of the gexpit function remains (0, 1). Since its value at $x = c$ is now $1/(1 + b)$, the parameter $b > 0$ changes the rotational symmetry unless b = 1. Secondly, selecting the number of principal components is a major concern in PCR. We have used BIC in both PCR(d; c) and PCR(γ; c) for this purpose. BIC is derived in the fixed dimensional setting (i.e., fixed p and $n \to \infty$). It is worth noting that the dimension in the WOCR family is m instead of p. If m is close to n, the modified or generalized BIC (see, e.g., Chen and Chen,2008) can be used instead. In particular, the complexity penalty $[\ln\{\ln(m)\}\ln(n)]$ suggested by Wang, Li, and Leng(2009) to replace $\ln(n)$ in (13) for diverging dimensions fits well for WOCR models since the dimension m cannot exceed n. If there is prior information or belief that the optimal k is less than some pre-specified number, it is helpful to further restrain the search range of c on the basis of $\{d_j : j = 1, \ldots, m\}$.

The WOCR model framework generates several future research revenues. First of all, WOCR can be directly applicable to regression with components after a varimax rotation (Kaiser, 1958). WOCR can also be extended to PLSR and CR models. In those approaches, extraction of the orthogonal components takes associations with the response into consideration; thus both matrices **A** and **W** relate to **y**. To select the tuning parameter, v-fold cross validation can be conveniently used on the basis of Equations (5) and (6). To implement pre-tuning, finding the degrees of freedom involved in these approach becomes more complicated but remains doable by following Krämer and Sugiyama (2011). The weighting and pre-tuning strategy introduced in WOCR may help make improvement in terms of predictive accuracy, computational speed, and model parsimony for these methods. Secondly, the simulation results for Model B in (17) and Model C in (18) with $p = 50$ presented in Section4.3highlight the variable selection issue in high-dimensional modeling. To this end, Bair (2006) considered a univariate screening step; Ishwaran and Rao (2014) showed the generalized ridge regression (Hoerl and Kennard,1970) can help suppress the influence of unneeded predictors in certain conditions. Both approaches may be incorporated into WOCR to improve its predictive ability. Finally, WOCR can be extended to generalized linear models, e.g., via a local quadratic approximation of the log-likelihood function. The kernel trick (see, e.g., Rosipal and Trejo 2002, Rosipal, Trejo, and Cichoki, 2011, and Lee and Liu, 2013) can be integrated into WOCR as well.

APPENDIX

A Proof

Proof of Proposition (3.1). The proof when m = p (i.e., $p \le n$ and hence $\mathbf{V}^{-1} = \mathbf{V}^T$) can be found in, e.g., Hastie, Tibshirani, and Friedman (2009). We consider the general case including the $p > n$ scenario. With the general SVD form (9) of **X**, we have $\mathbf{U}^T\mathbf{U} = \mathbf{V}^T\mathbf{V} = \mathbf{I}_m$, but it is not necessarily true that $\mathbf{U}\mathbf{U}^T = \mathbf{I}_n$, nor for $\mathbf{V}\mathbf{V}^T = \mathbf{I}_p$.

First, plugging the SVD of **X** into $\hat{\mathbf{y}}_R$ yields

$$\hat{\boldsymbol{y}}_R = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^T + \lambda \boldsymbol{I}_p)^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{y}. \tag{19}$$

Define $\mathbf{V}' = [\mathbf{v}_1, \cdots, \mathbf{v}_m, \mathbf{v}_{m+1}, \ldots, \mathbf{v}_p] \in \mathbf{R}^{p\times p}$ by completing an orthonormal basis for R^p. Hence $\mathbf{V}'$ is invertible with $\mathbf{V}'^{-1} = \mathbf{V}'^T$. Also define $\mathbf{U}_0 = [\mathbf{U}, \mathbf{O}] \in \mathbf{R}^{p\times p}$ and $\mathbf{D}_0 = \mathrm{diag}\{d_1, \ldots, d_m, 0, \ldots, 0\} \in \mathbf{R}^{p\times p}$ by appending 0 matrix $\mathbf{O}$ or components to $\mathbf{U}$ and $\mathbf{D}$. Then it can be easily checked that $\hat{\mathbf{y}}_R$ in (19) can be rewritten as

