The Degrees of Freedom of Partial Least Squares Regression

Nicole Krämer
Weierstrass Institute Berlin
nicole.kraemer@wias-berlin.de

Masashi Sugiyama
Tokyo Institute of Technology
sugi@cs.titech.ac.jp
http://sugiyama-www.cs.titech.ac.jp/~sugi

Abstract

The derivation of statistical properties for Partial Least Squares regression can be a challenging task. The reason is that the construction of latent components from the predictor variables also depends on the response variable. While this typically leads to good performance and interpretable models in practice, it makes the statistical analysis more involved. In this work, we study the intrinsic complexity of Partial Least Squares Regression. Our contribution is an unbiased estimate of its Degrees of Freedom. It is defined as the trace of the first derivative of the fitted values, seen as a function of the response. We establish two equivalent representations that rely on the close connection of Partial Least Squares to matrix decompositions and Krylov subspace techniques. We show that the Degrees of Freedom depend on the collinearity of the predictor variables: The lower the collinearity is, the higher the Degrees of Freedom are. In particular, they are typically higher than the naive approach that defines the Degrees of Freedom as the number of components. Further, we illustrate how our Degrees of Freedom estimate can be used for the comparison of different regression methods. In the experimental section, we show that our Degrees of Freedom estimate in combination with information criteria is useful for model selection.

Keywords

Regression, Model selection, Partial Least Squares, Degrees of Freedom

1 Introduction

Partial Least Squares regression (PLSR) (Wold, 1975) is a two-step regularized regression technique. It iteratively constructs an orthogonal set of latent components from the predictor variables which have maximal covariance with the response variable. This low-dimensional representation of the data is then used to fit a linear regression model. PLSR is extended to nonlinear regression problems via a transformation of the predictor variables (Durand and Sabatier, 1997; Rosipal and Trejo, 2001).

For model selection in PLSR, the optimal number of components has to be determined. While cross-validation is the standard approach, an alternative is the use of information criteria, which use the complexity of the fitted model. In regression, the complexity of a fitting method is defined in terms of its Degrees of Freedom. Apart from their usefulness for model selection, Degrees of Freedom also quantify the intrinsic complexity of a regression method (see e.g. Van der Voet (1999) for an overview). In contrast to other standard regression techniques as Principal Components Regression or Ridge Regression (where the Degrees of Freedom equal the trace of the hat-matrix), the derivation of the Degrees of Freedom for PLSR is not straightforward. This is due to the fact that PLSR is not linear in the sense that the fitted response does not depend linearly on the response. As the set of latent components is constructed in a supervised fashion, the projection of the response variable onto these components is a highly nonlinear function of the response. Therefore, it has been argued (e.g Martens and Naes (1989); Frank and Friedman (1993)) that the Degrees of Freedom of PLSR exceed the number of components.

We provide an unbiased estimate of the generalized Degrees of Freedom of PLSR. It is defined as the trace of the Jacobian matrix of the fitted values, seen as a function of the response. We illustrate on benchmark data that the complexity of PLSR depends on the collinearity of the predictor variables: The higher the collinearity is, the lower the complexity is. Under additional assumptions on the collinearity structure of the data, we provide bounds for the Degrees of Freedom if one component is used.

We present two different implementations. (i) The first one is derived via an iterative computation of the first derivative of the PLSR fit. To do so, we use the equivalence of Partial Least Squares Regression to the Lanczos decomposition of the matrix of predictor variables. This implementation has the advantage that it also provides an asymptotic distribution of the PLSR regression coefficients, which can be used for the construction of confidence intervals (Denham, 1997; Phatak et al., 2002). (ii) The second implementation computes the Degrees of Freedom directly, i.e. it avoids the computation of the derivative itself. This leads to a more favorable runtime. For the derivation, we use the close connection of PLS regression to Krylov subspace techniques. Both algorithms are implemented in the R package ‘plsdof’ (Krämer and Braun, 2009).

We investigate the performance of the Degrees of Freedom of PLSR with respect to model selection of the number of PLSR components. We compare the test errors based on 10-fold cross-validation and based on the Bayesian Information Criterion (BIC) (Schwarz, 1978) in a simulation study. For the latter information criterion, we use our Degrees of Freedom estimate and the naive approach that defines the Degrees of Freedom of PLSR via the number of components. Our experiments show that the model selected based on our Degrees of Freedom estimate is typically less complex than the model selected by the naive approach. This can lead to a higher test error for the naive approach. In terms of prediction accuracy, our Degrees of Freedom approach is on a par with the gold-standard of cross-validation, providing further evidence that our estimates captures the true model complexity correctly.

2 Methodological Background

We consider a multivariate regression problem

$$\mathbb{R} \ni Y_i \quad = \quad f(\boldsymbol{x}_i) + \varepsilon_i \,, \, \varepsilon_i \sim \mathcal{N}\left(0, \sigma^2\right) \,, \tag{1}$$

and the task is to estimate the unknown function $f : \mathbb{R}^p \to \mathbb{R}$ from a finite set of n examples $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n) \in \mathbb{R}^p \times \mathbb{R}$, where y_i is drawn from (1). We assume that the error variables ε_i are independent. Let us denote by $\overline{\boldsymbol{x}}$ and $s(\boldsymbol{x})$ the mean and the standard deviation of the predictor examples $\boldsymbol{x}_i$ and by $\overline{\boldsymbol{y}}$ the mean of the response samples y_i. The $n \times p$ data matrix $\boldsymbol{X}$ is the matrix whose rows are the centered and scaled (to unit variance) $\boldsymbol{x}_i$, and the vector $\boldsymbol{y} \in \mathbb{R}^n$ consists of the centered response y_i. While in the course of this paper, we assume that the regression function f is linear,

$$f(\boldsymbol{x}) \quad = \quad \beta^{(0)} + \boldsymbol{\beta}^\top \boldsymbol{x} \,, \, \beta^{(0)} \in \mathbb{R}, \boldsymbol{\beta} \in \mathbb{R}^p \,, \tag{2}$$

the definitions given in Subsection 2.1 do not require f to be a linear function. Finally, we define

$$\boldsymbol{S} = \frac{1}{n-1} \boldsymbol{X}^\top \boldsymbol{X} \in \mathbb{R}^{p \times p} \quad \text{and} \quad \boldsymbol{s} = \frac{1}{n-1} \boldsymbol{X}^\top \boldsymbol{y} \in \mathbb{R}^p. \tag{3}$$

2.1 Degrees of Freedom and Model Selection

Regularized regression methods typically yield a set of estimates $\widehat{f}_\lambda$ of the true regression function f, and the parameter λ determines the amount of regularization. In Partial Least Squares Regression, the parameter λ corresponds to the number of latent components.

The task is to determine the optimal parameter value λ. Information criteria are based on the rationale that the true error of $\widehat{f}_\lambda$ can be estimated in terms of its training error and its complexity. In regression problems, the complexity is defined via Degrees of Freedom. These are defined for the class of methods that are linear in the sense that the fitted values are a linear function of $\boldsymbol{y}$, i.e. $\widehat{\boldsymbol{y}}_\lambda = \boldsymbol{H}_\lambda \boldsymbol{y}$ with $\boldsymbol{H}_\lambda \in \mathbb{R}^{p \times p}$ a matrix that does not depend on $\boldsymbol{y}$. Popular examples are Ridge Regression and Principal Components Regression. The matrix $\boldsymbol{H}_\lambda$ is called the hat-matrix. In the linear case, the Degrees of Freedom are defined as the trace of the hat-matrix,

$$\mathrm{DoF}(\lambda) \quad = \quad \mathrm{trace}\left(\boldsymbol{H}_\lambda\right) \,. \tag{4}$$

As we point out below, PLS regression is not a linear method, and the above definition cannot be applied. In order to extend the notion of Degrees of Freedom to PLSR, we employ the generalized definition proposed by Efron (2004).

Definition 1. *Let $\widehat{f}_\lambda$ be an estimate of the true regression function f, parameterized by λ. We define the vector of fitted values as $\widehat{\boldsymbol{y}}_\lambda = (\widehat{f}_\lambda(\boldsymbol{x}_1,), \ldots, \widehat{f}_\lambda(\boldsymbol{x}_n))^\top$. The Degrees of Freedom are*

$$\mathrm{DoF}\,(\lambda) \quad = \quad \mathbb{E}\left[\mathrm{trace}\left(\frac{\partial \widehat{\boldsymbol{y}}_\lambda}{\partial \boldsymbol{y}}\right)\right] .$$

Here, the input $\boldsymbol{X}$ is assumed to be fixed and the expectation $\mathbb{E}$ is taken with respect to $y_1, \ldots, y_n$.

