NMBU

Abstract

In this thesis we have tried to find if or when multiresponse Partial Least Squares Regression(PLS2)predicts better than uniresponse PLS(PLS1).

With a simulation study and analysis of variance we have investigated how PLS1 predicts with different simulation parameter settings. The result showed that if we had small relevant eigenvalues, the predictor based on PLS1 does not predict well. We have also compared the estimated values with the true values of parameters, with focus on eigenvalues and covariances. Then we found that if we had small relevant eigenvalues, the estimated values was often very different from the true parameters.

The estimated regression coefficients found by PLS1 and PLS2 differ. We found empirical that for one component the PLS2 estimator is a linear combination of the two PLS1 estimators, one for each response.

For prediction the two PLS1 predictors and PLS2 predictor provide very similar result. The results showed that with some simulation parameter settings PLS2 was a better predictor than PLS1. This happened if we had only one common relevant component with a small relevant eigenvalue. Based on analysis of variance we found that the difference in prediction error between the two methods was larger, when the number of observations were few and there was high degree of collinearity simultaneous. However the variation between replications was found to be large. We have also tested the methods on real data sets, but PLS2 did not predict better than PLS1 on these. Therefore we concluded with that as far as we have seen PLS1 is a better choice as a predictor than PLS2.

Sammendrag

I denne oppgaven har vi forsøkt å finne ut om eller når multirespons Partial Least Squares Regression (PLS2) predikerer bedre enn unirespons PLS (PLS1).

Med en simuleringsstudie og variansanalyse har vi undersøkt hvordan PLS1 predikerer med forskjellige simuleringparameterinnstillinger. Resultatet viste at hvis vi hadde små relevante egenverdier, så vil prediktoren basert på PLS1 predikere dårlig. Vi har også sammenlignet estimerte verdier med de sanne verdiene fra parameterne, med fokus på egenverdier og kovarianser. Da fant vi at hvis vi hadde små relevante egenverdier, så var de estimerte verdiene ofte svært forskjellige fra de sanne parameterne.

De estimerte regresjon koeffisientene funnet av PLS1 og PLS2 er forskjellige. Vi fant empirisk at for en komponent så vil PLS2 estimatoren være en lineær kombinasjon av de to PLS1 estimatorene, en for hver respons.

For prediksjon ga PLS1 prediktorene og PLS2 prediktoren svært lignende resultat. Resultatene viste at med noen simuleringsparameterinnstillinger, så var PLS2 en bedre prediktor enn PLS1. Det skjedde når vi hadde kun en felles relevant komponent med en liten relevant egenverdi. Basert på en variansanalyse fant vi at forskjellen i prediksjonsfeil mellom de to metodene var større når antall observasjoner var få og det var høy grad av kollinearitet samtidig. Men variasjonen mellom replikasjoner ble funnet til å være stor. Vi har også testet metodene på virkelige datasett, men PLS2 predikerte ikke bedre enn PLS1 på disse. Derfor har vi konkludert med at så langt som vi har sett, så er PLS1 et bedre valg som prediktor enn PLS2.

Acknowledgement

This thesis is written at the Institute of Chemistry, Biotechnology and Food Science at the Norwegian University of Life Science.

This thesis would not have been possible without my supervisor Trygve Almøy. Thank you for all support, discussions and academic help. To Solve Sæbø for joining our discussions. And to both for always keeping their doors open and taking time to answer my questions.

I would also give a big thank you to friends and family for supporting me. And a special thanks to Morten for pushing me to work hard, motivating me and supporting me.

Ås, May 13, 2015

May Tove Alseth

Contents

Chapter 1

Introduction

1.1 Introduction

If there are more than one response that is to be predicted, we can either use an uniresponse or a multiresponse model. Using uniresponse we are constructing separate models for each response, while using multiresponse we are constructing one model for all responses. In [Höskuldsson and Esbensen, 2003] the authors argue that 'If we cannot distinguish the residuals derived by the model which is common for all $\boldsymbol{Y}$ variables from the ones obtained by using separate models, we *may* use either approach'. If the separate uniresponse model provide significant smaller residuals, the uniresponse models should be used. In some situation it is desirable to use only one model for all responses, but if the multiresponse model gives the same prediction as or worse prediction than the separate uniresponse models, there is no point in using multiresponse in prediction.

Many statistical methods do not yield different prediction, or estimated regression coefficients, when modeling as seperate uniresponse models compared to one multireponse model. The Least Squares regression does not use any possible correlation or other information among the responses. Therefore it will provide equal predictors. Whereas Partial Least Squares(PLS) is one method that does not yield similar results for uniresponse and multiresponse modeling. Uniresponse PLS(PLS1) will fit separate models for each response while multiresponse PLS(PLS2) will fit one model for all responses. It is only useful to use PLS2 in prediction if it provides better predictions than PLS1. Hence our main goal is to find if and when PLS2 predicts better than PLS1.

In [Frank and Friedman, 1993] the authors suggest that, unless response variables are uncorrelated, there might be something to gain by considering them together, compared to performing separate regressions.
In [Martens and Næs, 1989] similar argumentation is used, the authors claim that PLS2 is useful when the responses are strongly intercorrelated by stabilizing the determination of the loading weights against random noise in the individual responses. The correlation, both conditional and unconditional between responses, may affect the PLS1 and PLS2 model differently and have to be considered.

It is not only the response correlations that might result in PLS2 model predicting better than PLS1. Other aspects as the relevant components, the size of relevant eigenvalues, collinearity, the number of observations and et cetera are possible factors that influence the prediction ability of the models. In this thesis we will first look at a simulation study at how PLS1 predicts in several situation before looking at how PLS1 and PLS2 estimate regression coefficient differently and in the end we will try to find if there exist situations where PLS2 predicts better than PLS1.

Chapter 2

Stastical model and methods

2.1 Notation

In this thesis we use the following notation with a few exceptions:

All random variables are written with latin letters.

- All one-dimensional random variables are written with capital letter. Example Y.

- A vector is written with bold, lowercase latin letter. Example

$$\mathbf{y} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}$$

- A matrix of random variables is written with capital, bold letters, example $\mathbf{X}$. In some situations the dimensions of the matrix is given as $\underset{n\times p}{\mathbf{X}}$. The matrix has n rows and p columns.

- The transpose of a vector $\mathbf{y}$ is written $\mathbf{y}^t$

All parameters are written with greek letters

- A one-dimensional parameter is written with lowercase letter, example β
- A vector of parameters is written with lowercase, bold letter, example $\boldsymbol{\beta}$
- A matrix with parameters is written with a capital, bold letter, example $\mathbf{\Sigma}$
- An estimate of a parameter is written with a hat. An estimate of β is written as $\hat{\beta}$

2.2 Variables and Models

The number n is the number of observations in the dataset. For each observation, a response variable Y and p explanatory variables $\mathbf{x}$ is measured for the uniresponse case. The n observations are collected in a response vector $\underset{n\times 1}{\mathbf{y}}$ and an explanatory matrix $\underset{n\times p}{\mathbf{X}}$. For the multiresponse case with two responses the response is a matrix $\underset{n\times 2}{\mathbf{Y}}$.

All variables are centred.

$$\mathbf{y}_j^* = \mathbf{y}_j - \bar{y}_j\mathbf{1}$$

and

$$\mathbf{x}_i^* = \mathbf{x}_i - \bar{x}_i\mathbf{1}$$

Where $\bar{y}_j$ is the average of the j-th response vector, and j=1,2 for multiresponse, and j=1 for uniresponse. the vector $\mathbf{1}$ consist of ones. $\bar{x}_i$ is the average of the i-th explanatory vector, and i=1,...,p. We let $\mathbf{x}_i = \mathbf{x}_i^*$ and $\mathbf{y}_j = \mathbf{y}_j^*$(with a few exceptions).

The models are based on random calibration. The variables are drawn at random.

2.2.1 Uniresponse Model

For the uniresponse case we assume that $\underset{1\times 1}{Y}$ and $\underset{p\times 1}{\mathbf{x}}$ are multivariate normally distributed as:

$$\begin{bmatrix} Y \\ \mathbf{x} \end{bmatrix} \sim N_{p+1}\left(\begin{bmatrix} \mu_y \\ \boldsymbol{\mu_x} \end{bmatrix}, \begin{bmatrix} \sigma_y^2 & \boldsymbol{\sigma}_{xy}^t \\ \boldsymbol{\sigma}_{xy} & \boldsymbol{\Sigma}_{xx} \end{bmatrix}\right) \tag{2.1}$$

where μ_y is the expected value of Y and $\boldsymbol{\mu_x}$ is a vector with the expected values for $\mathbf{x}$, σ_y^2 is the variance of Y, $\underset{p\times 1}{\boldsymbol{\sigma}_{xy}}$ is the covariance between $\mathbf{x}$ and Y and $\underset{p\times p}{\boldsymbol{\Sigma}_{xx}}$ is the variance matrix for $\mathbf{x}$. Due to centring, $\mu_y = 0$ and $\boldsymbol{\mu_x} = \mathbf{0}$. The variance matrix $\boldsymbol{\Sigma}_{xx}$ can be written as

$$\boldsymbol{\Sigma}_{xx} = \sum_{i=1}^{p} \lambda_i \boldsymbol{e}_i \boldsymbol{e}_i^t \tag{2.2}$$

where λ_i is the i-th largest eigenvalue of $\boldsymbol{\Sigma}_{xx}$ and $\boldsymbol{e}_i$ is it's corresponding eigenvector. All p eigenvectors are orthogonal and has length 1. The matrix $\boldsymbol{X}^t\boldsymbol{X}$(which can be used as an estimate of $\boldsymbol{\Sigma}_{xx}$ usually by dividing by $n-1$) can be decomposed in a similar way.

$$\boldsymbol{X}^t\boldsymbol{X} = \sum_{i=1}^{p} \hat{\lambda}_i \hat{\boldsymbol{e}}_i \hat{\boldsymbol{e}}_i^t$$

Where $\hat{\lambda}_i$ is the i-th largest eigenvalue of $\boldsymbol{X}^t\boldsymbol{X}$.

