Weierstraß-Institut
für Angewandte Analysis und Stochastik

im Forschungsverbund Berlin e.V.

Preprint ISSN 0946 – 8633

The degrees of freedom of partial least squares regression

Nicole Krämer[1] Masashi Sugiyama[2]

submitted: March 8, 2010

[1] Weierstrass Institute
for Applied Analysis and Stochastics
Mohrenstr. 39
10117 Berlin
Germany
E-Mail: nicole.kraemer@wias-berlin.de

[2] Tokyo Institute of Technology
Department of Computer Science
2-12-1-W8-74, O-okayama, Meguro-ku
Tokyo, 152-8552
Japan
E-Mail: sugi@cs.titech.ac.jp

No. 1487
Berlin 2010

2000 *Mathematics Subject Classification.* 62F10, 62J05, 62H25.

Key words and phrases. regression, model selection, Partial Least Squares, Degrees of Freedom.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Mohrenstraße 39
10117 Berlin
Germany

Fax: + 49 30 2044975
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

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 that the Degrees of Freedom are useful for model selection. Our experiments indicate that the model complexity based on the Degrees of Freedom estimate is lower than the model complexity of the naive approach. In terms of prediction accuracy, both methods obtain the same accuracy as cross-validation.

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 for fitting 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 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 definition of Degrees of Freedom of 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 various benchmark data that the complexity of PLSR depends on the collinearity of the predictor variables: The higher the collinearity is, the lower is the complexity. 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 (Phatak et al., 2002; Denham, 1997). (ii) The second one 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 `ppls` (Krämer and Braun, 2010).

We investigate the performance of the Degrees of Freedom of PLSR with respect to model selection by comparing the test errors based on 10fold cross-validation and based on the Bayesian Information Criterion (Schwarz, 1978). 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 complexity based on our Degrees of Freedom estimate is typically lower than the model complexity of the naive approach. In terms of prediction accuracy, the two approaches are on a par with the gold-standard of cross-validation.

2 Methodological Background

We consider a multivariate regression problem

$$Y \;=\; f(\boldsymbol{x}) + \varepsilon\,,\; \varepsilon \sim \mathcal{N}\left(0, \sigma^2\right)\,, \tag{1}$$

and the task is to estimate the unknown function $f : \mathbb{R}^p \to \mathbb{R}$ from a finite set of n examples $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n) \in \mathbb{R}^p \times \mathbb{R}$ drawn from (1). 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 $\overline{y}$ the mean of the response samples y_i. The $p \times p$ diagonal scaling matrix $\boldsymbol{D}$ is defined via $d_{ii} = 1/s(\boldsymbol{x})_i$. The $n \times p$ data matrix $\boldsymbol{X}$ is the matrix whose rows are the centered and scaled $\boldsymbol{x}_i$, 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,

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

the definitions given in Subsection 2.1 do not require f to be a linear function. Finally, we define the matrix and vector of empirical correlations via

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

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 a 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,

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

As we point out below, PLS regression is not a linear method, and the above definition cannot we applied. In order to extend the notion of Degrees of Freedom to PLS regression, we propose the generalized definition proposed in 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*

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

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 (aic) (Akaike, 1973)

$$\operatorname{aic}(\lambda) \;=\; \|\widehat{\boldsymbol{y}}_\lambda - \boldsymbol{y}\|^2 + 2\sigma^2 \operatorname{DoF}(\lambda)\,,$$

the Bayesian information criterion (bic) (Schwarz, 1978)

$$\operatorname{bic}(\lambda) \;=\; \|\widehat{\boldsymbol{y}}_\lambda - \boldsymbol{y}\|^2 + \log(n)\sigma^2 \operatorname{DoF}(\lambda)\,, \tag{5}$$

and the generalized minimum description length (gmdl) (Hansen and Yu, 2001)

$$\operatorname{gmdl}(\lambda) \;=\; n\log(S_\lambda) + \operatorname{DoF}(\lambda)\log(F_\lambda) + \log(n)$$

with

$$S_\lambda = \|\widehat{\boldsymbol{y}}_\lambda - \boldsymbol{y}\|^2/n - \operatorname{DoF}\lambda \quad , \quad F_\lambda = \|\widehat{\boldsymbol{y}}_\lambda\|^2/\operatorname{DoF}(\lambda)\boldsymbol{S}_\lambda\,.$$