The Degrees of Freedom measure the sensitivity of the fitted values, seen as a function of $\boldsymbol{y}$. Note that for the special case of linear methods, this definition coincides with (4).

Popular examples of information criteria include the Akaike information criterion (Akaike, 1973), the generalized minimum description length (Hansen and Yu, 2001) and the Bayesian information criterion (BIC) (Schwarz, 1978)

$$\mathrm{bic}\,(\lambda) \quad = \quad \left\|\widehat{\boldsymbol{y}}_\lambda - \boldsymbol{y}\right\|^2 + \log(n)\sigma^2\,\mathrm{DoF}(\lambda) .$$

For information criteria, we need an estimate of the noise level σ defined in (1). For linear methods $\widehat{\boldsymbol{y}}_\lambda = \boldsymbol{H}_\lambda \boldsymbol{y}$, this is accomplished as follows. The residual is of the form $\boldsymbol{y} - \widehat{\boldsymbol{y}}_\lambda = (\boldsymbol{I}_n - \boldsymbol{H}_\lambda)\,\boldsymbol{y}$. The bias-variance decomposition of the mean squared error yields

$$E\left[\|\boldsymbol{y} - \widehat{\boldsymbol{y}}_\lambda\|^2\right] \quad = \quad \left\|E[\boldsymbol{y} - \widehat{\boldsymbol{y}}_\lambda]\right\|^2 + \mathrm{trace}\left((\boldsymbol{I}_n - \boldsymbol{H}_\lambda)(\boldsymbol{I}_n - \boldsymbol{H}_\lambda^\top)\right)\sigma^2 .$$

By dropping the unknown bias term $\left\|E[\boldsymbol{y} - \widehat{\boldsymbol{y}}_\lambda]\right\|^2$, we yield an estimate of σ via

$$\widehat{\sigma}^2_* \quad = \quad \frac{\|\boldsymbol{y} - \widehat{\boldsymbol{y}}_\lambda\|^2}{\mathrm{trace}\left((\boldsymbol{I}_n - \boldsymbol{H}_\lambda)(\boldsymbol{I}_n - \boldsymbol{H}_\lambda^\top)\right)} . \tag{5}$$

If $\boldsymbol{H}_\lambda$ is a projection operator (which is e.g. true for Principal Components Regression), the expression is simplified to

$$\widehat{\sigma}^2 \quad = \quad \frac{\|\boldsymbol{y} - \widehat{\boldsymbol{y}}_\lambda\|^2}{n - \mathrm{DoF}(\lambda)} . \tag{6}$$

We note that the latter estimate is most commonly used, even if the above assumption is not fulfilled.

2.2 Partial Least Squares Regression

PLSR constructs m latent components $\boldsymbol{T} = (\boldsymbol{t}_1, \ldots, \boldsymbol{t}_m) \in \mathbb{R}^{n\times m}$ from the predictor variables $\boldsymbol{X}$ such that the components $\boldsymbol{t}_i$ are mutually orthogonal and that they have maximum covariance to the response $\boldsymbol{y}$. In the NIPALS algorithm (Wold, 1975), the first component $\boldsymbol{t}_1 = \boldsymbol{X}\boldsymbol{w}_1$ maximizes the squared covariance to the response $\boldsymbol{y}$,

$$\boldsymbol{w}_1 \;=\; \arg\max_{\boldsymbol{w}} \frac{\|\mathrm{cov}(\boldsymbol{X}\boldsymbol{w},\boldsymbol{y})\|^2}{\boldsymbol{w}^\top\boldsymbol{w}} = \arg\max_{\boldsymbol{w}} \frac{\boldsymbol{w}^\top\boldsymbol{X}^\top\boldsymbol{y}\boldsymbol{y}^\top\boldsymbol{X}\boldsymbol{w}}{\boldsymbol{w}^\top\boldsymbol{w}} \propto \boldsymbol{X}^\top\boldsymbol{y}\,. \tag{7}$$

Subsequent components $\boldsymbol{t}_2, \boldsymbol{t}_3, \ldots$ are chosen such that they maximize the squared covariance to $\boldsymbol{y}$ and that all components are mutually orthogonal. Orthogonality is enforced by deflating the original variables $\boldsymbol{X}$,

$$\boldsymbol{X}_i \;=\; \boldsymbol{X} - \mathcal{P}_{\boldsymbol{t}_1,\ldots,\boldsymbol{t}_{i-1}}\boldsymbol{X}\,. \tag{8}$$

Here, $\mathcal{P}_{\boldsymbol{t}_1,\ldots,\boldsymbol{t}_{i-1}}$ denotes the orthogonal projection onto the space spanned by $\boldsymbol{t}_1,\ldots,\boldsymbol{t}_{i-1}$. We then replace $\boldsymbol{X}$ by $\boldsymbol{X}_i$ in (7). While the matrix $\boldsymbol{T} = (\boldsymbol{t}_1,\ldots,\boldsymbol{t}_m)$ is orthogonal by construction, it can be shown that the matrix $\boldsymbol{W} = (\boldsymbol{w}_1,\ldots,\boldsymbol{w}_m) \in \mathbb{R}^{d\times m}$ is orthogonal as well (e.g. Hoskuldsson (1988)). The m latent components $\boldsymbol{T}$ are used as regressors in a least squares fit in place of $\boldsymbol{X}$, leading to fitted values

$$\widehat{\boldsymbol{y}}_m \;=\; \overline{\boldsymbol{y}}\boldsymbol{1}_n + \boldsymbol{T}\left(\boldsymbol{T}^\top\boldsymbol{T}\right)^{-1}\boldsymbol{T}^\top\boldsymbol{y} = \overline{\boldsymbol{y}}\boldsymbol{1}_n + \mathcal{P}_{\boldsymbol{T}}\boldsymbol{y}\,. \tag{9}$$

We emphasize again that PLSR is not a linear estimator as defined in Section 2.1. The projection matrix $\mathcal{P}_{\boldsymbol{T}}$ depends on the response as well, as the latent components $\boldsymbol{T}$ are defined in terms of both $\boldsymbol{X}$ and $\boldsymbol{y}$.

To determine the estimated regression coefficients and intercept in (2), we define

$$\boldsymbol{L} \;=\; \boldsymbol{T}^\top\boldsymbol{X}\boldsymbol{W} \in \mathbb{R}^{m\times m}\,, \tag{10}$$

and obtain $\widehat{\boldsymbol{\beta}}_m = \boldsymbol{D}\boldsymbol{W}\boldsymbol{L}^{-1}\boldsymbol{T}^\top\boldsymbol{y}$ for the regression coefficients (Manne, 1987; Hoskuldsson, 1988) and $\widehat{\beta}_m^{(0)} = \overline{\boldsymbol{y}} - \overline{\boldsymbol{x}}^\top\widehat{\boldsymbol{\beta}}_m$ for the intercept. Here, $\boldsymbol{D}$ is the diagonal $p\times p$ scaling matrix with entries $d_{ii} = 1/s(\boldsymbol{x})_i$.

3 Unbiased Estimation of the Degrees of Freedom

The latent components $\boldsymbol{T}$ of PLSR depend on the response $\boldsymbol{y}$. Therefore, the relationship between $\boldsymbol{y}$ and the fitted PLSR values $\widehat{\boldsymbol{y}}_m$ is nonlinear, and the compact formula (4) for the Degrees of Freedom cannot be applied. However, we can use the more general Definition 1 to obtain an unbiased plug-in estimate.

Proposition 2. *An unbiased estimate of the Degrees of Freedom of PLSR with m latent components $\boldsymbol{T} = (\boldsymbol{t}_1,\ldots,\boldsymbol{t}_m)$ is given by*

$$\widehat{\mathrm{DoF}}(m) \;=\; 1 + \mathrm{trace}\left(\frac{\partial \mathcal{P}_{\boldsymbol{T}}\boldsymbol{y}}{\partial \boldsymbol{y}}\right). \tag{11}$$

The constant term 1 corresponds to the estimation of the intercept $\beta^{(0)}$, which consumes one Degree of Freedom. For the derivation, we need to compute the trace of the derivative in (11) explicitly. We propose two equivalent algorithms in Subsections 3.3 and 3.4.