$$\begin{aligned}
\hat{\boldsymbol{y}} &= \boldsymbol{U_0}\boldsymbol{D_0}\boldsymbol{V'^T}\left(\boldsymbol{V}\boldsymbol{D_0^2}\,\boldsymbol{V^T} + \boldsymbol{\lambda}\boldsymbol{I_p}\right)^{-1}\boldsymbol{V'}\boldsymbol{D_0}\boldsymbol{U_0^T}\,\boldsymbol{y} \\
&= \boldsymbol{U_0}\boldsymbol{D_0}\boldsymbol{V'^T}\left\{\boldsymbol{V}(\boldsymbol{D_0^2} + \lambda\boldsymbol{I_p})\boldsymbol{V'^T}\right\}^{-1}\boldsymbol{V'}\boldsymbol{D_0}\boldsymbol{U_0^T}\,\boldsymbol{y} \\
&= \boldsymbol{U_0}\boldsymbol{D_0}\boldsymbol{V'^T}\boldsymbol{V}'\left(\boldsymbol{D_0^2} + \lambda\boldsymbol{I_p}\right)^{-1}\boldsymbol{V'^{-1}}\boldsymbol{V'}\boldsymbol{D_0}\boldsymbol{U_0^T}\,\boldsymbol{y} \\
&= \boldsymbol{U_0}\boldsymbol{D_0}\left(\boldsymbol{D_0^2} + \lambda\boldsymbol{I_p}\right)^{-1}\boldsymbol{D_0}\boldsymbol{U_0^T}\,\boldsymbol{y}
\end{aligned}$$

with $W = \mathrm{diag}\{d_j^2/(d_j^2 + \lambda)\}$.

Proof of Proposition (3.2). The WOCR model in this case is $\tilde{\boldsymbol{y}} = \Sigma_{j=1}^{m} w_j\gamma_j\boldsymbol{u}_j$, with $\gamma_j = \boldsymbol{u}_j^T\boldsymbol{y}, w_j = w(\gamma_j^2;\lambda)$. It follows by chain rule that

$$\frac{d\tilde{y}}{dy} = \Sigma_{j=1}^{n}(2\gamma_j^2\,\dot{w}_j + w_j)\boldsymbol{u}_j\boldsymbol{u}_j^T = \boldsymbol{U}\,diag\left(2\gamma^2\dot{w}_j + w_j\right)\boldsymbol{U}^T.$$

Following the definition of DF by Efron (2004), an estimate is given by

$$\mathrm{tr}\left(\frac{d\tilde{\mathbf{y}}}{d\mathbf{y}}\right) = \mathrm{diag}(2\gamma_j^2\,\dot{w}_j + w_j)\mathbf{U}^T\mathbf{U} = \Sigma_{j=1}^{m}\left(2\gamma^2\dot{w}_j + w_j\right),$$

which completes the proof.

References

[1] Akaike, H. (1974). A new look at model identification, IEEE Transactions an Automatic Control, 19: 716–723.

[2] Artemiou, A. A. and Li, B. (2009). On principal components and regression: A statistical explanation of a natural phenomenon. Statistica Sinica, 19: 1557–1565.

[3] Bair, E., Hastie, T., Paul, D., and Tibshirani, R. (2006). Prediction by supervised principal components. Journal of the American Statistical Association, 101(473): 119–137.

[4] Breheny, P. (2018). R Package **ncvreg**: Regularization paths for SCAD and MCP penalized regression models. URL https://cran.r-project.org/web/packages/ncvreg/

[5] Brent, R. (1973). Algorithms for Minimization without Derivatives. Englewood Cliffs, NJ: Prentice- Hall.

[6] Butler, N. and Denham, M. (2000). The peculiar shrinkage properties of partial least squares regression. Journal of the Royal Statistical Society, Series B, 62: 585–594.

[7] Chen, J. and Chen, Z. (2008). Extended Bayesian information criteria for model selection with large model spaces. Biometrika 95, 759–771.

[8] Efron, B. (2004). The estimation of prediction error: covariance penalties and cross-validation. Journal of the American Statistical Association, 99: 619–633.

[9] Frank, I. and Friedman, J. (1993). A statistical view of some chemometrics regression tools. Technometrics, 35: 109–135.

[10] Friedman, J. H. (1991). Multivariate adaptive regression splines. Annals of Statistics, 19: 1–67.

[11] Golub, G. H., Heath, M., and Wahba, G. (1979). Generalized cross-validation as a method for choosing a good ridge parameter. Technometrics, 21(2): 215–223.

[12] Hadi, A. S. and Ling, R. F. (1998). Some cautionary notes on the use of principal components regression. The American Statistician, 52: 15–19.

[13] Hastie, T., Tibshirani, R., and Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition.