The conditional distribution of $Y|\boldsymbol{x}$ can be written as

$$Y_i|\boldsymbol{x}_i = \boldsymbol{\beta}^t\boldsymbol{x}_i + \epsilon_i, \qquad i = 1, 2, ..., n$$

which is exactly the linear model. Here $\boldsymbol{\beta}$ is an unknown $p \times 1$ parameter vector that must be estimated and ϵ is the error-term. Alternative we can write the model as

$$\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\epsilon} \tag{2.3}$$

Where $\boldsymbol{y}$ is the $n \times 1$ response vector and $\boldsymbol{X}$ is the $n \times p$ explanatory matrix. $\boldsymbol{\epsilon}$ is multivariate normally distributed

$$\boldsymbol{\epsilon} \sim N_n(\mathbf{0}, \boldsymbol{\Sigma}_\epsilon),$$

where $\boldsymbol{\Sigma}_\epsilon$ is a matrix of parameters. If the error-terms are independent, the matrix $\boldsymbol{\Sigma}_\epsilon$ is diagonal, and if the variance is constant, then

$$\boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2 \boldsymbol{I}),$$

where σ^2 is an unknown parameter.

Since Y and $\mathbf{x}$ are normally distributed then also $(Y \mid \mathbf{x})$ is normally distributed. The expected value of $(Y \mid \mathbf{x})$ is

$$E(Y \mid \mathbf{x}) = \mu_y + \boldsymbol{\sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1}(\mathbf{x} - \boldsymbol{\mu_x})$$

and the variance is

$$Var(Y \mid \mathbf{x}) = \sigma_y^2 - \boldsymbol{\sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\sigma}_{xy}$$

[Johnson and Wichern, 2007]. Then

$$(Y \mid \mathbf{x}) \sim N(\mu_y + \boldsymbol{\sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1}(\mathbf{x} - \boldsymbol{\mu_x}), \sigma_y^2 - \boldsymbol{\sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\sigma}_{xy})$$

Since $\mu_y = 0$ and $\boldsymbol{\mu_x} = \mathbf{0}$ due to centring the data

$$(Y \mid \mathbf{x}) \sim N(\boldsymbol{\sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1} \mathbf{x}, \sigma_y^2 - \boldsymbol{\sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\sigma}_{xy})$$

Since $E(Y \mid \mathbf{x}) = \boldsymbol{\beta}^t\mathbf{x}$, that means

$$\boldsymbol{\beta} = \boldsymbol{\Sigma}_{xx}^{-1}\boldsymbol{\sigma}_{xy} \tag{2.4}$$

and that

$$\sigma^2 = \sigma_y^2 - \boldsymbol{\sigma}_{xy}^t\boldsymbol{\Sigma}_{xx}^{-1}\boldsymbol{\sigma}_{xy}$$

The population coefficient of determination R^2 is the correlation between Y and $\boldsymbol{\beta}^t\boldsymbol{x}$, squared. It can be written as

$$R^2 = Corr(\boldsymbol{\beta}^t\boldsymbol{x}, Y)^2 = \frac{\boldsymbol{\sigma}_{xy}^t\boldsymbol{\Sigma}^{-1}\boldsymbol{\sigma}_{xy}}{\sigma_y^2}$$

This gives

$$\sigma^2 = \sigma_y^2(1 - R^2)$$

2.2.2 Multiresponse Model

The multiresponse case with two responses is similar to the uniresponse case. The vectors $\underset{2\times1}{\mathbf{y}}$ and $\underset{p\times1}{\mathbf{x}}$ is normally distributed:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} \sim N_{p+2}\left(\begin{bmatrix} \boldsymbol{\mu}_y \\ \boldsymbol{\mu}_x \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{yy} & \boldsymbol{\Sigma}_{xy}^t \\ \boldsymbol{\Sigma}_{xy} & \boldsymbol{\Sigma}_{xx} \end{bmatrix}\right)$$

where $\boldsymbol{\mu}_y$ is a vector with the two expected values for the two corresponding responses. And $\underset{2\times2}{\boldsymbol{\Sigma}_{yy}}$ is the covariance matrix for the responses and $\underset{p\times2}{\boldsymbol{\Sigma}_{xy}}$ is the covariance between $\mathbf{x}$ and $\mathbf{y}$. The model used for multiresponse is

$$\mathbf{y}_i = \boldsymbol{\beta}^t\mathbf{x}_i + \boldsymbol{\epsilon}_i, i = 1, 2, ..., n$$

where $\boldsymbol{\beta}$ is a $p \times 2$ matrix with unknown parameters and $\boldsymbol{\epsilon}$ is a 2×1 vector of error terms for the two responses. Alternative we can write the model as

$$\underset{n\times2}{\boldsymbol{Y}} = \underset{n\times p}{\boldsymbol{X}}\underset{p\times2}{\boldsymbol{\beta}} + \underset{n\times2}{\boldsymbol{\epsilon}} \tag{2.5}$$

Again it is interesting to look at the conditional distribution $(\mathbf{y} \mid \mathbf{x})$. The expected value is

$$E(\mathbf{y} \mid \mathbf{x}) = \boldsymbol{\mu}_y + \boldsymbol{\Sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1}(\mathbf{x} - \boldsymbol{\mu}_{\boldsymbol{x}})$$

Since all variables are centred

$$E(\mathbf{y} \mid \mathbf{x}) = \boldsymbol{\Sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1} \mathbf{x}$$

The variance is

$$Var(\mathbf{y} \mid \mathbf{x}) = \boldsymbol{\Sigma}_{yy} - \boldsymbol{\Sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\Sigma}_{xy} = Var(\boldsymbol{\epsilon}) \tag{2.6}$$

The conditional distribution is then

$$(\mathbf{y} \mid \mathbf{x}) \sim N_2(\boldsymbol{\Sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1} \mathbf{x}, \boldsymbol{\Sigma}_{yy} - \boldsymbol{\Sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\Sigma}_{xy})$$

This means that

$$\underset{2\times p}{\boldsymbol{\beta}}{}^t = \boldsymbol{\Sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1}$$

which is similar to the uniresponse 2.4

$$\underset{1\times p}{\boldsymbol{\beta}}{}^t = \boldsymbol{\sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1}$$

The unconditional correlation between the responses is

$$Corr(Y_1, Y_2) = \rho = \frac{\sigma_{y_1 y_2}}{\sigma_{y_1} \sigma_{y_2}}$$

The conditional variance based on eq. 2.6 between Y_1 and Y_2 is

$$Cov(Y_1 | \boldsymbol{x}, Y_2 | \boldsymbol{x}) = Var(\epsilon_1, \epsilon_2) = \sigma_{y_1 y_2}^2 - \boldsymbol{\sigma}_{xy_1}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\sigma}_{xy_2} =$$

$$\rho \sqrt{\sigma_{y_1}^2 \sigma_{y_2}^2} - \boldsymbol{\sigma}_{xy_1}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\sigma}_{xy_2}$$

Using

$$R_1^2 = \frac{\boldsymbol{\sigma}_{xy_1}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\sigma}_{xy_1}}{\sigma_{y_1}^2} \text{ and } R_2^2 = \frac{\boldsymbol{\sigma}_{xy_2}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\sigma}_{xy_2}}{\sigma_{y_2}^2}$$

the conditional correlation is

$$Corr(\epsilon_1, \epsilon_2) = \varrho = \frac{\rho\sqrt{\sigma_{y_1}^2 \sigma_{y_2}^2} - \boldsymbol{\sigma}_{xy_1}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\sigma}_{xy_2}}{\sqrt{\sigma_{y_1}^2 \sigma_{y_2}^2 (1 - R_1^2)(1 - R_2^2)}} \tag{2.7}$$

We will look closer at the conditional correlation in section 3.2.