Table 1: Properties of the three benchmark data sets. A detailed description of the data can be found in the appendix.

data set	variables	examples	mean absolute correlation
`kin (fh)`	32	8192	low (0.009)
`ozone`	12	203	medium (0.260)
`cookie`	700	70	high (0.867)

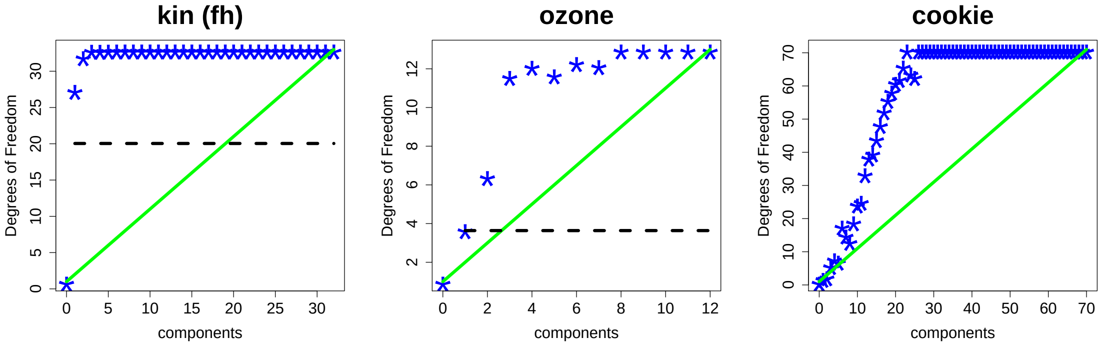

Figure 1: Estimated Degrees of Freedom (stars) for the three benchmark data sets. The solid line displays the naive estimate $\mathrm{DoF}(m) = m + 1$. If the assumption of theorem 3 is fulfilled, we also display the lower bound on the Degrees of Freedom for 1 component (dashed horizontal line).

3.1 Illustration and a Lower Bound

Before delving deeper into the details of the implementation, we illustrate the properties of the Degrees of Freedom on benchmark data. An overview of the data sets is given in Table 1 (see the appendix for more details). We choose the three particular data sets as they differ with respect to the collinearity structure of the predictor variables. As an indicator for the degree of collinearity, we compute the mean of the absolute empirical correlation coefficients defined by $\overline{s} = (2/(p^2 - p)) \sum_{i<j}^{p} |s_{ij}|$. Here s_{ij} is the (i, j)-entry of the empirical correlation matrix $\boldsymbol{S}$ of $\boldsymbol{X}$. The values of $\overline{s}$ are displayed in the fourth column of Table 1.

In Figure 1, we plot the Degrees of Freedom for the three data sets as a function of the number of components. (For the large data set `kin (fh)`, we use a subsample of size 300.) In addition, we display the naive estimate $\mathrm{DoF}(m) = m + 1$. In the examples, our Degrees of Freedom estimate always exceeds the naive approach. This supports the conjecture that $\mathrm{DoF}(m) \geq m + 1$, which is voiced e.g. in Martens and Naes (1989) and Frank and Friedman (1993).

Furthermore, Figure 1 shows that the correlation structure of the predictor variables determines the shape of the DoF-curve. In our examples, the complexity is higher for data with low correlation. We underpin this observation with a lower bound on the Degrees of Freedom of PLSR with $m = 1$ component.

Theorem 3. *If the largest eigenvalue $\lambda_{\max}$ of the empirical correlation matrix $\boldsymbol{S}$ defined in* (3) *fulfills*

$$\lambda_{\max} \;\leq\; \frac{1}{2}\mathit{trace}(\boldsymbol{S})\,, \tag{12}$$

then

$$\widehat{\mathrm{DoF}}(m=1) \;\geq\; 1 + \frac{\mathit{trace}(\boldsymbol{S})}{\lambda_{\max}}\,. \tag{13}$$

Condition (12) controls the amount of collinearity of $\boldsymbol{X}$. If the collinearity is low, the decay of the eigenvalues of $\boldsymbol{S}$ is slow (and condition (12) is fulfilled). The lower bound (13) is higher for data with low collinearity. In Figure 1, we add the lower bound for the data sets `kin (fh)` and `ozone`, which fulfill condition (12).

Proof. We express the PLS fit for one component in terms of $\boldsymbol{S}$ and $\boldsymbol{s}$ defined in (3). Recall that the first latent component is defined as $\boldsymbol{t}_1 = \boldsymbol{X}\boldsymbol{s}$, which implies

$$\widehat{\boldsymbol{y}}_1 \;=\; \overline{\boldsymbol{y}} + \frac{\boldsymbol{s}^\top\boldsymbol{s}}{\boldsymbol{s}^\top\boldsymbol{S}\boldsymbol{s}}\boldsymbol{X}\boldsymbol{s}\,.$$

After computing the derivative of this term with respect to $\boldsymbol{y}$ and computing its trace, we obtain

$$\widehat{\mathrm{DoF}}(m=1) \;=\; 3 + \frac{\boldsymbol{s}^\top\boldsymbol{s}}{\boldsymbol{s}^\top\boldsymbol{S}\boldsymbol{s}}\left[\mathrm{trace}(\boldsymbol{S}) - 2\frac{\left(\boldsymbol{s}^\top\boldsymbol{S}^2\boldsymbol{s}\right)}{\boldsymbol{s}^\top\boldsymbol{S}\boldsymbol{s}}\right]\,.$$

Now, by definition of maximal eigenvalues, $\boldsymbol{s}^\top\boldsymbol{S}^2\boldsymbol{s}/\boldsymbol{s}^\top\boldsymbol{S}\boldsymbol{s} \leq \lambda_{max}$ and $\boldsymbol{s}^\top\boldsymbol{s}/\boldsymbol{s}^\top\boldsymbol{S}\boldsymbol{s} \geq 1/\lambda_{max}$. It follows that

$$\mathrm{trace}(\boldsymbol{S}) - 2\frac{\left(\boldsymbol{s}^\top\boldsymbol{S}^2\boldsymbol{s}\right)}{\boldsymbol{s}^\top\boldsymbol{S}\boldsymbol{s}} \;\geq\; \mathrm{trace}(\boldsymbol{S}) - 2\lambda_{\max}\,.$$

Condition (12) ensures that the right-hand side of this inequality is non-negative, hence,

$$\begin{aligned}\widehat{\mathrm{DoF}}(m=1) \;&\geq\; 3 + \frac{\boldsymbol{s}^\top\boldsymbol{s}}{\boldsymbol{s}^\top\boldsymbol{S}\boldsymbol{s}}\left[\mathrm{trace}(\boldsymbol{S}) - 2\lambda_{\max}\right]\\ &\geq\; 3 + \frac{1}{\lambda_{max}}\left[\mathrm{trace}(\boldsymbol{S}) - 2\lambda_{\max}\right] = 1 + \frac{\mathrm{trace}(\boldsymbol{S})}{\lambda_{\max}}\,.\end{aligned}$$

□

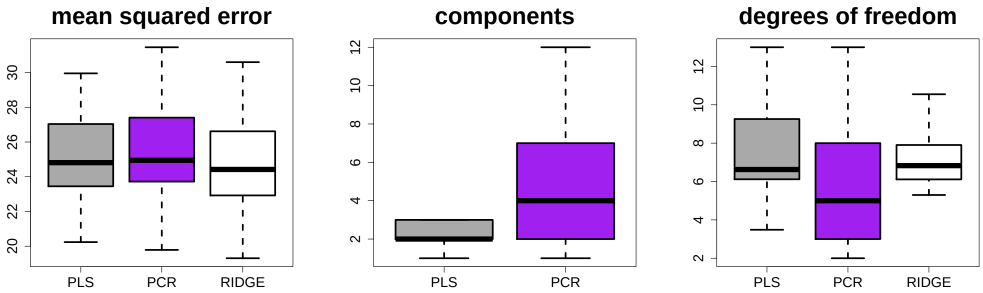

Figure 2: Comparison of PLS, PCR and Ridge Regression. Left: mean squared error of prediction for the three regression methods. Center: Number of cross-validation optimal number of components for PLS and PCR. Right: Cross-validation optimal Degrees of Freedom for the three regression methods.

3.2 Comparison of Regression Methods

While the regularization parameters λ_1 and λ_2 for two competing regression methods cannot be compared directly, their corresponding Degrees of Freedom values $\mathrm{DoF}(\lambda_1)$ and $\mathrm{DoF}(\lambda_2)$ are on the same scale. Hence, Degrees of Freedom allow us to compare the model complexity across different regularized regression approaches.