For both aic and bic, 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. We have

$$\begin{aligned} E\left[\|\widehat{\boldsymbol{y}}_\lambda - \boldsymbol{y}\|^2\right] &= \|E[\widehat{\boldsymbol{y}}_\lambda - \boldsymbol{y}]\|^2 + \operatorname{var}(\widehat{\boldsymbol{y}}_\lambda - \boldsymbol{y}) \\ &= \|E[\widehat{\boldsymbol{y}}_\lambda - \boldsymbol{y}]\|^2 + \operatorname{trace}\left((\boldsymbol{I}_n - \boldsymbol{H}_\lambda)(\boldsymbol{I}_n - \boldsymbol{H}_\lambda^\top)\right)\sigma^2\,. \end{aligned}$$

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

$$\widehat{\sigma}_*^2 \;=\; \frac{\|\widehat{\boldsymbol{y}}_\lambda - \boldsymbol{y}\|^2}{\operatorname{trace}\left((\boldsymbol{I}_n - \boldsymbol{H}_\lambda)(\boldsymbol{I}_n - \boldsymbol{H}_\lambda^\top)\right)}\,. \tag{6}$$

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

$$\widehat{\sigma}^2 \;=\; \frac{\|\widehat{\boldsymbol{y}}_\lambda - \boldsymbol{y}\|^2}{n - \operatorname{DoF}(\lambda)}\,. \tag{7}$$

The latter estimate is most commonly used, even if the above assumption is not fulfilled. Equation (7) also shows that the estimation of the noise level is implicit in the gmdl criterion (via the quantity S_λ).

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 $\boldsymbol{t}_1 = \boldsymbol{X}\boldsymbol{w}_i$ maximizes the squared covariance to the response $\boldsymbol{y}$,

$$\boldsymbol{w}_1 = \arg\max_{\boldsymbol{w}} \quad \frac{\|\mathrm{cov}(\boldsymbol{X}\boldsymbol{w}, \boldsymbol{y})\|^2}{\boldsymbol{w}^\top \boldsymbol{w}} = \arg\max_{\boldsymbol{w}} \frac{\boldsymbol{w}^\top \boldsymbol{X}^\top \boldsymbol{y}\boldsymbol{y}^\top \boldsymbol{X}\boldsymbol{w}}{\boldsymbol{w}^\top \boldsymbol{w}} \propto \boldsymbol{X}^\top \boldsymbol{y}\,. \tag{8}$$

The vector $\boldsymbol{w}_i$ is called a weight vector. 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 $\boldsymbol{X}$. That is, we only consider the part of $\boldsymbol{X}$ that is orthogonal on $\boldsymbol{t}_j, j < i$:

$$\boldsymbol{X}_i \;=\; \boldsymbol{X} - \mathcal{P}_{\boldsymbol{t}_1,\ldots,\boldsymbol{t}_{i-1}} \boldsymbol{X}\,. \tag{9}$$

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 (8). 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

$$\widehat{\boldsymbol{y}}_m \;=\; \overline{y} + \boldsymbol{T}\left(\boldsymbol{T}^\top \boldsymbol{T}\right)^{-1} \boldsymbol{T}^\top \boldsymbol{y} = \overline{y} + \mathcal{P}_{\boldsymbol{T}} \boldsymbol{y}\,. \tag{10}$$

We emphasize 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

$$\boldsymbol{L} \;=\; \boldsymbol{T}^\top \boldsymbol{X}\boldsymbol{W} \in \mathbb{R}^{m \times m} \tag{11}$$

and obtain (Manne, 1987; Hoskuldsson, 1988)

$$\widehat{\boldsymbol{\beta}}_m = \boldsymbol{D}\boldsymbol{W}\boldsymbol{L}^{-1}\boldsymbol{T}^\top \boldsymbol{y} \quad \text{and} \quad \widehat{\beta}_m^{(0)} = \overline{y} + \overline{\boldsymbol{x}}^\top \widehat{\boldsymbol{\beta}}_m\,.$$

Here, $\boldsymbol{D}$ is the diagonal scaling matrix.