2.3 Estimation

In our models (see eq. 2.3 and 2.5) $\boldsymbol{\beta}$ is unknown and must be estimated. For this purpose there are many methods to choose among. A natural choice should be the estimator which has the best performance. The performance of an estimator is measured by finding the mean square error(MSE) which can be defined as[Bickel and Doksum, 1977]

$$MSE = E(\hat{\theta} - \theta)^2$$

With some calculations

$$E(\hat{\theta} - \theta)^2 =$$

$$E[(\hat{\theta} - E(\hat{\theta})) - (\theta - E(\hat{\theta})]^2 =$$

$$E[(\hat{\theta} - E(\hat{\theta}))^2 - 2(\hat{\theta} - E(\hat{\theta}))(\theta - E(\hat{\theta})) + (\theta - E(\hat{\theta}))^2] =$$

$$E(\hat{\theta} - E(\hat{\theta}))^2 + E(\theta - E(\hat{\theta}))^2 =$$

$$Var(\hat{\theta}) + [E(\hat{\theta}) - \theta]^2$$

So it is a trade-off between biasedness and variance of the estimator. This is for the one parameter situation. If we have a vector of parameters the MSE is calculated as

$$E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^t] = Var(\hat{\boldsymbol{\theta}}) + (E(\hat{\boldsymbol{\theta}}) - \boldsymbol{\theta})(E(\hat{\boldsymbol{\theta}}) - \boldsymbol{\theta})^t \tag{2.8}$$

To compare two estimators($\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$) a suggestion is to use the trace of MSE. If

$$tr(E[(\hat{\boldsymbol{\theta}}_1 - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}}_1 - \boldsymbol{\theta})^t]) < tr(E[(\hat{\boldsymbol{\theta}}_2 - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}}_2 - \boldsymbol{\theta})^t]) \tag{2.9}$$

then $\hat{\boldsymbol{\theta}}_1$ is said to be a better estimator than $\hat{\boldsymbol{\theta}}_2$. If we set $\hat{\boldsymbol{\theta}}_2 = \hat{\boldsymbol{\beta}}_2 = \mathbf{0}$(the nullmodel as described in section 2.8.1) and $\hat{\boldsymbol{\theta}}_1 = \hat{\boldsymbol{\beta}}_1$ then we get that equation 2.9 turns out to be

$$\begin{aligned}
tr(E[(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta})(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta})^t]) &< tr(E[(\hat{\boldsymbol{\beta}}_2 - \boldsymbol{\beta})(\hat{\boldsymbol{\beta}}_2 - \boldsymbol{\beta})^t]) \\
E[(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta})^t(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta})] &< E[(\mathbf{0} - \boldsymbol{\beta})^t(\mathbf{0} - \boldsymbol{\beta})] \\
\frac{E[(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta})^t(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta})]}{E(\boldsymbol{\beta}^t\boldsymbol{\beta})} &< 1 \\
\frac{E[(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta})^t(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta})]}{\boldsymbol{\beta}^t\boldsymbol{\beta}} &< 1 \qquad (2.10)
\end{aligned}$$

The result in eq. 2.10 can also be used as a measure of estimation error of $\hat{\boldsymbol{\beta}}$. For some methods we can not find the $E(\hat{\boldsymbol{\beta}})$ or $Var(\hat{\boldsymbol{\beta}})$ by calculations. Therefore we need to simulate data to be able to estimate them instead. If the number on the left side of eq 2.10, from now on called the estimation error, is less than 1 then we have an estimator of $\hat{\boldsymbol{\beta}}$ that is better than the nullmodel. To estimate the estimation error we use the following equation

$$\frac{1}{r}\sum_{i=1}^{r} \frac{(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta})^t(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta})}{\boldsymbol{\beta}^t\boldsymbol{\beta}} \tag{2.11}$$

If $\underset{p\times 2}{\boldsymbol{\beta}}$ is a matrix, then eq. 2.10 can not be used to calculate the estimation error. A solution to this is to split up the matrix into two vectors and split up the estimator into two vectors.

$$\underset{p\times 2}{\boldsymbol{\beta}} = \begin{bmatrix} \underset{p\times 1}{\boldsymbol{\beta}_1} & \underset{p\times 1}{\boldsymbol{\beta}_2} \end{bmatrix} \text{ and } \underset{p\times 2}{\hat{\boldsymbol{\beta}}} = \begin{bmatrix} \underset{p\times 1}{\hat{\boldsymbol{\beta}}_1} & \underset{p\times 1}{\hat{\boldsymbol{\beta}}_2} \end{bmatrix}$$

Then for each vector in the matrix with corresponding estimator vector, we calculate the estimation error as in eq. 2.10. We have then split the estimation error in two for the $p \times 2$ parameter matrix.

2.4 Prediction

Prediction is to ”guess” the value of a new response given the corresponding new explanatory values. We must of course assume that there is some dependencies between the Y variables and the **x**-variables, which follows from the models described in 2.3 and 2.5. Since the true $\boldsymbol{\beta}$ is unknown it has to be estimated by some trainingdata, which is from the same distribution as the new observation we want to predict. We predict

$$\hat{Y} = \bar{Y} + \hat{\boldsymbol{\beta}}^t(\boldsymbol{x} - \bar{\boldsymbol{x}}) \tag{2.12}$$

for uniresponse. In 2.12 $\hat{Y}$ is the prediction of Y, $\bar{\boldsymbol{x}}$ is the mean of each explanatory variable from the trainingdata and $\bar{Y}$ is the mean of the response in the training data. For the multiresponse case

$$\hat{\boldsymbol{y}} = \bar{\boldsymbol{y}} + \hat{\boldsymbol{\beta}}^t(\boldsymbol{x} - \bar{\boldsymbol{x}}) \tag{2.13}$$

In 2.13 $\hat{\boldsymbol{y}}$ is a vector with the prediction of each element respectively in **y** and $\hat{\boldsymbol{\beta}}$, $\bar{\boldsymbol{y}}$ and $\bar{\boldsymbol{x}}$ is estimated based on the trainingdata.

2.4.1 Prediction Error, uniresponse

The predicted value will (nearly) always deviate from the true value. This is due to the fact that $\boldsymbol{\beta}$ is estimated and to the error terms(ϵ) in the model. The prediction error is a measure of how well a model predicts any new observations. It is defined as

$$\theta^2 = E(\hat{Y} - Y)^2 \tag{2.14}$$

A practician needs either a test-set or to do cross-validation to be able to estimate the prediction error(see sec 2.5). It is usually done by calculating the Mean Square Error of Prediction(MSEP)

$$MSEP = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

However in simulation studies it is possible to have many replicates, and the expected values in 2.14 can be estimated by the mean. We assume that the expected values of $\boldsymbol{x}$ is zero to simplify the calculations and it is then shown that the prediction error is

$$\theta^2 = E(\hat{Y} - Y)^2 = E[\hat{\boldsymbol{\beta}}^t\boldsymbol{x} - (\boldsymbol{\beta}^t\boldsymbol{x} + \epsilon)]^2 =$$

$$E[(\hat{\boldsymbol{\beta}}^t\boldsymbol{x} - E\hat{\boldsymbol{\beta}}^t\boldsymbol{x}) - (\boldsymbol{\beta}^t\boldsymbol{x} - E\hat{\boldsymbol{\beta}}^t\boldsymbol{x}) - \epsilon]^2 =$$

$$E[(\hat{\boldsymbol{\beta}} - E(\hat{\boldsymbol{\beta}}))^t\boldsymbol{x} - (\boldsymbol{\beta} - E(\hat{\boldsymbol{\beta}}))^t\boldsymbol{x} - \epsilon]^2 =$$

$$E[\boldsymbol{x}^t(\hat{\boldsymbol{\beta}} - E(\hat{\boldsymbol{\beta}}))(\hat{\boldsymbol{\beta}} - E(\hat{\boldsymbol{\beta}}))^t\boldsymbol{x} + (\boldsymbol{\beta} - E\hat{\boldsymbol{\beta}})^t\boldsymbol{x}\boldsymbol{x}^t(\boldsymbol{\beta} - E\hat{\boldsymbol{\beta}}) + \epsilon^2] =$$

$$tr(Var(\hat{\boldsymbol{\beta}})\boldsymbol{\Sigma}_{xx}) + (\boldsymbol{\beta} - E\hat{\boldsymbol{\beta}})^t\boldsymbol{\Sigma}_{xx}(\boldsymbol{\beta} - E\hat{\boldsymbol{\beta}}) + \sigma^2 =$$

$$\sigma^2 + E(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^t\Sigma_{xx}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \tag{2.15}$$

Notice that the prediction error does not need any new observations. This make us able to estimate the prediction error without training-data or cross-validation. The natural estimator of θ^2 is

$$\bar{\hat{\theta}}^2 = \frac{1}{r}\sum_{i=1}^{r}\hat{\theta}^2 = \sigma^2 + \frac{1}{r}\sum_{i=1}^{r}(\hat{\beta} - \beta)^t\Sigma_{xx}(\hat{\beta} - \beta)) \tag{2.16}$$

Where r is the number of replicates. When r is sufficiently high, $\bar{\hat{\theta}}^2$ approaches θ^2.

$$E(\bar{\hat{\theta}}^2)\longrightarrow\theta^2 \text{ and } Var(\bar{\hat{\theta}}^2)\longrightarrow 0$$

Then

$$\bar{\hat{\theta}}^2 \xrightarrow{P} \theta^2, \text{ when } r \longrightarrow \infty$$

Therefore 2.16 is a consistent estimator of θ^2. To find how the prediction error varies between replications we can estimate the standard deviation as

$$\widehat{sd(\hat{\theta}^2)} = \sqrt{\frac{\sum_{j=1}^{r}(\hat{\theta}_j^2 - \bar{\hat{\theta}}^2)^2}{r-1}}, \qquad j = 1, ..., r \tag{2.17}$$

If we look closer to eq. 2.15, we see that the lower limit of the prediction error is σ^2. That happens when $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}$ which yields $\hat{Y} = E(Y|\boldsymbol{x})$. In terms of R^2 and σ_y^2 the lower limit is

$$\sigma^2 = \sigma_y^2(1 - R^2)$$

As R^2 increases, σ^2 decreases and the lower limit of the prediction error decreases. With higher R^2 we could get better predictions.

$$R^2 \longrightarrow 1, \text{ then } \sigma^2 \longrightarrow 0.$$

There is no upper limit to the prediction error. If the prediction error is greater than the prediction error for the Null-Model(as described in section 2.8.1), then it is better to use the mean of the response as a prediction instead. We do not consider or use models that gives a higher prediction error than that of the Null Model.