To illustrate this point, we compare the model complexity of PLSR, Principal Components Regression (PCR), and Ridge Regression on the `ozone` data set. Recall that for PCR, the Degrees of Freedom are the number of principal components plus one, and for Ridge Regression,

$$\mathrm{DoF}(\lambda)_{\text{ridge}} = 1 + \mathrm{trace}\left(\boldsymbol{X}\left(\boldsymbol{X}^\top\boldsymbol{X} + \lambda\boldsymbol{I}_p\right)^{-1}\boldsymbol{X}^\top\right).$$

We split the data set into 50 training examples and 153 test examples. On the training data, we compute the optimal model parameter for the respective methods with 10-fold cross-validation. We assess the predictive performance by computing the mean squared error of prediction on the test set. This procedure is repeated 50 times.

Let us start with the observation that the predictive performance of the three methods is similar on this data set (left plot in Figure 2). This is in accordance with previous observations that the methods often perform similarly (see e.g. Frank and Friedman (1993)). Next, we compare the model complexity of PLSR and PCR in terms of the cross-validation-optimal number of components. This is displayed in the center plot of Figure 2. On average, PLSR selects fewer components than PCR. However, in terms of Degrees of Freedom, PLSR selects more complex models than PCR (right plot in Figure 2).

Further, it is known (De Jong, 1993) that for a fixed number of components, PLSR obtains a lower approximation error $\|\widehat{\boldsymbol{y}}_{ols} - \widehat{\boldsymbol{y}}_m\|^2/n$ than PCR. Here, $\widehat{\boldsymbol{y}}_{ols}$ are the fitted values obtained by ordinary least squares. This implies that for a fixed number of components, the training error $\|\boldsymbol{y} - \widehat{\boldsymbol{y}}_m\|^2/n$ of PLSR is lower than the one of PCR. This is

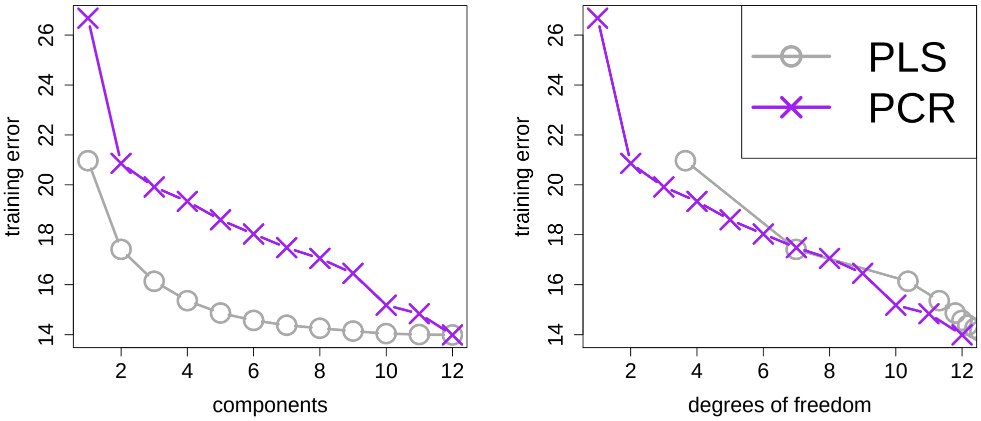

Figure 3: Training error of PLSR and PCR. Left: Training error as a function of the number of components. Right: Training error as a function of the Degrees of Freedom.

illustrated in the left plot of Figure 3. Here, we plot the mean training error for PLSR and PCR over all 50 runs of the experiments as a function of the number of components.

The lower approximation error is sometimes used as an argument in favor of PLSR, as it often leads to the selection of fewer components compared to PCR (as illustrated in Figure 2). While this can be advantageous in problems where the latent components are used for model interpretation (e.g. for the visualization of the data in terms of a 2-D or 3-D plot), we stress again that the number of components do not capture the intrinsic model complexity, and that models with the same Degrees of Freedom should be compared instead. In the right plot of Figure 3, we therefore plot the mean training error of PLSR and PCR as functions of the corresponding Degrees of Freedom. The plot shows that the difference of the approximation error is marginal in terms of Degrees of Freedom. Furthermore, the plot illustrates that PLSR concentrates a lot of components on regions with low training error and high complexity: All PLS models (with $m \geq 1$) consume at least 4 Degrees of Freedom, and for $m \geq 3$, the Degrees of Freedom already exceed 10. Compared to PCR, this might be disadvantageous in situations where the true underlying model has a low complexity, as PLSR models do not explore the full range of Degrees of Freedom.

To summarize, the direct comparison of model parameters for different regression methods is in general not possible, because they are either on a different scale (e.g. for PLSR and Ridge Regression) or they are just seemingly on the same scale (e.g. for PLSR and PCR) and a direct comparison would lead to misleading results. Degrees of Freedom offer a valuable solution to this problem.

3.3 First Derivative of the Lanczos Decomposition

We now present the first implementation of the Degrees of Freedom estimate. We extend the approaches that are proposed in Denham (1997) and Serneels et al. (2004). There, the iterative formulation of the NIPALS algorithm is used to construct the derivative of $\widehat{\boldsymbol{\beta}}_m$. Instead, our algorithm is based on the matrix decomposition defined by (10). Note that the matrix $\boldsymbol{L}$ is upper bidiagonal, i.e. $l_{ij} = 0$ for $i > j$ or $i < j - 1$. The relationship (10) defines a Lanczos decomposition (Lanczos, 1950) of $\boldsymbol{X}$, i.e. a decomposition into orthonormal matrices $\boldsymbol{T}$ and $\boldsymbol{W}$ and an upper bidiagonal matrix $\boldsymbol{L}$. For fixed $\boldsymbol{y}$ and m, the decomposition is unique. The Lanczos decomposition can be interpreted as the analogue to the singular value decomposition that is defined by Principal Components Analysis. For more details on the equivalence of PLSR and Lanczos decompositions, we refer the readers to Eldén (2004).

We propose an implementation of PLSR that iteratively constructs the derivative of the projection operator $\mathcal{P}_T$ based on the Lanczos decomposition. This has the advantage that we also obtain the derivative of the regression coefficients $\widehat{\boldsymbol{\beta}}_m$, which can then be used to construct approximative confidence intervals (Denham, 1997; Phatak et al., 2002).

We proceed in three steps. First, we derive a fast recursive PLSR algorithm based on the Lanczos decomposition. This algorithm avoids the explicit deflation of $\boldsymbol{X}$ as in (8), and only depends on projections onto one-dimensional subspaces. Second, we determine the first derivative of one-dimensional projection operators (Krämer and Braun, 2007), which are

$$\frac{\partial\left(\boldsymbol{v}/\|\boldsymbol{v}\|_{\boldsymbol{S}}\right)}{\partial \boldsymbol{y}} := \frac{\partial\left(\boldsymbol{v}/\sqrt{\boldsymbol{v}^\top \boldsymbol{S}\boldsymbol{v}}\right)}{\partial \boldsymbol{y}} = \frac{1}{\|\boldsymbol{v}\|_{\boldsymbol{S}}}\left(\boldsymbol{I}_n - \frac{\boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{S}}{\boldsymbol{v}^\top \boldsymbol{v}}\right)\frac{\partial \boldsymbol{v}}{\partial \boldsymbol{y}},$$

and

$$\frac{\partial\left(\boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{z}\right)}{\partial \boldsymbol{y}} = \left(\boldsymbol{v}\boldsymbol{z}^\top + \boldsymbol{v}^\top \boldsymbol{z}\boldsymbol{I}_n\right)\frac{\partial \boldsymbol{v}}{\partial \boldsymbol{y}} + \boldsymbol{v}\boldsymbol{v}^\top \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{y}}.$$

Finally, we differentiate the recursive formulas of the Lanczos representation. Algorithm 1 displays the result. Its derivation can be found in the appendix.

As we compute the derivative of the regression coefficients as well, we can estimate the covariance of the PLSR coefficients by using a first order Taylor approximation $\widehat{\boldsymbol{\beta}}_m \approx \left(\partial\widehat{\boldsymbol{\beta}}_m/\partial \boldsymbol{y}\right)\boldsymbol{y}$, which leads to

$$\widehat{\text{cov}}\left(\widehat{\boldsymbol{\beta}}_m\right) = \widehat{\sigma}^2 \frac{\partial \widehat{\boldsymbol{\beta}}_m}{\partial \boldsymbol{y}}\left(\frac{\partial \widehat{\boldsymbol{\beta}}_m}{\partial \boldsymbol{y}}\right)^\top.$$

Furthermore, we can use the first order Taylor expansion to construct an approximate hat-matrix for PLSR via

$$\boldsymbol{H}_{m,\,\text{approx}} = \frac{\partial \widehat{\boldsymbol{y}}_m}{\partial \boldsymbol{y}}. \tag{14}$$

This matrix can be plugged into formula (5) for the estimation of the noise level.