[14] Hwang, J. T. and Nettleton, D. (2003). Principal components regression with data-chosen components and related methods. Technometrics, 45: 70–79.

[15] Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. Technometrics, 12: 55–67.

[16] Ishwaran, H. and Rao, J. S. (2014). Geometry and properties of generalized ridge regression in high dimensions. Contemporary Mathematics, 622: 81–93.

[17] Jollife, I. T. (1982). A note on the use of principal components in regression. Applied Statistics, 31: 300–303.

[18] Kaiser, H. F. (1958). The varimax criterion for analytic rotation in factor analysis. Psychometrika, 23: 187–200.

[19] Kr¨amer, N. and Sugiyama, M. (2011). The degrees of freedom of partial least squares regression. Journal of the American Statistical Association, 106: 697–705.

[20] Lee, M. H. and Liu, Y. (2013). Kernel continuum regression. Computational Statistics & Data Anal- ysis, 68: 190–201.Leisch, F. and Dimitriadou, E. (2012). R Package **mlbench**: Machine learning benchmark problems. URL https://cran.r-project.org/web/packages/mlbench/

[21] Massy, W. F. (1965). Principal components regression in exploratory statistical research. Journal of the American Statistical Association, 60: 234–256.

[22] Mevik, B.-H. and Wehrens, R. (2007). The pls package: Principal component and partial least squares regression in R. Journal of Statistical Software, 18(2). URL: https://www.jstatsoft.org/article/view/v018i02

[23] Mullen, K. M. (2014). Continuous global optimization in R. Journal of Statistical Software, 60(6). URL https://www.jstatsoft.org/article/view/v060i06/v60i06.pdf

[24] Ni, L. (2011). Principal component regression revisited. Statistica Sinica, 21: 741–747.

[25] R Core Team (2018). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

[26] Rosipal, R., Trejo, L. J., and Cichoki, A. (2001). Kernel principal component regression with EM constructed approach to nonlinear principal component extraction. Technical Report, University of Paisley, UK.

[27] Rosipal, R. and Trejo, L. J. (2002). Kernel partial least squares regression in reproducing kernel Hilbert space. Journal of Machine Learning Research, 2: 97–123.

[28] Schwarz, G. (1978). Estimating the dimension of a model. The Annals of Statistics, 6: 461–464.

[29] Shibata, R. (1981). An optimal selection of regression variables. Biometrika, 68: 45–54.

[30] Stone, M. and Brooks, R. J. (1990). Continuum regression: Cross-validated sequentially constructed prediction embracing ordinary least squares, partial least squares and principal components regression. Journal of the Royal Statistical Society, Series B, 52: 237–269.

[31] Su, X. (2015).Variable selection via subtle uprooting. Journal of Computational and Graphical Statistics, 24: 1092–1113.

[32] Su, X., Wijayasinghe, C. S., Fan, J., and Zhang, Y. (2016). Sparse estimation of Cox proportional hazards models via approximated information criteria. Biometrics, 72: 751–759.

[33] Su, X., Fan, J., Levine, R., Nunn, M., and Tsai, C.-L. (2018). Sparse Estimation of Generalized Linear Models (GLM) via Approximated Information Criteria. Statistica Sinica, 28: 1561–1581.

[34] Tibshirani, R. J. (1996). Regression shrinkage and selection via the LASSO. Journal of the Royal Statistical Society, Series B, 58: 267–288.

[35] Tsallis, C. and Stariolo, D. A. (1996). Generalized simulated annealing. Physica A, 233: 395–406.

[36] Wang, H., Li, B., and Leng, C. (2009). Shrinkage tuning parameter selection with a diverging number of parameters. Journal of the Royal Statistical Society, Series B, 71: 671–683.

[37] Wold, H. (1966). Estimation of principal components and related models by iterative least squares. In Multivariate Analysis (Ed. P.R. Krishnaiaah), pp. 391–420. New York, NY: Academic Press.

[38] Wold, H. (1984). PLS regression. In Encyclopedia of Statistical Sciences (eds N. L. Johnson and S. Kotz), vol. 6, pp. 581–591. New York, NY: Wiley.

[39] Xiang, Y., Gubian, S., Suomela, B., and Hoeng, J. (2013). Generalized simulated annealing for global optimization: The GenSA package. The R Journal, 5(1). URL https://journal.r-project.org/archive/2013/RJ-2013-002/RJ-2013-002.pdf

[40] Yang, Y. (2005). Can the strengths of AIC and BIC be shared? A conflict between model identification and regression estimation. Biometrika, 92: 937–950.