If we center the variables, the prediction error is not as described in eq. 2.16. We have to multiply with $\frac{n+1}{n}$. This will in most cases(when n is large enough) not change the prediction error much. Therefore we choose to estimate the prediction error as described in eq.2.16.

2.4.2 Prediction Error Multiresponse

The combined prediction error for multiresponse can be written on the form

$$\theta^2 = E(\hat{\boldsymbol{y}} - \boldsymbol{y})^t \boldsymbol{A}^{-1}(\hat{\boldsymbol{y}} - \boldsymbol{y})$$

Where $\boldsymbol{A}$ could be $\boldsymbol{I}, \boldsymbol{\Sigma}_{yy}$ or $\boldsymbol{\Sigma}_{y|x}$[Vining, 1998].

For uniresponse the prediction error was defined as in eq. 2.14. If we use a similar measure of prediction error for multi response it could be

$$\theta^2 = E(\hat{\boldsymbol{y}} - \boldsymbol{y})^t(\hat{\boldsymbol{y}} - \boldsymbol{y}) = \sum_{i=1}^{2} E(\hat{Y}_i - Y_i)^2 = \theta_1^2 + \theta_2^2 \tag{2.18}$$

which is the sum of two prediction errors as we defined it in eq. 2.14. Here $\boldsymbol{A} = \boldsymbol{I}$.

There are two possible options for a covariance matrix for $\boldsymbol{A}$. The unconditional or the conditional covariance matrix for $\boldsymbol{y}$. Using the unconditional covariance matrix the distance is

$$\theta^2 = E(\hat{\boldsymbol{y}} - \boldsymbol{y})^t \boldsymbol{\Sigma}_{yy}^{-1}(\hat{\boldsymbol{y}} - \boldsymbol{y}) =$$

Using the conditional covariance matrix

$$\theta^2 = E(\hat{\boldsymbol{y}} - \boldsymbol{y})^t \boldsymbol{\Sigma}_{y|x}^{-1}(\hat{\boldsymbol{y}} - \boldsymbol{y}) =$$

$$E(\hat{\boldsymbol{y}} - \boldsymbol{y})^t(\boldsymbol{\Sigma}_{yy} - \boldsymbol{\Sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\Sigma}_{xy})^{-1}(\hat{\boldsymbol{y}} - \boldsymbol{y})$$

All 3 options gives us a combined prediction error for the two responses. By using a combined distance of the two responses it is not possible to detect if a prediction method does better in predicting the first response and not as well for the second response. A fourth option is to estimate two prediction errors, one for each response. Then we can use the same prediction error as we did for uniresponse(see eq. 2.16) and we can compare the prediction error for multiresponse directly with the prediction errors for uniresponse models.

2.4.3 Prediction Error and Model Complexity

The prediction error can mainly be explained by three parts. The model error, the estimation error and the error term ϵ [Martens and Næs, 1989]. The error term we can not do anything with. The model error is the underlying bias that is due to not including all components or variables. Adding more and more terms in the model(it can be explanatory variables, components or even the number of responses), making it more complex will cause the model error to decrease(see Figure 2.1). By adding more terms in the model we will increase the number of parameters to estimate from a set of calibration data. As a consequence the estimation error will increase and the prediction error increases. This is what often is called overfitting the model. Using to few terms, the model error is large, but the estimation error is small because there are only a few parameters to be estimated with the available calibration data(underfitting). We should not include to many predictors or to few. To find the right number of components or predictors we have to find the point where the estimation error and the model error balance each other to find the minimum prediction error.

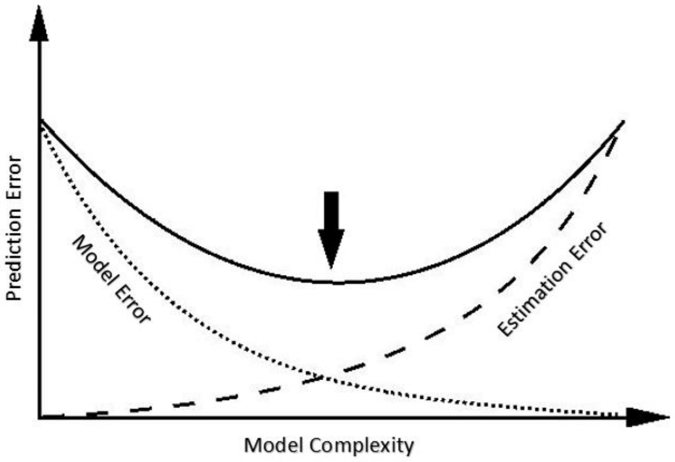

Figure 2.1: How the model complexity affects the prediction error

We can always lower the prediction error by including more observations. But this might not be possible or to expensive. Later it is shown that in some situation, the effect of adding more observations is small.

2.5 Validation

If we don't know the value of the true parameters, we have to estimate the prediction error in another way than we did in eq. 2.16. As an estimator of the prediction error we use Mean Square Error of Prediction(MSEP) the formula will vary slightly for different validation methods.

2.5.1 Test set

The basic idea is that we split the observations in two groups. One of the groups of observations is used to fit the model, called training data. The a observations in the second group, usually called a test set, is predicted using the fitted model. We then estimate MSEP as

$$MSEP_{test} = \frac{1}{a}\sum_{i=1}^{a}(Y_i - \hat{Y_i})^2$$

where Y_i is a new observation from the test set and $\hat{Y_i}$ is the predicted value of the new observation when using the model fitted with training data. This requires that we have enough observations to fit the model well and enough left to get a good estimate of the prediction error.

2.5.2 Cross Validation(CV)

When there are too few observations to split the data in two groups we can do cross-validation instead. We will consider the Leave One Out Cross Validation. We leave out one observation and fit the model with the remaining observations. Then we predict the observations left out and estimate it's prediction error. We repeat the procedure but leave out another observations.

This we do for all observations and the MSEP can be estimated as

$$MSEP_{CV} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

2.6 Relevant Components

A component is a linear combination of the explanatory variables. To be relevant it has to have non-zero correlation to the response. From [Næs and Helland, 1993] they define a component to be relevant if there is some eigenvector($\boldsymbol{e}_j$, see eq. 2.2) of $\boldsymbol{\Sigma}_{xx}$ where

$$\boldsymbol{e}_j^t \boldsymbol{\sigma}_{xy} \neq 0 \tag{2.19}$$

These eigenvectors are called relevant eigenvectors and their corresponding eigenvalues are called relevant eigenvalues. A relevant component is the linear combination $\boldsymbol{e}_j^t \boldsymbol{x}$. The eigenvectors where $\boldsymbol{e}_j^t \boldsymbol{\sigma}_{xy} = 0$ are called the irrelevant eigenvectors and its corresponding eigenvalues are the irrelevant eigenvalues. The irrelevant components are the linear combination $\boldsymbol{e}_j^t \boldsymbol{x}$ of these eigenvectors.

If we have m relevant components we can express these as

$$\boldsymbol{z} = \boldsymbol{R}^t \boldsymbol{x}$$

where $\underset{p\times m}{\boldsymbol{R}}$ consist of the m relevant eigenvectors, not necessarily the eigenvectors with largest eigenvalues. The irrelevant components can be expressed as

$$\boldsymbol{v} = \boldsymbol{U}^t \boldsymbol{x}$$

where $\underset{p\times(p-m)}{\boldsymbol{U}}$ consist of the $p - m$ irrelevant eigenvectors. Then

$$\boldsymbol{\Sigma}_{xx} = \boldsymbol{R}\boldsymbol{\Lambda}_m \boldsymbol{R}^t + \boldsymbol{U}\boldsymbol{\Lambda}_{p-m}\boldsymbol{U}^t$$

Where $\mathbf{\Lambda}_m$ is a diagonal matrix with the m relevant eigenvalues and $\mathbf{\Lambda}_{p-m}$is a diagonal matrix with the $p-m$ irrelevant eigenvalues. We have divided the space spanned by $\mathbf{\Sigma}_{xx}$ into two orthogonal subspaces spanned by $\boldsymbol{U}$ and $\boldsymbol{R}$, where one spans the relevant space($\boldsymbol{R}$) and the other the irrelevant space($\boldsymbol{U}$).