Algorithm 1 Derivative of the regression coefficients and Degrees of Freedom

```
Input: centered and scaled data $\boldsymbol{X}$, $\boldsymbol{y}$, number $m$ of components
$n =$ number or examples
$\boldsymbol{S} = \left(\boldsymbol{X}^\top \boldsymbol{X}\right)/(n-1)$, $\boldsymbol{s} = \left(\boldsymbol{X}^\top \boldsymbol{y}\right)/(n-1)$
Initialization: $\widehat{\boldsymbol{\beta}}_0 = \boldsymbol{0}_p$, $\left(\partial \widehat{\boldsymbol{\beta}}_0 / \partial \boldsymbol{y}\right) = \boldsymbol{0}_{p\times n}$
for $i = 1, \dots, m$ do
    $\boldsymbol{w}_i = \boldsymbol{s} - \boldsymbol{S}\widehat{\boldsymbol{\beta}}_{i-1}$
        $(\partial \boldsymbol{w}_i / \partial \boldsymbol{y}) = \boldsymbol{X}^\top/(n-1) - \boldsymbol{S}\left(\partial \widehat{\boldsymbol{\beta}}_{i-1}/\partial \boldsymbol{y}\right)$
    $\boldsymbol{v}_i = \boldsymbol{w}_i - \sum_{j=1}^{i-1} \boldsymbol{v}_j \boldsymbol{v}_j^\top \boldsymbol{S}\boldsymbol{w}_i$
        $(\partial \boldsymbol{v}_i/\partial \boldsymbol{y}) = (\partial \boldsymbol{w}_i/\partial \boldsymbol{y}) - \sum_{j=1}^{i-1} \left(\partial \boldsymbol{v}_j \boldsymbol{v}_j^\top \boldsymbol{S}\boldsymbol{w}_i / \partial \boldsymbol{y}\right)$
    $\boldsymbol{v}_i = \sqrt{n-1}\boldsymbol{v}_i/\|\boldsymbol{v}_i\|_{\boldsymbol{S}}$
        $(\partial \boldsymbol{v}_i/\partial \boldsymbol{y}) = \sqrt{n-1}\partial\left(\boldsymbol{v}_i/\|\boldsymbol{v}_i\|_{\boldsymbol{S}}\right)/\partial \boldsymbol{y}$
    $\widehat{\boldsymbol{\beta}}_i = \widehat{\boldsymbol{\beta}}_{i-1} + \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{s}$
        $\partial \widehat{\boldsymbol{\beta}}_i / \partial \boldsymbol{y} = \partial \widehat{\boldsymbol{\beta}}_{i-1}/\partial \boldsymbol{y} + \left(\partial \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{s}/\partial \boldsymbol{y}\right)$
end for
$\mathrm{DoF}(m) = 1 + \mathrm{trace}\left(\boldsymbol{X}\partial \widehat{\boldsymbol{\beta}}_m / \partial \boldsymbol{y}\right)$.
```

3.4 Trace of the Krylov Representation

The computation of the derivative of $\widehat{\boldsymbol{y}}_m$ in Subsection 3.3 involves repeated matrix-matrix-multiplications. For high-dimensional data, this can become very time-consuming. As we do not need the derivative itself for the Degrees of Freedom but only its trace, we reduce the computational load by cleverly rearranging the computation of the derivative.

To this end, we use a closed form expression of the fitted values $\widehat{\boldsymbol{y}}_m$ that is based on Krylov subspaces. We use the fact (Hoskuldsson, 1988) that $\mathrm{span}\{\boldsymbol{t}_1, \dots, \boldsymbol{t}_m\} = \mathrm{span}\{\boldsymbol{K}\boldsymbol{y}, \dots, \boldsymbol{K}^m\boldsymbol{y}\} =: \mathcal{K}_m$, with $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{X}^\top$ the $n \times n$ kernel matrix. The space on the right-hand side is called the Krylov subspace defined by $\boldsymbol{K}$ and $\boldsymbol{K}\boldsymbol{y}$. We use the explicit representation $\widehat{\boldsymbol{y}}_m = \overline{\boldsymbol{y}} + \mathcal{P}_{\mathcal{K}_m}$ to compute its derivative. In Phatak et al. (2002), a corresponding formula for $\widehat{\boldsymbol{\beta}}_m$ is used to determine its approximate distribution. We extend this result to the derivative of the fitted values. Additionally, after computing the derivative, we apply the basis transformation $\boldsymbol{B} = (\langle \boldsymbol{t}_i, \boldsymbol{K}^j\boldsymbol{y}\rangle) \in \mathbb{R}^{m\times m}$ to improve numerical stability. This yields the following result.

Proposition 4. *Set*

$$\mathrm{c} = \boldsymbol{B}^{-1}\boldsymbol{T}^\top\boldsymbol{y} \in \mathbb{R}^m \quad \textit{and} \quad \boldsymbol{V} = (\boldsymbol{v}_1, \dots, \boldsymbol{v}_m) = \boldsymbol{T}\left(\boldsymbol{B}^{-1}\right)^\top \in \mathbb{R}^{n\times m}\,.$$

We have

$$\frac{\partial \widehat{\boldsymbol{y}}_m}{\partial \boldsymbol{y}} \;=\; \frac{1}{n}\boldsymbol{I}_n + \sum_{j=1}^{m} c_j\left(\boldsymbol{I}_n - \boldsymbol{T}\boldsymbol{T}^\top\right)\boldsymbol{K}^j + \sum_{j=1}^{m} \boldsymbol{v}_j\left(\boldsymbol{y} - \widehat{\boldsymbol{y}}_m\right)^\top \boldsymbol{K}^j + \boldsymbol{T}\boldsymbol{T}^\top\,.$$

In contrast to the Lanczos representation (Subsection 3.3), the representation in Proposition 4 is more convenient for the computation of the Degrees of Freedom, as its trace can be computed directly.

Proposition 5. *The unbiased estimate for the Degrees of Freedom of PLSR with m components equals*

$$\widehat{\mathrm{DoF}}(m) \;=\; 1+\sum_{j=1}^{m} c_j \operatorname{trace}\left(\boldsymbol{K}^j\right) - \sum_{j,l=1}^{m} \boldsymbol{t}_l^\top \boldsymbol{K}^j \boldsymbol{t}_l + \left(\boldsymbol{y}-\widehat{\boldsymbol{y}}_m\right)^\top \sum_{j=1}^{m} \boldsymbol{K}^j \boldsymbol{v}_j + m\,.$$

Hence, for the computation of the Degrees of Freedom of PLSR, we need a single fit of the PLSR algorithm that returns the matrix $\boldsymbol{T}$ of latent components. One can either use the original formulation of the NIPALS algorithm (Subsection 2.2) or the Lanczos decomposition (Algorithm 1) without the computation of the derivative.

4 Experiments

In this section, we evaluate the performance of the Degrees of Freedom estimate with respect to model selection for PLSR. We investigate the performance both with respect to prediction accuracy and model complexity (Subsection 4.1), and with respect to the estimation of the noise level σ of the regression model (Subsection 4.2). Further, in Subsection 4.3, we discuss numerical issues and the computational efficiency of our proposed algorithms.

In our experiments, we consider the Bayesian Information Criterion. We conducted experiments with the Akaike Information Criterion and the generalized Minimum Description Length as well, but we found that our main observations on the difference between our Degrees of Freedom estimate and the naive approach $\mathrm{DoF}(m) = m+1$ do not depend on the particular criterion. Hence, for the sake of clarity, we only report the results for the Bayesian Information Criterion.

We would like to stress that the primary goal of this section is not to advertise BIC in combination with our Degrees of Freedom estimate as a novel model selection criterion that is universally preferable to existing methods. Instead, we want to provide further empirical evidence that our estimate captures the intrinsic model complexity of PLSR reliably.

We use the `ozone` data set as a starting point for our simulation study. As a preprocessing step, we scale all variables of the data set to lie in the interval [-1,+1]. The true regression function f in equation (1) is of the form

$$f(\boldsymbol{x}) \;=\; \sum_{j=1}^{d} \beta_j \phi_j(\boldsymbol{x}) \;\text{ with fixed basis functions }\; \phi_j(\boldsymbol{x}) = \exp\left(-\|\boldsymbol{x}-\boldsymbol{c}_j\|^2\right)\,.$$

The coefficients of the center points $\boldsymbol{c}_j \in \mathbb{R}^p$ are chosen uniformly from $[-1,+1]$. The regression coefficients β_j are chosen randomly from a uniform distribution over $[1,3]$. We stress that the center points and hence the basis functions are fixed a-priori. This is to ensure that we are still in a parametric regression scenario, for which the Bayesian Information Criterion is suited. In the simulation study, we vary the number d of basis functions from 10 to 210 in steps of 40. We choose the variance σ^2 of the noise variable such that the signal-to-noise-ratio equals 9. After the transformation of the initial data matrix $\boldsymbol{X}$ via the basis functions $(\phi_1, \ldots, \phi_d)$, we obtain a d-dimensional data set, and are in the setting of a linear regression model.