3 Unbiased Degrees of Freedom

As explained above, 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*

$$\widehat{\mathrm{DoF}}(m) \;=\; 1 + \mathrm{trace}\left(\frac{\partial \mathcal{P}_{\boldsymbol{T}} \boldsymbol{y}}{\partial \boldsymbol{y}}\right). \tag{12}$$

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 (12) explicitly. We propose two equivalent algorithms in Subsections 3.2 and 3.3.

Table 1: Properties of the six benchmark data sets.

data set	variables	examples	training examples	mean absolute correlation
`kin (fh)`	32	8192	60	low (0.009)
`bank (fh)`	8	8192	30	low (0.009)
`ozone`	12	203	50	medium (0.260)
`boston`	13	506	50	medium (0.390)
`tecator`	100	215	129	high (0.986)
`cookie`	700	70	39	high (0.867)

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 various benchmark data. An overview on the six data sets is given in Table 1(see the appendix for more details). We choose the six 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 fifth column of Table 1. In Figure 1, we plot the Degrees of Freedom for each of the six data sets. We use all samples for the computation except for the large data sets `kin (fh)` and `bank (fh)`, where we use a subsample of size 300. In addition, we display the naive estimate $\mathrm{DoF}(m) = m+1$, i.e. the Degrees of Freedom equals the number of components plus 1. In all examples, our Degrees of Freedom estimate 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); 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 by a lower bound on the Degrees of Freedom of PLSR with $m = 1$ components.

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

$$\lambda_{\max} \;\leq\; \frac{1}{2} \mathit{trace}(\boldsymbol{S})\,, \tag{13}$$

then

$$\widehat{\mathrm{DoF}}(m=1) \;\geq\; 1 + \frac{\mathit{trace}(\boldsymbol{S})}{\lambda_{\max}}\,. \tag{14}$$

Condition (13) 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 (13) is fulfilled). The lower