2.7 Collinearity

When the columns of $\mathbf{X}$ are linear dependent or nearly linear dependent, then the $\mathbf{X}$-matrix is said to be collinear(or multicollinear) [Martens and Næs, 1989]. The set $(\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_p)$ is said to be linear dependent if there exist weights $c_1, c_2, ..., c_p$ that are not all zero, such that

$$c_1\boldsymbol{x}_1 + c_2\boldsymbol{x}_2 + ... + c_p\boldsymbol{x}_p = \mathbf{0}$$

[Lay, 2012]. When $n < p$ the matrix $\boldsymbol{X}$ does not have full rank and the columns in $\boldsymbol{X}$ is linearly dependent and therefore collinearity is present. This causes a problem for some prediction methods. One example is the Least Squares Regression as described in section 2.8.2. The method can not be used when $n < p$ because $\boldsymbol{X}^t\boldsymbol{X}$ is not invertible.

Another problem is when the columns in $\boldsymbol{X}$ are nearly collinear

$$\sum_{i=1}^{p} c_i\boldsymbol{x}_i \approx 0$$

When this problem occur, the ratio between the largest and smallest eigenvalue is large. For Least Squares Regression, the smallest eigenvalues in the matrix

$$(\boldsymbol{X}^t\boldsymbol{X})^{-1} = \sum_{i=1}^{p} \frac{e_i e_i^t}{\hat{\lambda}_i}$$

causes problems. The smallest eigenvalues has the greatest effect on the matrix above. A small change in these eigenvalues will change the matrix completely. As a result we get a large variation. Many methods(Principal Component regression, Partial Least Squares are a few examples) handle this problem by creating some new variables(less then p) that are linear combinations of the original variables.

2.8 Prediction Methods

It is impossible to find a uniform best estimator of $\boldsymbol{\beta}$ or a method that always gives the best prediction of a new observation. There exists several methods to estimate $\boldsymbol{\beta}$, some are presented in the sections below.

Some prediction methods reduce the number of explanatory variables by using some linear combinations of the explanatory variables, by creating a transformation matrix R with rank $k < n$ and $k < p$. Let

$$\underset{n\times k}{Z} = \underset{n\times p}{\boldsymbol{X}} \underset{p\times k}{R}$$

and use Z instead of X. The matrix Z will hopefully contain much of the information about Y which already is in $\boldsymbol{X}$. We assume the model

$$\boldsymbol{y} = Z\boldsymbol{\beta}_z + \boldsymbol{\epsilon}_z$$

We estimate $\hat{\boldsymbol{\beta}}_z$ by Least Squares method.

$$\hat{\boldsymbol{\beta}}_z = (Z^t Z)^{-1} Z^t \boldsymbol{y}$$

$$= (R^t \boldsymbol{X}^t \boldsymbol{X} R^t)^{-1} R^t \boldsymbol{X}^t \boldsymbol{y}$$

We transform back by

$$\hat{\boldsymbol{\beta}} = R\hat{\boldsymbol{\beta}}_z = R(\underbrace{R^t \boldsymbol{X}^t \boldsymbol{X} R}_{k\times k})^{-1} R^t \boldsymbol{X}^t \boldsymbol{y}$$

The matrix R is dependent on the method used and the number of components(k). In Partial Least Squares R is dependent on Y, which means that it is impossible or extremely difficult to calculate the expectation and variance of $\hat{\boldsymbol{\beta}}$ which is needed to find the prediction error(see section 2.4.1).

There are several methods to estimate $\boldsymbol{\beta}$. A few examples are Principal Component Regression, Ridge Regression and Lasso [Hastie et al., 2001]. All these methods will give the same result, whether we model as uniresponse or multireponse. One of the few methods that will give different result, whether we model as uniresponse or multiresponse, is Partial Least Squares.

2.8.1 'The Null Model'

In some situations when there is no or little correlation between Y and $\boldsymbol{x}$ it might be a good idea to predict

$$\hat{Y} = \bar{Y}$$

by letting $\hat{\boldsymbol{\beta}} = \mathbf{0}$. This means that we do not consider any information that might be in the X-variables. The prediction error for the Null Model is

$$E(\bar{Y} - Y)^2 = \sigma_y^2 \tag{2.20}$$

when we exclude the $\frac{n+1}{n}$ term of the prediction error.

2.8.2 Least Squares Regression(LS)

The Least Squares Regression is sort of the opposite of the Null Model, because it uses all information in $\mathbf{X}$. If $k = p$ and rules for inverting and transposing [Lay, 2012] we have

$$\hat{\boldsymbol{\beta}} = RR^{-1}(\boldsymbol{X}^t\boldsymbol{X})^{-1}(R^t)^{-1}R^t\boldsymbol{X}^t\boldsymbol{y} = (\boldsymbol{X}^t\boldsymbol{X})^{-1}\boldsymbol{X}^t\boldsymbol{y} \tag{2.21}$$

the least squares estimator of $\boldsymbol{\beta}$. It minimizes the residual sums of squares [Mardia et al., 1982]. It can also bee shown that the Least Squares estimator is an unbiased estimator of $\boldsymbol{\beta}$.

$$E(\hat{\boldsymbol{\beta}}) = (\boldsymbol{X}^t\boldsymbol{X})^{-1}\boldsymbol{X}^tE(\boldsymbol{y}) = ((\boldsymbol{X}^t\boldsymbol{X})^{-1}\boldsymbol{X}^t(\boldsymbol{X}\boldsymbol{\beta} + E(\boldsymbol{\epsilon})) = \boldsymbol{\beta}$$

And the variance of $\hat{\boldsymbol{\beta}}$ is

$$Var(\hat{\boldsymbol{\beta}}) = (\boldsymbol{X}^t\boldsymbol{X})^{-1}\boldsymbol{X}^tVar(\boldsymbol{y})\boldsymbol{X}(\boldsymbol{X}^t\boldsymbol{X})^{-1}$$

$$= (\boldsymbol{X}^t\boldsymbol{X})^{-1}\boldsymbol{X}^t\boldsymbol{I}\sigma^2\boldsymbol{X}(\boldsymbol{X}^t\boldsymbol{X})^{-1} = \sigma^2(\boldsymbol{X}^t\boldsymbol{X})^{-1}$$

The MSE of $\boldsymbol{\beta}$(eq. 2.8) for LS is the same as the variance of the estimator(due to unbiasedness). The trace of a matrix is equal to the sum of its eigenvalues [Lay, 2012]. Then

$$Tr(Var(\hat{\boldsymbol{\beta}})) = Tr(\sigma^2(\boldsymbol{X}^t\boldsymbol{X})^{-1}) = \sigma^2 Tr(\hat{\boldsymbol{\Lambda}}^{-1}) = \sigma^2\sum_{i=1}^{p}\frac{1}{\hat{\lambda}_i} \tag{2.22}$$

Using this result for the estimation error in eq. 2.10 and using the notation in sec 3.1 we find the estimation error to be

$$\frac{\sigma^2\sum_{i=1}^{p}\frac{1}{\hat{\lambda}_i}}{\boldsymbol{\beta}^t\boldsymbol{\beta}} = \frac{\sigma^2\sum_{i=1}^{p}\frac{1}{\hat{\lambda}_i}}{\boldsymbol{\sigma}^t_{xy}(\boldsymbol{\Sigma}^{-1}_{xx})\boldsymbol{\Sigma}^{-1}_{xx}\boldsymbol{\sigma}_{xy}} = \frac{(1-R^2)\sum_{i=1}^{p}\frac{1}{\hat{\lambda}_i}}{\boldsymbol{\sigma}^t_{zy}(\boldsymbol{\Lambda}^{-1})^2\boldsymbol{\sigma}_{zy}} \tag{2.23}$$

When we have eigenvalues that falls quikly, meaning we have many small eigenvalues and a few large ones, the estimation by Least Squares method of $\boldsymbol{\beta}$ has high variation(see sec 2.7).

If there is a linear dependency meaning that $\mathbf{X}$ has a rank < p the Least Squares Estimator can not be estimated as in eq. 2.21, because $\boldsymbol{X}^t\boldsymbol{X}$ is not invertible. This happens when $n < p$.

2.8.3 Partial Least Squares Regression(PLSR)

The PLSR-algorithm tries to find the components that maximizes the covariance between the response and explanatory variables. The algorithm will give different results for uniresponse and multiresponse due to that $\boldsymbol{Y}$ affects the modeling of $\boldsymbol{X}$(it influences the matrix R). Except for when p components are included. Then both the uniresponse PLSR and multiresponse PLSR will give the Least Squares solution. If zero components are included the result is the Null Model.

Uniresponse

There are several different PLSR algorithms. The original PLSR algorithm was developed by Wold [Martens and Næs, 1989] and is presented here. The algorithm can be divided into several steps.

1. All variables(both explanatory($\boldsymbol{X}_0$) and response($\boldsymbol{y}_0$) are centred and the number of components to find is set to K_{max}. It should at least be higher then the number of phenomena we expect to find in $\boldsymbol{X}$. The following 5 steps(a - e) are repeated K_{max} times.