We split the data into 50 training and 153 test points. The small training sample size allows us to consider high-dimensional settings, and the large test sample size ensures a reliable estimation of the test error. On the training data, we apply four different model selection criteria.

(a) **CV**: 10-fold cross-validation.

(b) **LANCZOS**: Bayesian Information Criterion with the Degrees of Freedom computed from the Lanczos decomposition (Algorithm 1). For the estimation of the noise level, we use equation (5) with the approximate hat-matrix defined in (14).

(c) **KRYLOV**: Bayesian Information Criterion with the Degrees of Freedom computed from the Krylov representation (Proposition 5). For the estimation of the noise level, we use equation (6).

(d) **NAIVE**: Bayesian Information Criterion with the naive Degrees of Freedom $\mathrm{DoF}(m) = m + 1$. For the estimation of the noise level, we use equation (6).

Note that LANCZOS and KRYLOV use the same Degrees of Freedom estimate, and only differ in the estimation of the noise level σ . As the computation of the Degrees of Freedom depends on two different implementations, their runtime differs. Further, for all four methods, we set the range of the number of components from 0 to 30.

For each of the four criteria, we measure the performance on the hold-out set of size 153 by computing the normalized mean squared error: We divide the mean squared test error by the mean squared test error of the trivial model, i.e. the constant model equal to the mean of the training data. This normalization facilitates the comparison between different values of d. The procedure is repeated 50 times.

4.1 Prediction Accuracy and Model Complexity

We display the median and the boxplots of normalized test errors in Figure 4. On average, our Degrees of Freedom estimate in combination with BIC shows a comparable prediction accuracy to cross-validation (right plot of Figure 4). The two different approaches for the computation of the Degrees of Freedom (KRYLOV and LANCZOS), which only differ in the estimation of σ, do not show any clear difference over the different simulation settings. Note however that LANCZOS is in general computationally more expensive. We refer the readers to Subsection 4.3 for more details on computation time.

For the higher-dimensional scenarios ($d \geq 90$) in our simulation study, the naive approach leads to considerably worse results than the three other methods (left plot in Figure 4). Figure 5 illustrates that the naive approach tends to underestimate the true underlying complexity of the model. Compared to cross-validation and our DoF estimate, it selects more components and Degrees of Freedom respectively. A closer look at the simulation reveals that the selection of more components does not automatically lead to a higher test error. More precisely, for $d = 50$ dimensions, the naive approach selects considerably more components compared to the three other methods, yet the prediction accuracy is on the same level. In contrast, for higher number of dimensions, the increased model complexity also affects the prediction accuracy. A possible explanation for this phenomenon is the following. For many moderate sized data sets, the test error first decreases sharply with the number of components, and then reaches a flat plateau for higher number of components. In this case, a more complex model can lead to a comparable prediction accuracy even with higher number of components. To underpin this point, we plot a scaled test error for $d = 50$ and $d = 210$ as a function of the number of components (see Figure 6). Here, we scale the test error such that its minimal value for a fixed d is equal to 1. For $d = 50$, the relative decrease in prediction accuracy is only moderate if we choose too many components, and the test error curve is flat. In contrast, for $d = 210$, the relative increase of the test error is steep, and a selection of more components than the test-error optimal ones can lead to poor results.

4.2 Estimation of the Noise Level σ

We now investigate the quality of the three different PLS based estimates for the noise level σ that are obtained by KRYLOV, LANCZOS and NAIVE.

Figure 8 displays normalized estimates $\widehat{\sigma}$ obtained on the 50 training samples: We divide each estimate by the true noise level σ that is determined by the signal-to-noise ratio (which is set to 9). The dashed horizontal line indicates the theoretical optimum of 1. We observe that KRYLOV and LANCZOS slightly overestimate the noise level, which would in turn lead to slightly too conservative confidence intervals when used in inference problems. In contrast, the naive approach underestimates σ, and due to the complex models that it selects for higher dimensions (see Subsection 4.1), the quality of the estimate deteriorates.

4.3 Numerical Stability and Runtime

As explained in Subsection 3.3, the sparse structure of the matrix $\boldsymbol{L}$ defined in (10) allows us to derive a fast iterative algorithm for PLSR and its derivative (Algorithm 1). In practice, we observe that the sparsity leads to numerical problems: After a certain number of components, the latent components $\boldsymbol{t}_i$ are not mutually orthogonal anymore. This typically affects the computation of the Degrees of Freedom as well and leads to implausible results (e.g. negative Degrees of Freedom). In Krämer and Braun (2007), the sparse structure of $\boldsymbol{L}$ is used and an additional stopping criterion is imposed to ensure

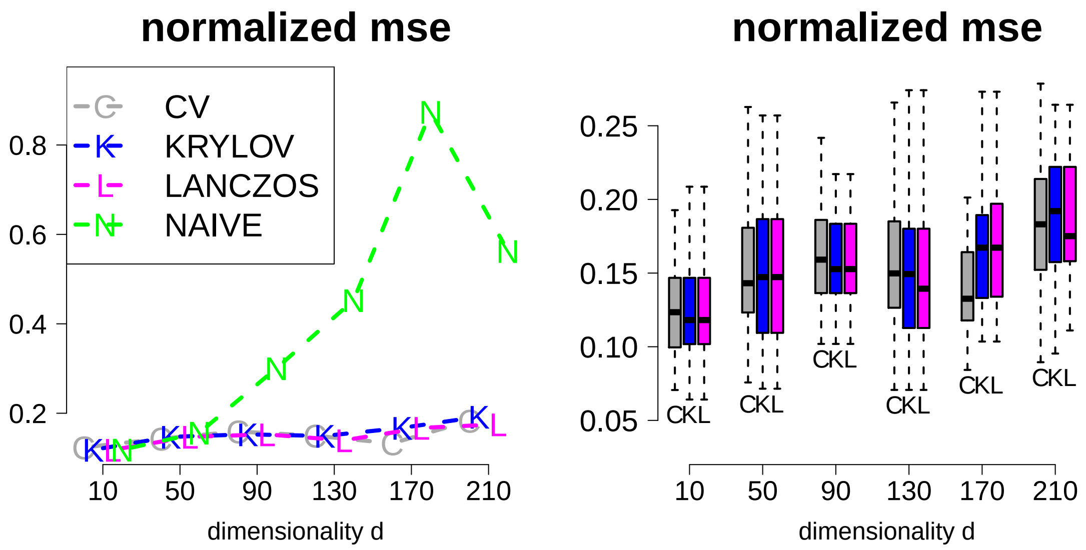

Figure 4: Simulation results. Left: Median of the normalized test error for the four different model selection criteria. Right: Boxplot of the normalized test error for all methods except for the naive approach.

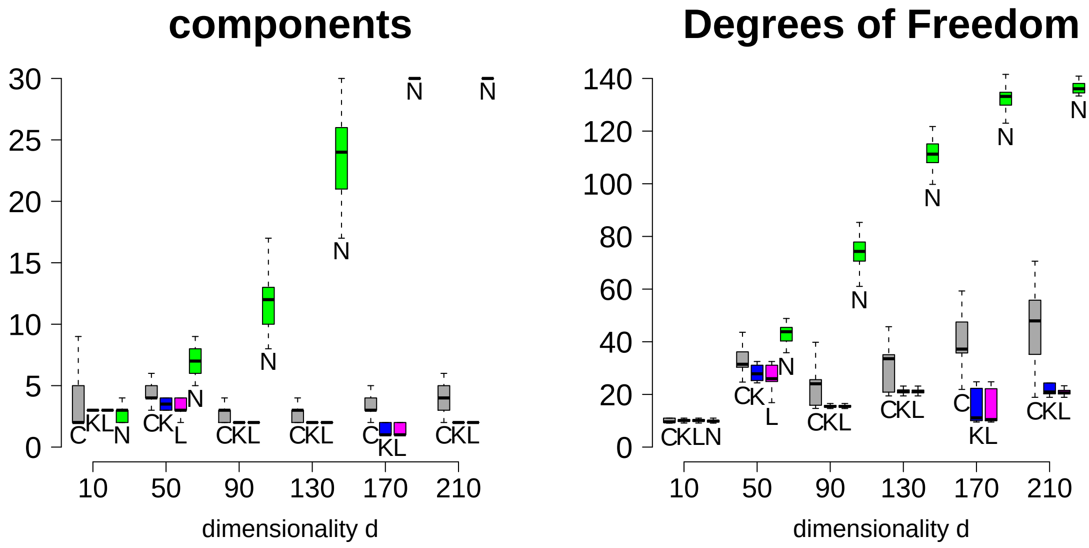

Figure 5: Simulation results. Left: Number of selected components. Right: Degrees of Freedom of the selected model.