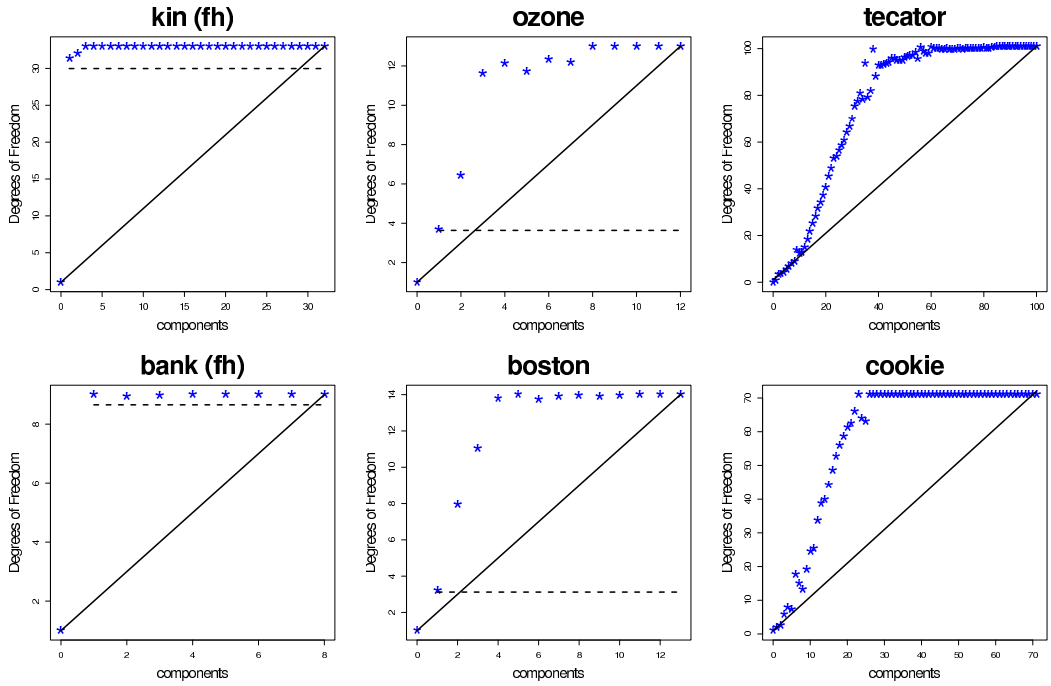

Figure 1: Estimated Degrees of Freedom (blue stars) for the six benchmark data sets. The solid line displays the naive estimate $\mathrm{DoF}(m) = m + 1$. If the assumption of theorem 3 is fulfilled, we also display the lower bound on the Degrees of Freedom for 1 component (dashed horizontal line).

bound (14) is higher for data with low collinearity. In Figure 1, we add the lower bound for the data sets which fulfill the condition. This is the case for all but the two data sets `tecator` and `cookie`.

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

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

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

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

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

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

Condition (13) ensures that the right-hand side of this inequality is ≥ 0, hence,

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

□

3.2 First Derivative of the Lanczos Decomposition

We extend the approaches that are proposed in Denham (1997); 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 (11). Note that the matrix $\boldsymbol{L}$ is upper bidiagonal, i.e. $l_{ij} = 0$ for $i > j$ or $i < j - 1$. The relationship (11) 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 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 (Phatak et al., 2002; Denham, 1997).

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 (9), and only depends on projections onto one-dimensional subspaces. Second, we determine the first derivative of one-dimensional projection operators, which are based on (Krämer and Braun, 2007)

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

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

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: $\widehat{\boldsymbol{\beta}}_0 = \boldsymbol{0}_p$, $(\partial \boldsymbol{\beta}_0/\partial \boldsymbol{y}) = \boldsymbol{0}_{p\times n}$
for $i = 1, \ldots, m$ **do**
  $\boldsymbol{w}_i = \boldsymbol{s} - \boldsymbol{S}\widehat{\boldsymbol{\beta}}_{i-1}$
    $(\partial \boldsymbol{w}_i/\partial \boldsymbol{y}) = \boldsymbol{X}^\top/(n-1) - \boldsymbol{S}\left(\partial \boldsymbol{\beta}_{i-1}/\partial \boldsymbol{y}\right)$
  $\boldsymbol{v}_i = \boldsymbol{w}_i - \sum_{j=1}^{i-1} \boldsymbol{v}_j \boldsymbol{v}_j^\top \boldsymbol{S}\boldsymbol{w}_i$
    $(\partial \boldsymbol{v}_i/\partial \boldsymbol{y}) = (\partial \boldsymbol{w}_i/\partial \boldsymbol{y}) - \sum_{j=1}^{i-1}\left(\partial \boldsymbol{v}_j \boldsymbol{v}_j^\top \boldsymbol{S}\boldsymbol{w}_i/\partial \boldsymbol{y}\right)$
  $\boldsymbol{v}_i = \sqrt{n-1}\boldsymbol{v}_i/\|\boldsymbol{v}_i\|_{\boldsymbol{S}}$
    $(\partial \boldsymbol{v}_i/\partial \boldsymbol{y}) = \sqrt{n-1}\partial\left(\boldsymbol{v}_i/\|\boldsymbol{v}_i\|_{\boldsymbol{S}}\right)/\partial \boldsymbol{y}$
  $\widehat{\boldsymbol{\beta}}_i = \widehat{\boldsymbol{\beta}}_{i-1} + \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{s}$
    $\partial \boldsymbol{\beta}_i/\partial \boldsymbol{y} = \partial \boldsymbol{\beta}_{i-1}/\partial \boldsymbol{y} + \left(\partial \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{s}/\partial \boldsymbol{y}\right)$
end for
DoF$(m) = 1 + \operatorname{trace}\left(\boldsymbol{X}\partial \boldsymbol{\beta}_m/\partial \boldsymbol{y}\right)$.