   (a) Find loading weights $\boldsymbol{w}_k$ as

   $$\hat{\boldsymbol{w}}_k = \boldsymbol{X}_{k-1}^t \boldsymbol{y}_{k-1}$$

   and scale the loading weights to length 1.

   (b) Estimate the scores $\hat{\boldsymbol{t}}_k$ by

   $$\hat{\boldsymbol{t}}_k = \boldsymbol{X}_{k-1} \hat{\boldsymbol{w}}_k$$

   (c) Estimate the X-loadings $\hat{\boldsymbol{p}}_k$ by

   $$\hat{\boldsymbol{p}}_k = \frac{\boldsymbol{X}_{k-1}^t \hat{\boldsymbol{t}}_k}{\hat{\boldsymbol{t}}_k^t \hat{\boldsymbol{t}}_k}$$

(d) Estimate the Y-loadings $\hat{q}_k$ by

$$\hat{q}_k = \frac{\boldsymbol{y}_{k-1}^t \hat{\boldsymbol{t}}_k}{\hat{\boldsymbol{t}}_k^t \hat{\boldsymbol{t}}_k}$$

(e) Create new $\boldsymbol{X}$ and $\boldsymbol{y}$ residuals by subtracting the estimated effect and set these as $\boldsymbol{X}_k$ and $\boldsymbol{y}_k$

$$\hat{\boldsymbol{E}} = \boldsymbol{X}_{k-1} - \hat{\boldsymbol{t}}_k \hat{\boldsymbol{p}}_k^t = \boldsymbol{X}_k$$

$$\hat{\boldsymbol{f}} = \boldsymbol{y}_{k-1} - \hat{\boldsymbol{t}}_k \hat{q}_k = \boldsymbol{y}_k$$

$$k = k + 1$$

2. Determine the number of components(K) to be included, usually by using some sort of validation.

3. Compute $\hat{\boldsymbol{\beta}}$ with K components

$$\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{W}}(\hat{\boldsymbol{P}}^t \hat{\boldsymbol{W}})^{-1} \hat{\boldsymbol{q}}$$

where

$$\hat{\boldsymbol{W}} = [\hat{\boldsymbol{w}}_1 ... \hat{\boldsymbol{w}}_K]$$

$$\hat{\boldsymbol{P}} = [\hat{\boldsymbol{p}}_1 ... \hat{\boldsymbol{p}}_K]$$

$$\hat{\boldsymbol{q}} = [\hat{q}_1 ... \hat{q}_K]$$

Multiresponse

The algorithm for multiresponse is almost the same as for uniresponse. We replace the vectors $\boldsymbol{f}, \boldsymbol{y}$ and $\boldsymbol{q}$ with matrices and introduce the vector $\hat{\boldsymbol{u}}_k$ that replaces the vector $\boldsymbol{y}_{k-1}$ when finding the loading weights. In the first iteration $\hat{\boldsymbol{u}}_k$ is given some starting values(ex one of the columns in $\boldsymbol{Y}$). The following steps are repeated until $\hat{\boldsymbol{t}}_k$ converges.

1. $$\hat{\boldsymbol{w}}_k = \boldsymbol{X}_{k-1}^t \hat{\boldsymbol{u}}_k \text{ and scale it to length 1.}$$

2. estimate the scores, X-loadings and Y-loadings as for uniresponse.

3. check if $\hat{\boldsymbol{t}}_k$ has converged. If not estimate $\hat{\boldsymbol{u}}_k$ by

$$\hat{\boldsymbol{u}}_k = \boldsymbol{Y}_{k-1}\hat{\boldsymbol{q}}_k(\hat{\boldsymbol{q}}_k^t\hat{\boldsymbol{q}}_k)^{-1}$$

   and go back to step 1.

When $\hat{\boldsymbol{t}}_k$ converges we can create $\boldsymbol{X}$ and $\boldsymbol{Y}$ residuals as we did for uniresponse. And repeat the procedure K_{max} times. Then $\boldsymbol{\beta}$ can be estimated the same way as for uniresponse.

$$\underset{p\times 2}{\hat{\boldsymbol{\beta}}} = \hat{\boldsymbol{W}}(\hat{\boldsymbol{P}}^t\hat{\boldsymbol{W}})^{-1}\hat{\boldsymbol{Q}}$$

If we include only one component the PLSR-solution is

$$\underset{p\times 2}{\hat{\boldsymbol{\beta}}} = \underbrace{\hat{\boldsymbol{w}}(\hat{\boldsymbol{p}}^t\hat{\boldsymbol{w}})}_{p\times 1}{}^{-1}\underset{1\times 2}{\hat{\boldsymbol{q}}} = \underbrace{\hat{\boldsymbol{w}}(\hat{\boldsymbol{p}}^t\hat{\boldsymbol{w}})}_{p\times 1}{}^{-1}[\hat{q}_1 \quad \hat{q}_2] = [\hat{\boldsymbol{\beta}}_1 \quad \frac{\hat{q}_2}{\hat{q}_1}\hat{\boldsymbol{\beta}}_1]$$

Which means that the two $\hat{\boldsymbol{\beta}}$'s are parallel when one component is included. Since $(\hat{\boldsymbol{p}}^t\hat{\boldsymbol{w}})^{-1}$ is a scalar, $\underset{p\times 2}{\hat{\boldsymbol{\beta}}}$ can be written as

$$\underset{p\times 2}{\hat{\boldsymbol{\beta}}} = \hat{\boldsymbol{w}}[\frac{\hat{q}_1}{\hat{\boldsymbol{p}}^t\hat{\boldsymbol{w}}} \quad \frac{\hat{q}_2}{\hat{\boldsymbol{p}}^t\hat{\boldsymbol{w}}}] = \hat{\boldsymbol{w}}[k_1 \quad k_2] \tag{2.24}$$

for one component.

For simplicity we will call uniresponse PLSR, PLS1 and multireponse PLSR, PLS2. The PLS-algorithm used in the simulation study is the Kernel PLS[Dayal and MacGregor, 1997].

2.9 Comparing $\hat{\boldsymbol{\beta}}$'s from PLS1 and PLS2

Later in this study we had a suspicion that the PLS2 estimators is an average or linear combination of PLS1 estimators. We let $\hat{\boldsymbol{\beta}}_{PLS2,Y_i}$ be $\hat{\boldsymbol{\beta}}$ when using PLS2 as estimator for the i-th response. Similar for PLS1. If we fit two models, one for each PLS2 estimator

$$\hat{\boldsymbol{\beta}}_{PLS2,Y_i} = \alpha_1 \hat{\boldsymbol{\beta}}_{PLS1,Y_1} + \alpha_2 \hat{\boldsymbol{\beta}}_{PLS1,Y_2} + \boldsymbol{\epsilon}, \qquad i = 1, 2 \tag{2.25}$$

with LS for each component included, we should be able to detect if PLS2 $\hat{\boldsymbol{\beta}}$'s is an average or linear combination of PLS1 $\hat{\boldsymbol{\beta}}$'s.

2.10 Analysis of Variance(ANOVA)

In the simulations we have several parameters which decides the distribution of the variables. We let each parameter have one high and one low value, hence we consider the parameters as factors with two levels each. To investigate which of the parameters that affect the prediction error the most, an Analysis of Variance(ANOVA) can be performed. In a more general situation lets say that we only have one factor with a levels and one response. We can use the model

$$y_{ij} = \mu + \tau_i + \epsilon_{ij}, \qquad \begin{cases} i = 1, 2, ..., a \\ j = 1, 2, ..., n \end{cases}$$

where y_{ij} is the response for the ith factor level and observation number j. It should not be confused with the response Y or $\boldsymbol{y}$ in the models described in sec. 2.2.1 and sec. 2.2.2. The parameter μ is the over-all mean. In other words the expected mean of all the observations. And τ_i is the effect of treatment or factor level i. This is a single-factor analysis of variance(ANOVA)[Montgomery, 2013]. The model errors(ϵ_{ij}) are assumed to be normally and independent distributed with mean 0 and variance σ^2 (not the same as σ^2 mentioned in sec. 2.2.1). The variance is assumed to be constant for all factor levels. We have the restriction that

$$\sum_{i=1}^{a} \tau_i = 0$$

What we want to test is if the factor has any effect at all.

$$H_0 : \tau_1 = \tau_2 = ... = \tau_a = 0$$

$$H_1 : \tau_i \neq \tau_j \text{ for at least one pair where } j \neq i$$

This is done by using a F-test.

$$F = \frac{MSG}{MSE} \sim F_{a-1,N-a}$$

Where N is the total number of observations, MSE is the Mean Sum Squared Error and MSG is the Mean Sum Squared Group.

$$MSE = \frac{\sum\limits_{i=1}^{a} \sum\limits_{j=1}^{n_i} (y_{ij} - \bar{y_{i.}})^2}{N - a}$$

And

$$MSG = \frac{\sum\limits_{i=1}^{a} n_i(\bar{y}_{i.} - \bar{y_{..}})^2}{a - 1}$$

Where n_i is the number of observations of factor level i.