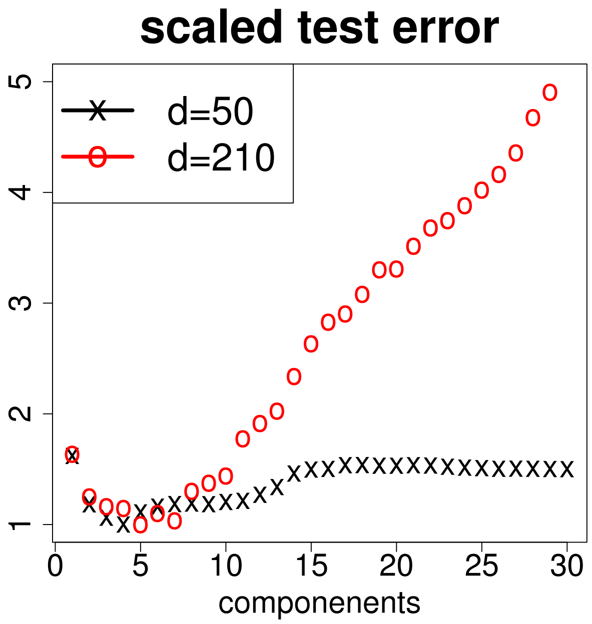

Figure 6: Illustration: Scaled mean test error as a function of the number of components. For each value of d (dimensionality), the test error is scaled such that its minimum value is 1.

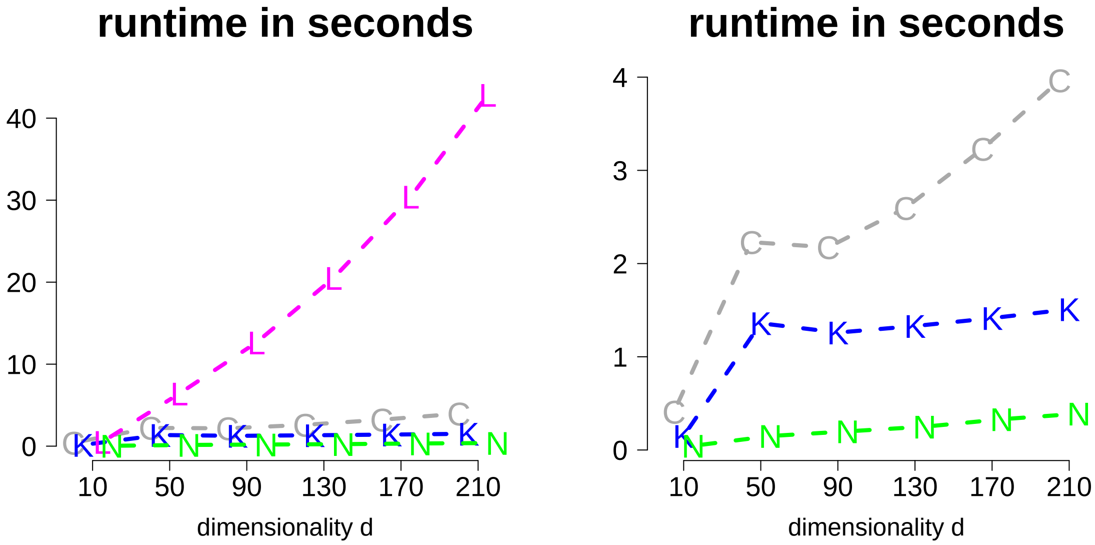

Figure 7: Simulation results. Left: Median runtime in seconds for the four different methods. Left: Median runtime in seconds for all methods except for LANCZOS.

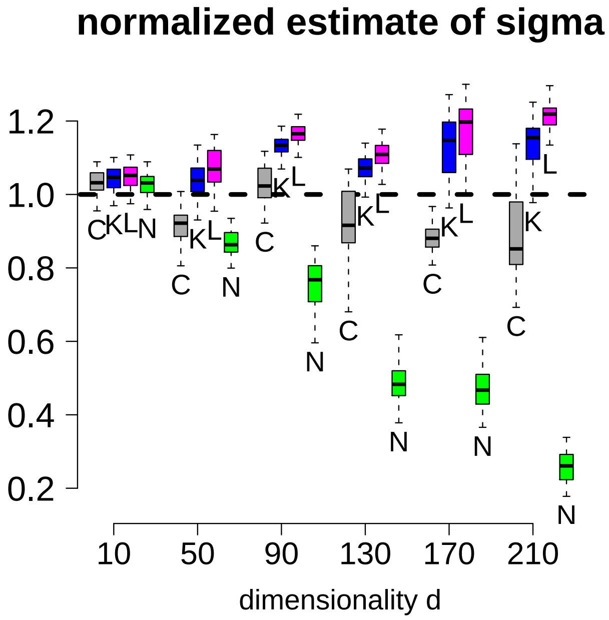

Figure 8: Simulation results. Normalized estimated noise levels $\widehat{\sigma}/\sigma$. The dashed horizontal line indicates the theoretical optimum.

that the latent components are orthogonal. However, this algorithm can stop very early. Therefore, we use equation (16) in the appendix that requires little additional computation time but ensures stability.

In some of the data, we observe that for a rather large number of components, both implementations for the Degrees of Freedom return negative Degrees of Freedom. This indicates a numerical problem. Therefore, in our experiments, we set the maximum number of components to m_* if we observe negative Degrees of Freedom for $m_* + 1$ components.

Finally, in Figure 7, we compare the runtime of the four different methods in our simulation study. While the absolute values are low, the four criteria show clear differences. The Lanczos decomposition is by far the slowest approach, as it first computes the derivative of the PLSR fit before computing the Degrees of Freedom. With respect to runtime, the algorithm based on the Krylov representation is therefore preferable, if the explicit derivative is not needed. It is also faster than 10-fold cross-validation. The naive approach is always the fastest method, as it only requires a single run of the PLSR algorithm and no additional computation of the Degrees of Freedom.

5 Discussion and Conclusion

Our findings show that typically, the Degrees of Freedom are higher for data sets with predictor variables that have low correlation and that each PLSR component consumes *more* than one Degree of Freedom. This confirms the longstanding conjecture that $\mathrm{DoF}(m) \geq m + 1$. This result may not come as a big surprise: For a fixed number of components, PLSR is less biased than Principal Components Regression (De Jong, 1993). This decrease in bias is balanced by an increased complexity in terms of Degrees of Freedom.

On average, the Degrees of Freedom of PLSR in combination with information criteria yield a similar prediction accuracy when compared to cross-validation. The naive approach that defines the Degrees of Freedom as the number of components plus one selects more complex models, which can in turn lead to worse prediction accuracy. This also effects the estimation of the noise level σ of the regression model. While our approach slightly overestimates σ, the naive approach yields estimates that are considerably biased downwards. From the computational point of view, the implementation based on the Krylov representation is preferable to the Lanczos-based algorithm, and – depending on the number k of splits – is faster than k-fold cross-validation.

In this paper, we applied the Degrees of Freedom estimate to the selection of the optimal number of PLSR components. It is possible to extend our framework to penalized PLSR (Goutis and Fearn, 1996; Reiss and Ogden, 2007; Krämer et al., 2008), where an additional smoothing parameter has to be selected. The derivation of the Degrees of Freedom can be adapted accordingly.

The two implementations for the Degrees of Freedom capitalize on the close connection between PLSR and methods from numerical linear algebra, namely the Lanczos decompositions and Krylov subspace approximations. Apart from the computational advances that are pointed out in this paper, this connection is very fruitful to analyze statistical properties of PLSR in a concise way. Recent results on the correspondence of penalized PLSR to preconditioning (Krämer et al., 2008) and on the prediction consistency of PLSR (Blanchard and Krämer, 2010a,b) underpin the potential of this connection. We strongly believe that the interplay between numerical linear algebra and PLSR will further stimulate the field of statistics.