$\widehat{\boldsymbol{\beta}}_m \approx \left(\partial \widehat{\boldsymbol{\beta}}_m/\partial \boldsymbol{y}\right)\boldsymbol{y}$, which leads to

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

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

$$\boldsymbol{H}_m \;=\; \frac{\partial \boldsymbol{y}_m}{\partial \boldsymbol{y}}. \tag{15}$$

This matrix can be plugged into formula (6) for the estimation of the noise level.

3.3 Trace of the Krylov Representation

The computation of the derivative of $\widehat{\boldsymbol{y}}_m$ in Subsection 3.2 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 $\operatorname{span}\{\boldsymbol{t}_1, \ldots, \boldsymbol{t}_m\} = \operatorname{span}\{\boldsymbol{K}\boldsymbol{y}, \ldots, \boldsymbol{K}^m\boldsymbol{y}\} =: \mathcal{K}_m$, 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 $\widehat{\boldsymbol{y}}_m = \overline{y} + \mathcal{P}_{\mathcal{K}_m}$ 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*

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

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 3.2), 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

We evaluate the performance of the Degrees of Freedom with respect to model selection. We investigate the performance both with respect to the prediction accuracy (Subsection 4.1) and with respect to the complexity of the selected model (Subsection 4.2). 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 use the six benchmark data sets that are introduced in Table 1. For all data sets, we split the data into training and test data using n_{train} training examples (see Table 1 for the respective values of n_{train}). On the training data, we apply four different model selection criteria. (a) **CV**: 10-fold cross-validation. (b) **BIC(l)**: 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 (6) with the approximate hat-matrix defined in (15). (c) **BIC(k)**: 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 (7). (d) **BIC(n)**: Bayesian Information Criterion with the naive Degrees of Freedom $\mathrm{DoF}(m) = m + 1$. For the estimation of the noise level, we use equation (7). Note that BIC(l) and BIC(k) 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. For each of the four criteria, we measure the performance on a separate hold-out set. The procedure is repeated 50 times.

4.1 Prediction Accuracy

We display the boxplot of the test errors in Figure 2. If the notches of two boxes do not overlap this is evidence that the two medians differ. The four criteria do not show any strong difference in performance over the six data sets. A remarkable point is that on average, the naive approach BIC(n) does not lead to a higher test error than the criteria that use our estimates of the Degrees of Freedom of PLSR. The experiments indicate that in terms of prediction accuracy, the naive approach yields competitive results. This observation is important in scenarios with time constraints. If computationally intensive model selection is not feasible due to lack of time, the naive approach is a fast alternative. We refer to Subsection 4.3 for more details on computation time.

4.2 Model Complexity

Figure 3 displays the number of selected components for the four different criteria. In addition, we show the Degrees of Freedom for the selected number of components in Table 4. While their prediction accuracy is very similar, the model complexity of our Degrees of Freedom estimate and the naive approach differ significantly. Except for the `bank (fh)` data, the number of selected components is always higher for the naive approach. This underpins the fact that it underestimates the intrinsic complexity of PLSR. If the main goal of the data analysis is the construction of a predictive model with low dimensionality, the criteria based on cross-validation or our Degrees of Freedom estimates are therefore preferable.