For two factors, τ and κ with a and b levels, we have the model

$$y_{ijk} = \mu + \tau_i + \kappa_j + (\tau\kappa)_{ij} + \epsilon_{ijk}, \qquad \begin{cases} i = 1, 2, ..., a \\ j = 1, 2, ..., b \\ k = 1, 2, ..., n \end{cases}$$

where $(\tau\kappa)_{ij}$ is the interaction between the two factors(two-factor interaction). Meaning that the effect of κ is dependent on the level of τ. The effect of κ is different for the for different levels of τ. We have the restriction that

$$\sum_{i=1}^{a} \tau_i = 0, \qquad \sum_{j=1}^{b} \kappa_j = 0, \qquad \sum_{i=1}^{a} (\tau\kappa)_{ij} = \sum_{j=1}^{b} (\tau\kappa)_{ij} = 0$$

We can extend the model to l factors, and add more complex interactions up to l-factor interaction.

Chapter 3

Simulation

To figure out what structures works better than others we need a method of simulating data where we know the true structure. The R-package Simrel gives us the tool to do exactly that [Sæbø, 2015]. With only a few parameters we can decide the dimensions of the **Y** and the **X** matrix and their simulated distribution with only a few parameters.

3.1 The parameters in the simulation package for uniresponse

In the uniresponse case we must specify some parameter values. Those are listed in Table 3.1.

Table 3.1: *Simulation parameters with explanation*

Parameter	Explanation
n	Number of observations
p	Number of explanatory variables
m	Number of relevant components
q	Number of relevant predictors
γ	Level of collinearity in $\boldsymbol{\Sigma}_{xx}$
relpos	Vector with position for relevant components
R^2	The correlation between Y and $\boldsymbol{\beta}^t\boldsymbol{x}$

The Simrel package simulate data in the following way[Sæbø et al., 2015]. We let the expected values be zero($\mu_y = 0$ and $\boldsymbol{\mu}_x = \mathbf{0}$). The variance of Y, σ_y^2 is 1. The matrix $\underset{p\times p}{\boldsymbol{E}}$ consist of the p orthonormal eigenvectors(see eq. 2.2) for $\boldsymbol{\Sigma}_{xx}$. Let

$$\boldsymbol{z} = \boldsymbol{E}^t\boldsymbol{x}$$

Since E has full rank(p) we can always rotate $\boldsymbol{z}$ back to $\boldsymbol{x}$ by $E\boldsymbol{z} = EE^t\boldsymbol{x} = \boldsymbol{x}$, without loosing any information in $\boldsymbol{x}$.

$$Var(\boldsymbol{z}) = \boldsymbol{E}^tVar(\boldsymbol{x})\boldsymbol{E} = \boldsymbol{E}^t\boldsymbol{\Sigma}_{xx}\boldsymbol{E} = \boldsymbol{\Lambda}$$

$\boldsymbol{\Lambda}$ is a diagonal matrix with the eigenvalues of $\boldsymbol{\Sigma}_{xx}$ on the diagonal. The eigenvalues are decided by the simulation parameter γ and is calculated with the function

$$\lambda_j = e^{-\gamma(j-1)}, \qquad j = 1...p$$

The first eigenvalue is $e^{-\gamma(1-1)} = 1$. If γ has a high value, then the eigenvalues fall quickly and the collinearity between the X variables is high. Figure 3.1 gives an example of how quickly the eigenvalues decline for two different γ's

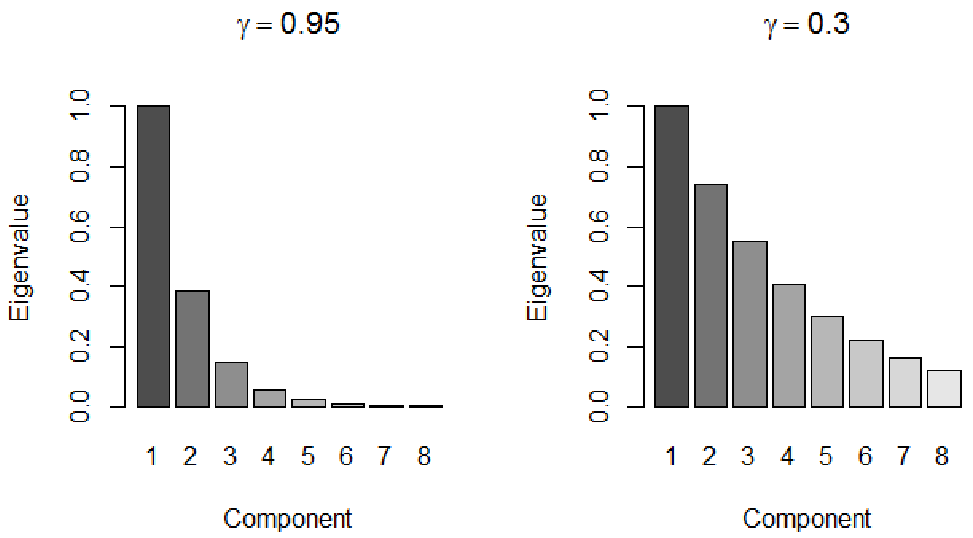

Figure 3.1: *The Eigenvalues for each eigenvector or component for two different γ's. We can see that the Eigenvalues decline much faster for a higher γ.*

Further we have that

$$Cov(\boldsymbol{z}, Y) = \boldsymbol{\sigma}_{zy} = \boldsymbol{E}^t Cov(\boldsymbol{x}, Y) = \boldsymbol{E}^t \boldsymbol{\sigma}_{xy} = \begin{bmatrix} \boldsymbol{e}_1^t \boldsymbol{\sigma}_{xy} \\ \boldsymbol{e}_2^t \boldsymbol{\sigma}_{xy} \\ \vdots \\ \boldsymbol{e}_p^t \boldsymbol{\sigma}_{xy} \end{bmatrix}$$

If $e_i^t \sigma_{xy} = 0$ for some i, it's an irrelevant component. The number of $\boldsymbol{e}_i^t \sigma_{xy} \neq 0$ is m and the parameter *relpos* tells us which ones of these that are not zero. To attain values on the m elements in $\boldsymbol{\sigma}_{zy}$ that are not zero, the coefficient of determination(R^2) is used.

$$R^2 = \frac{\boldsymbol{\sigma}_{xy}^t \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\sigma}_{xy}}{\sigma_y^2} = \frac{\boldsymbol{\sigma}_{zy}^t \boldsymbol{\Lambda}^{-1} \boldsymbol{\sigma}_{zy}}{\sigma_y^2} = \boldsymbol{\sigma}_{zy}^t \boldsymbol{\Lambda}^{-1} \boldsymbol{\sigma}_{zy} = \sum_{i \in relpos} \frac{(\boldsymbol{e}_i^t \boldsymbol{\sigma}_{xy})^2}{\lambda_i} \tag{3.1}$$

The simulation draws a random vector ($\boldsymbol{\sigma}_{zy}$) with zeros on the irrelevant positions and values on the relevant positions so that eq. 3.4 holds. Then

$$\begin{bmatrix} Y \\ \mathbf{z} \end{bmatrix} \sim N_{p+1}\left(\begin{bmatrix} 0 \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} 1 & \boldsymbol{\sigma}_{zy}^t \\ \boldsymbol{\sigma}_{zy} & \boldsymbol{\Lambda} \end{bmatrix} = \boldsymbol{\Sigma}_{zy}\right) \tag{3.2}$$

The program draws $n \times (p+1)$ standard independent normal distributed data an put these in $\boldsymbol{U}$. Let $\boldsymbol{\Sigma}_{zy}^{1/2}$ be some square root matrix of $\boldsymbol{\Sigma}_{zy}$ so that $(\boldsymbol{\Sigma}_{zy}^{1/2})^t\boldsymbol{\Sigma}_{zy}^{1/2} = \boldsymbol{\Sigma}_{zy}$. Then we compute $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}_{zy}^{1/2}$. The rows of $\boldsymbol{W}$ will have the distribution as in eq. 3.2. To obtain the correct number of relevant predictors(q) the matrix $\mathbf{W}$ is rotated. We will not go into any further details of how that is done here.

All possible values for these simulation parameters span the 7-dimensional parameter space called Ω. If we pick one value for all parameters in table 3.1 we are in a certain point in Ω, called ω.

3.2 The parameters in the simulation package for multiresponse

A similar simulation package can be used to simulate data for multiresponse(Solve Sæbø ,personal communication, February 20, 2015). Many of the parameters as explained in the uniresponse simulation in table 3.1 are also used for the multiresponse simulation. The multiresponse simulation parameters are presented in Table 3.2.