Acknowledgement This work is funded in part by the FP7-ICT Programme of the European Community, under the PASCAL2 Network of Excellence, ICT-216886, and by the Funding Program for World-Leading Innovative R&D on Science and Technology (FIRST).

References

Akaike, H. (1973). Information Theory and an Extension of the Likelihood Ratio Principle. In *Second International Symposium of Information Theory*, pages 267–281.

Blanchard, G. and Krämer, N. (2010a). Kernel Partial Least Squares is Universally Consistent. *JMLR Workshop and Conference Proceedings (AISTATS 2010)*, 9:57–64.

Blanchard, G. and Krämer, N. (2010b). Optimal learning rates for kernel conjugate gradient regression. *Advances in Neural Information Processing Systems (NIPS)*, 23:226–234.

Brown, P., Fearn, T., and Vannucci, M. (2001). Bayesian Wavelet Regression on Curves with Application to a Spectroscopic Calibration Problem. *Journal of the American Statistical Association*, 96(454).

De Jong, S. (1993). PLS fits closer than PCR. *Journal of Chemometrics*, 7:551–551.

Denham, M. C. (1997). Prediction Intervals in Partial Least Squares. *Journal of Chemometrics*, 11(1):39–52.

Durand, J. and Sabatier, R. (1997). Additive Splines for Partial Least Squares Regression. *Journal of the American Statistical Association*, 92:1546–1554.

Efron, B. (2004). The Estimation of Prediction Error: Covariance Penalties and Cross-Validation. *Journal of the American Statistical Association*, 99(467):619–633.

Eldén, L. (2004). Partial least-squares vs. Lanczos bidiagonalizationI: analysis of a projection method for multiple regression. *Computational Statistics and Data Analysis*, 46(1):11–31.

Frank, I. and Friedman, J. (1993). A Statistical View of Some Chemometrics Regression Tools. *Technometrics*, 35(2):109–135.

Goutis, C. and Fearn, T. (1996). Partial Least Squares Regression on Smooth Factors. *Journal of the American Statistical Association*, 91:627–632.

Hansen, M. and Yu, B. (2001). Model Selection and Minimum Descripion Length Principle. *Journal of the American Statistical Association*, 96:746–774.

Hoskuldsson, A. (1988). PLS Regression Methods. *Journal of Chemometrics*, 2(3):211–228.

Krämer, N. and Boulesteix, A.-L. (2009). *ppls: Penalized Partial Least Squares.* R package version 1.4.

Krämer, N., Boulesteix, A.-L., and Tutz, G. (2008). Penalized Partial Least Squares with Applications to B-Spline Transformations and Functional Data. *Chemometrics and Intelligent Laboratory Systems*, 94:60–69.

Krämer, N. and Braun, M. (2007). Kernelizing PLS, Degrees of Freedom, and Efficient Model Selection. In Ghahramani, Z., editor, *Proceedings of the 24th International Conference on Machine Learning*, pages 441 – 448.

Krämer, N. and Braun, M. L. (2009). *plsdof: Degrees of Freedom and Confidence Intervals for Partial Least Squares Regression.* R package version 0.2-1.

Lanczos, C. (1950). An Iteration Method for the Solution of the Eigenvalue Problem of Linear Differential and Integral Operators. *Journal of Research of the National Bureau of Standards*, 45:225–280.

Leisch, F. and Dimitriadou, E. (2010). *mlbench: Machine Learning Benchmark Problems.* R package version 1.1-6.

Manne, R. (1987). Analysis of Two Partial-Least-Squares Algorithms for Multivariate Calibration. *Chemometrics and Intelligent Laboratory Systems*, 2:187–197.

Martens, H. and Naes, T. (1989). *Multivariate Calibration.* Wiley, New York.

Osborne, B., Miller, A., Fearn, T., and Douglas, S. (1984). Application of Near Infrared Reflectance Spectroscopy to the Compositional Analysis of Biscuits and Biscuit Doughs. *Journal of the Science of Food and Agriculture*, 35:99–105.

Phatak, A., Rilley, P., and Penlidis, A. (2002). The Asymptotic Variance of the Univariate PLS Estimator. *Linear Algebra and its Applications*, 354:245–253.

Reiss, P. and Ogden, R. (2007). Functional Principal Component Regression and Functional Partial Least Squares. *Journal of the American Statistical Association*, 102(479):984–996.

Rosipal, R. and Trejo, L. (2001). Kernel Partial Least Squares Regression in Reproducing Kernel Hilbert Spaces. *Journal of Machine Learning Research*, 2:97–123.

Schwarz, G. (1978). Estimating the Dimension of a Model. *The Annals of Statistics*, 6(2):461–464.

Serneels, S., Lemberge, P., and Espen, P. V. (2004). Calculation of PLS Prediction Intervals Using Efficient Recursive Relations for the Jacobian Matrix. *Journal of Chemometrics*, 18:76–80.

Van der Voet, H. (1999). Pseudo-Degrees of Freedom for Complex Predictive Models: the Example of Partial Least Squares. *Journal of Chemometrics*, 13:195–208.

Wold, H. (1975). Path models with Latent Variables: The NIPALS Approach. In et al., H. B., editor, *Quantitative Sociology: International Perspectives on Mathematical and Statistical Model Building*, pages 307–357. Academic Press.

A Derivation of Algorithm 1

The weight vector $\boldsymbol{w}_i$ can be rewritten as $\boldsymbol{w}_i = \boldsymbol{X}^\top (\boldsymbol{y} - \widehat{\boldsymbol{y}}_{i-1}) \propto \boldsymbol{s} - \boldsymbol{S}\widehat{\boldsymbol{\beta}}_{i-1}$. We define the "pseudo"-weight vector $\boldsymbol{v}_i$ via $\boldsymbol{t}_i = \boldsymbol{X}_i \boldsymbol{w}_i =: \boldsymbol{X}\boldsymbol{v}_i$. Using the fact that the matrix $\boldsymbol{L} = \boldsymbol{T}^\top \boldsymbol{X}\boldsymbol{W}$ defined in (10) is upper-triangular, we yield

$$\begin{aligned} \boldsymbol{v}_i &= \boldsymbol{w}_i - \boldsymbol{v}_{i-1}\boldsymbol{v}_{i-1}^\top \boldsymbol{S}\boldsymbol{w}_i \qquad (15) \\ &\quad - \sum_{j=1}^{i-2} \boldsymbol{v}_j \boldsymbol{v}_j^\top \boldsymbol{S}\boldsymbol{w}_i \,. \qquad (16) \end{aligned}$$

Note that $\boldsymbol{L}$ is in fact also upper-diagonal, hence the term in second line is 0. However, to ensure numerical stability, we include the term in our computation. The normalization of $\boldsymbol{t}_i$ to unit length corresponds to

$$\boldsymbol{v}_i \leftarrow \frac{\sqrt{n-1}}{\|\boldsymbol{v}_i\|_{\boldsymbol{S}}} \boldsymbol{v}_i = \frac{\sqrt{n-1}}{\sqrt{\boldsymbol{v}_i^\top \boldsymbol{S}\boldsymbol{v}_i}} \boldsymbol{v}_i \,.$$

It follows that $\widehat{\boldsymbol{\beta}}_i = \widehat{\boldsymbol{\beta}}_{i-1} + \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{s}$.

B Description of the Data Sets

`kin (fh)` Simulation of the forward dynamics of an eight link all-revolute robot arm. The 32 predictor variables correspond to positions of joints and to twist angles, length and offset distance for links. The task is to predict the distance of the end-effector from a target. The problem is fairly linear (f) and contains a high amount of noise (h). The data is available at the delve-repository `http://www.cs.toronto.edu/~delve/`.

`ozone` Los Angeles ozone pollution data 1976. The 12 predictor variables contain the date of the measurement and information on wind speed, humidity, temperature etc. The task is to predict the daily maximum one-hour-average ozone reading. The original data contains missing values. From the 366 examples, we use the 203 examples with no missing values. The data is provided by the R-package 'mlbench' (Leisch and Dimitriadou, 2010).

`cookie` Quantitative NIR spectroscopy for dough piece. A Near Infrared reflectance spectrum is available for each dough piece. The spectral data consist of 700 points measured from 1100 to 2498 nanometers (nm) in steps of 2 nm. The task is to predict the percentage of fat. The data is first analyzed by Osborne et al. (1984) and Brown et al. (2001). The data set provided by the R-package 'ppls' (Krämer and Boulesteix, 2009) also contains the percentage of sucrose, dry flour, and water.