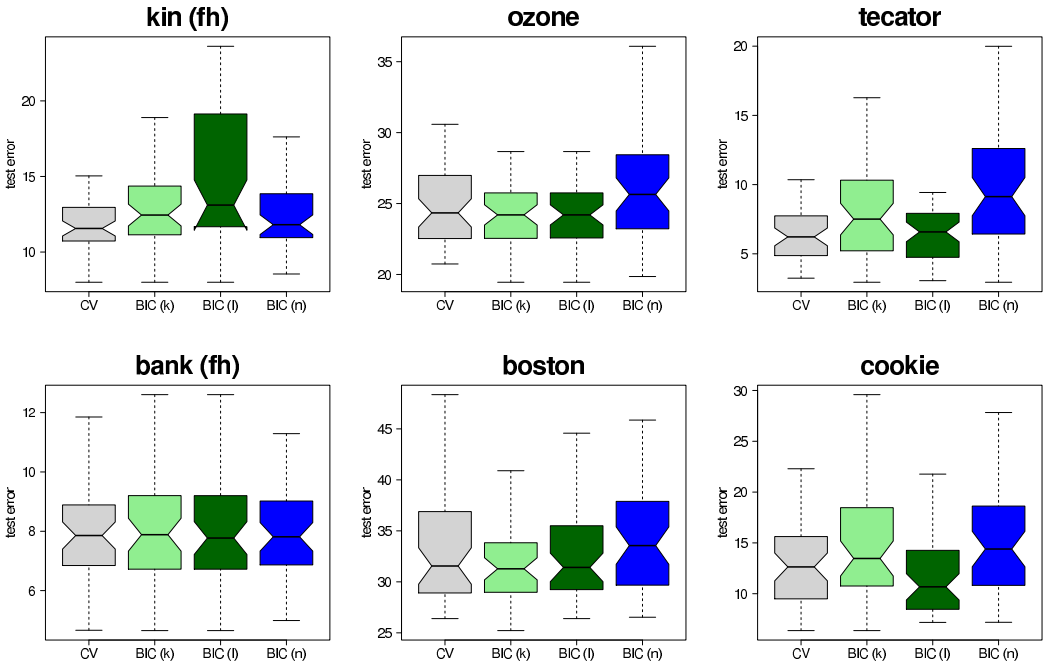

Figure 2: Test error for the model selection criteria. To improve the readability, we multiply the results for `kin (fh)` and `cookie` with 100, and the results for `bank (fh)` with 1 000.

4.3 Numerical Stability and Run-Time

As explained in Subsection 3.2, the sparse structure of $\boldsymbol{L}$ 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) we use the sparse structure of $\boldsymbol{L}$ and impose an additional stopping criterion to ensure that the latent components are orthogonal. However, this algorithm can stop very early (sometimes after 5 iterations on the `tecator` data set). Therefore, we use formulation (16) that requires little additional computation time but ensures stability.

In some of the data, we observe that for a rather large amount of components, both implementations for the Degrees of Freedom return negative Degrees of Freedom. E.g. on the `tecator` data set, this occurs for the Lanczos representation on average if we compute more than 26 components. 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.

Table 4.3 shows the runtime of the four model selection criteria on the six different data sets. For `cookie`, we use an implementation that relies on the kernel matrix

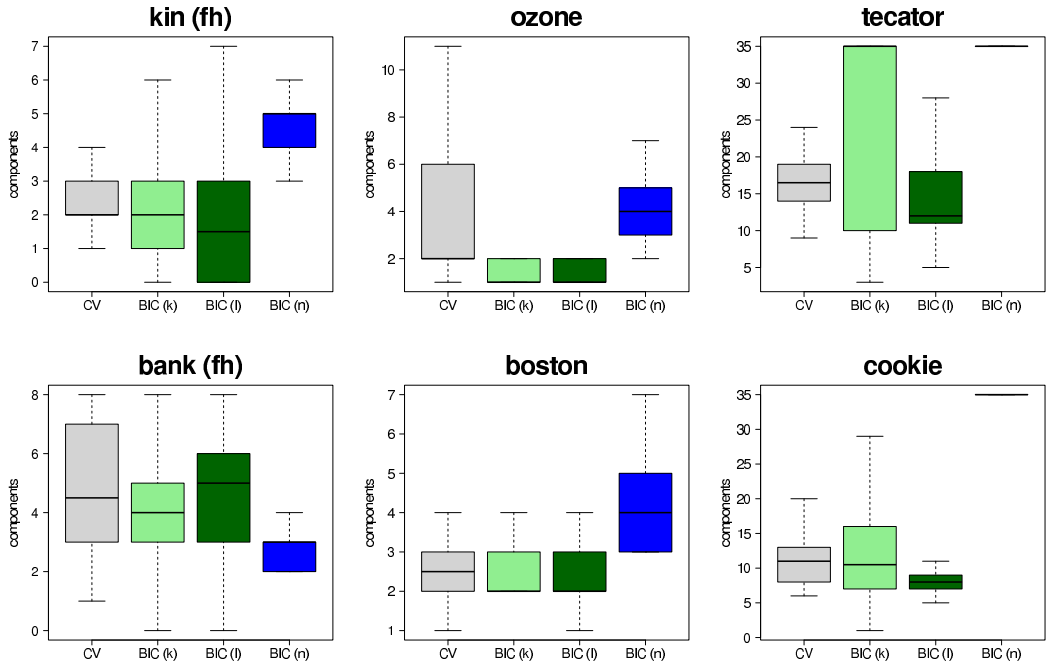

Figure 3: Number of selected components for the model selection criteria.

$\boldsymbol{K}$ and $\boldsymbol{y}$ (Krämer and Braun, 2007). While the absolute values are low, the four criteria shows 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. 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.

Table 2: Average runtime (in seconds) of the four criteria.

data	CV	BIC(l)	BIC(k)	BIC(n)
`kin (fh)`	0.236	0.549	0.238	0.024
`bank (fh)`	0.048	0.022	0.014	0.005
`ozone`	0.111	0.091	0.031	0.006
`boston`	0.079	0.062	0.032	0.008
`tecator`	0.945	6.259	0.688	0.093
`cookie`	1.325	0.672	0.435	0.132

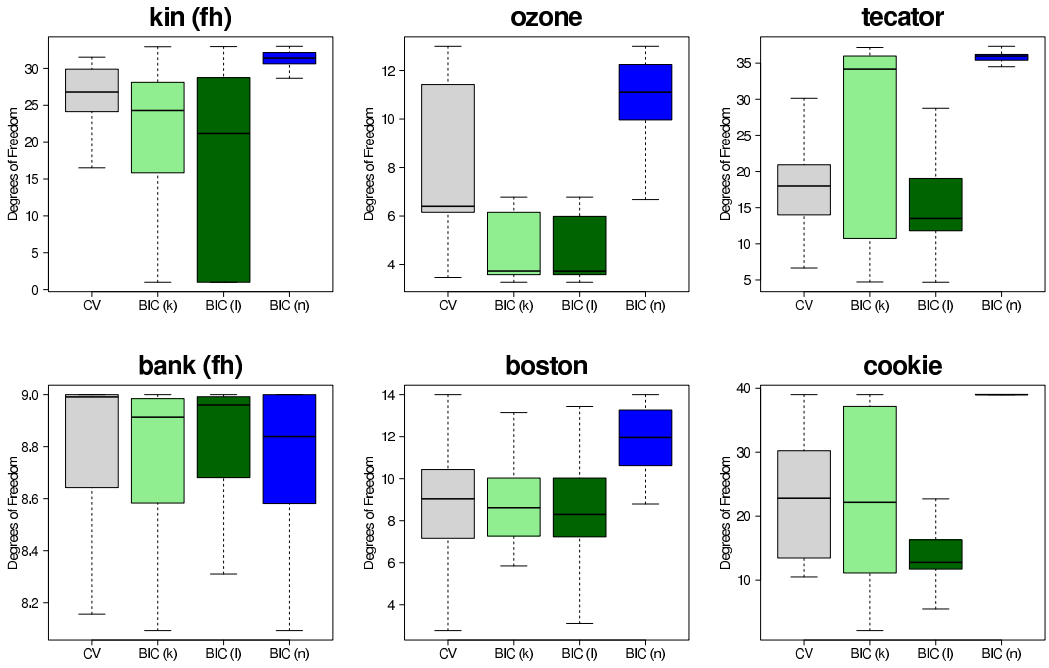

Figure 4: Degrees of Freedom for the mumber of selected components for the model selection criteria.

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 the same prediction accuracy as cross-validation. We find it remarkable that the accuracy of the naive Degrees of Freedom is on a par with both cross-validation and our Degrees of Freedom estimate. However, in terms of the number of PLSR components, the naive approach typically selects more complex models, confirming again that the naive Degrees of Freedom underestimate the intrinsic complexity of PLSR. For many moderate sized data sets, the prediction accuracy 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, and this is a possible explanation for the good performance of the naive approach.

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, 2010) 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.

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. (2010). Kernel Partial Least Squares is Universally Consistent. *Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*. accepted, `http://arxiv.org/abs/0902.4380`.

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 bidiagonalizationŮI: 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.

Harrison, D. and Rubinfeld, D. (1978). Hedonic Housing Prices and the Demand for Clean Air. *Journal of Environmental Economics and Management*, 5(1):81–102.

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. (2010). *plsdof: Degrees of Freedom and Confidence Intervals for Partial Least Squares Regression*. R package version 0.2-0.

Kuhn, M., Wing, J., Weston, S., Williams, A., Keefer, C., and Engelhardt, A. (2009). *caret: Classification and Regression Training*. R package version 4.31.

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.

Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. Springer, New York, fourth edition.

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 $\boldsymbol{t}_i = X_i \boldsymbol{w}_i =: \boldsymbol{X}\boldsymbol{v}_i$. Using the sparsity of the matrix $\boldsymbol{L}$, we obtain

$$\boldsymbol{v}_i = \boldsymbol{w}_i - \boldsymbol{v}_{i-1}\boldsymbol{v}_{i-1}^\top \boldsymbol{S}\boldsymbol{w}_i \left[- \sum_{j=1}^{i-2} \boldsymbol{v}_j \boldsymbol{v}_j^\top \boldsymbol{S}\boldsymbol{w}_i \right] . \tag{16}$$

Due to the upper-bidiagonal structure of $\boldsymbol{L}$, the term in the square brackets is 0 and hence superfluous. However, to ensure numerical stability, we include the term in our computation. The normalization of $\boldsymbol{t}_i$ to unit length corresponds to

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

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 hight amount of noise (h). The data is available at the delve-repository `http://www.cs.toronto.edu/~delve/`.

`bank (fh)` Simulation of how bank-customers choose their banks. The eight variables are the coordinates of the residual area of the costumer and of the banks, the population sizes of the areas of interest, a temperature controlling bank choice and the maximum possible length of queues. The task is to predict the fraction of bank customers who leave the bank because of full queues. The problem is fairly linear (f) and contains a hight amount of noise (h). The data is available at the delve-repository `http://www.cs.toronto.edu/~delve/`.

`boston` Census data. This data set contains information collected by the U.S Census Service concerning housing in the area of Boston. The 13 predictor variables (per capita crime rate by town, average number of rooms per dwelling, etc.) describe different aspects of the neighborhood. The task is to predict the medium value of a home. The data is first analyzed in Harrison and Rubinfeld (1978) and is provided by the R-package 'MASS' (Venables and Ripley, 2002).

`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).

`tecator` Near Infrared Transmission for samples of meat. For each meat sample the predictor variables are a 100 channel spectrum of absorbances. The task is to predict the amount of water in the meat. The data set provided by the R-package 'caret' (Kuhn et al., 2009) also contains the amount of fat and protein.

`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 in Brown et al. (2001); Osborne et al. (1984). The data set provided by the R-package 'ppls' (Krämer and

Boulesteix, 2009) also contains the percentage of sucrose, dry flour, and water.

For `tecator` and `cookie`, the number of training samples matches the number of training samples in the original experiments on the data. For the other data sets, we choose the number of training samples such that the regression problem is fairly high-dimensional.