We let the expected means of the response and explanatory variables be zero, $\boldsymbol{\mu}_x = \mathbf{0}$ and $\boldsymbol{\mu}_y = \mathbf{0}$. Let

$$\underset{p\times 1}{\boldsymbol{z}} = \boldsymbol{E}^t\boldsymbol{x}$$

as we did for simulation for uniresponse(Sec 3.1). We would like to find the variance matrix for $[Y_1 \quad Y_2 \quad \boldsymbol{z}^t]^t$. The matrix $\boldsymbol{\Sigma}_{zz}$ is obtained from

Table 3.2: *The simulation parameters for two responses*

Parameter	**Explanation**
n	the number of observations
p	the number of explanatory variables
q	a vector with 3 elements(a,b,c). a - the number of relevant predictors for the first response, b - the number of predictors for the second response and c - the number of relevant predictors that are common for both responses.
γ	Level of collinearity in $\boldsymbol{\Sigma}_{xx}$
relpos	Two vectors with positions of relevant components for each response.
R^2	A vector with 2 elements. Corr$(Y, \boldsymbol{\beta}^t \boldsymbol{x})$ for each response.
(ρ, ϱ)	A vector with 2 elements. The simulation parameter ρ is the unconditional correlation between the two responses, Corr(Y_1, Y_2). And ϱ is the conditional correlation between the two responses. $Corr(Y_1\|\boldsymbol{x}, Y_2\|\boldsymbol{x}) = Corr(\epsilon_1, \epsilon_2)$, see eq. 2.7

the simulation paramter γ the same way as for uniresponse. For $\boldsymbol{\Sigma}_{yy}$ we let $\sigma_{y_1} = \sigma_{y_2} = 1$, hence the unconditional covariance between Y_1 and Y_2 is the unconditional correlation. Let ρ denote this correlation.

$$\boldsymbol{\Sigma}_{yy} = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$

The covariance between $\boldsymbol{z}$ and each response i is

$$Cov(\boldsymbol{z}, Y_i) = \boldsymbol{\sigma}_{zy_i} = \boldsymbol{E}^t\boldsymbol{\sigma}_{xy_i} = \begin{bmatrix} \boldsymbol{e}_1^t\boldsymbol{\sigma}_{xy_i} \\ \boldsymbol{e}_2^t\boldsymbol{\sigma}_{xy_i} \\ \vdots \\ \boldsymbol{e}_p^t\boldsymbol{\sigma}_{xy_i} \end{bmatrix}, \qquad i = 1, 2$$

The first vector in the simulation parameter *relpos* decides which of the $\boldsymbol{e}_k^t\boldsymbol{\sigma}_{xy_1}$ that should not be equal to zero for the first response. The second vector in *relpos* decide which of the $\boldsymbol{e}_k^t\boldsymbol{\sigma}_{xy_2}$ that should not be equal to zero for the second response. The rest of the $\boldsymbol{e}_k^t\boldsymbol{\sigma}_{xy_i}$ is zero.

For those $\boldsymbol{e}_k^t\boldsymbol{\sigma}_{xy_i} \neq 0$ the values are chosen randomly under some restrictions. The correlation between $\boldsymbol{z}$ and Y_1

$$R_1^2 = \frac{\boldsymbol{\sigma}_{xy_1}^t\boldsymbol{\Sigma}_{xx}^{-1}\boldsymbol{\sigma}_{xy_1}}{\sigma_{y_1}^2} = \frac{\boldsymbol{\sigma}_{zy_1}^t\boldsymbol{\Lambda}^{-1}\boldsymbol{\sigma}_{zy_1}}{\sigma_{y_1}^2} = \boldsymbol{\sigma}_{zy_1}^t\boldsymbol{\Lambda}^{-1}\boldsymbol{\sigma}_{zy_1} \tag{3.3}$$

Similar for Y_2

$$R_2^2 = \boldsymbol{\sigma}_{zy_2}^t\boldsymbol{\Lambda}^{-1}\boldsymbol{\sigma}_{zy_2} \tag{3.4}$$

We let

$$R_{12} = \boldsymbol{\sigma}_{zy_1}^t\boldsymbol{\Lambda}^{-1}\boldsymbol{\sigma}_{zy_2}$$

The conditional variance $Var(Y_1, Y_2|\boldsymbol{z})$ is then

$$\boldsymbol{\Sigma}_{yy} - \boldsymbol{\Sigma}_{zy}^t\boldsymbol{\Lambda}^{-1}\boldsymbol{\Sigma}_{zy} = \tag{3.5}$$

$$\begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} - \begin{bmatrix} \boldsymbol{\sigma}_{zy_1}^t \\ \boldsymbol{\sigma}_{zy_2}^t \end{bmatrix} \boldsymbol{\Lambda}^{-1} \begin{bmatrix} \boldsymbol{\sigma}_{zy_1} & \boldsymbol{\sigma}_{zy_2} \end{bmatrix} =$$

$$\begin{bmatrix} 1-\boldsymbol{\sigma}_{zy_1}^t\boldsymbol{\Lambda}^{-1}\boldsymbol{\sigma}_{zy_1} & \rho-\boldsymbol{\sigma}_{zy_1}^t\boldsymbol{\Lambda}^{-1}\boldsymbol{\sigma}_{zy_2} \\ \rho-\boldsymbol{\sigma}_{zy_1}^t\boldsymbol{\Lambda}^{-1}\boldsymbol{\sigma}_{zy_2} & 1-\boldsymbol{\sigma}_{zy_2}^t\boldsymbol{\Lambda}^{-1}\boldsymbol{\sigma}_{zy_2} \end{bmatrix} = \begin{bmatrix} 1-R_1^2 & \rho-R_{12} \\ \rho-R_{12} & 1-R_2^2 \end{bmatrix}$$

The conditional correlation matrix is

$$Corr(\boldsymbol{y}|\boldsymbol{z}) = \begin{bmatrix} 1 & \frac{\rho-R_{12}}{\sqrt{(1-R_1^2)(1-R_2^2)}} \\ \frac{\rho-R_{12}}{\sqrt{(1-R_1^2)(1-R_2^2)}} & 1 \end{bmatrix} \tag{3.6}$$

Where ϱ is the conditional correlation.

$$\varrho = Corr(\epsilon_1, \epsilon_2) = \frac{\rho - R_{12}}{\sqrt{(1-R_1^2)(1-R_2^2)}} \tag{3.7}$$

The three equations 3.3, 3.4 and 3.7 are used to draw values for $\boldsymbol{\sigma}_{zy_1}$ and $\boldsymbol{\sigma}_{zy_2}$. We now have constructed a variance matrix for $[Y_1 \quad Y_2 \quad \boldsymbol{z}^t]^t$. The distribution is

$$\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{z} \end{bmatrix} \sim N_{2+p}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{yy} & \boldsymbol{\Sigma}_{zy}^t \\ \boldsymbol{\Sigma}_{zy} & \boldsymbol{\Lambda} \end{bmatrix} = \boldsymbol{\Sigma} \right) \tag{3.8}$$

The program draws $n \times (p+2)$ standard independent normally distributed data and put these in the matrix $\boldsymbol{U}$. As for uniresponse $\boldsymbol{\Sigma}^{1/2}$ is some square root matrix of $\boldsymbol{\Sigma}$. By computing $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}^{1/2}$ The n rows in $\boldsymbol{W}$ will have the distribution in eq. 3.8. The two first columns in $\boldsymbol{W}$ is the two responses($\boldsymbol{Y}$) and the p columns left is $\boldsymbol{X}$.

Similar as with uniresponse simulation we define that all simulation parameters for multiresponse spans the multi-dimensional space Φ. One point in this space we call ϕ.

3.2.1 Restrictions on values of simulation parameter

There are some restrictions on the simulation parameters, since there are several covariance matrices that all have to be positive definite. The covariance matrix of the explanatory variables($\boldsymbol{\Lambda}$) is always positive definite. And $\boldsymbol{\Sigma}_{yy}$

is positive definite when $-1 < \rho < 1$. The second term in the conditional variance($\mathbf{\Sigma}_{zy}^t \mathbf{\Lambda}^{-1} \mathbf{\Sigma}_{zy}$) and the conditional variance itself($\mathbf{\Sigma}_{\boldsymbol{y}|\boldsymbol{x}}$) has to be positive definite. The conditional variance is positive definite when $-1 < \varrho < 1$.

To determine if a covariance matrix is positive definite it is better to check the correlation matrix. For a 2×2 correlation matrix, the off diagonal element have to be in the interval $< -1, 1 >$, for the correlation matrix to be positive definite.

The matrix $\mathbf{\Sigma}_{zy}^t \mathbf{\Lambda}^{-1} \mathbf{\Sigma}_{zy}$ can be expressed as

$$\begin{bmatrix} R_1^2 & R_{12} \\ R_{12} & R_2^2 \end{bmatrix}$$

Hence

$$-1 < \frac{R_{12}}{\sqrt{R_1^2 R_2^2}} < 1 \tag{3.9}$$

for $\mathbf{\Sigma}_{zy}^t \mathbf{\Lambda}^{-1} \mathbf{\Sigma}_{zy}$ to be positive definite. Using eq. 3.7 we have that R_{12} can be written as

$$R_{12} = \rho - \varrho\sqrt{(1 - R_1^2)(1 - R_2^2)}$$

Then eq. 3.9 can be written as

$$-1 < \frac{\rho - \varrho\sqrt{(1 - R_1^2)(1 - R_2^2)}}{\sqrt{R_1^2 R_2^2}} < 1$$

Therefore there are some combinations of ρ, ϱ, R_1^2 and R_2^2 that are impossible.

In addition there are some choices of relevant positions that put restrictions on the choices of the correlations. Example, if only the first component is relevant for both responses and $R_1^2 = R_2^2 = R^2$. Then $R^2 = \sigma_{z_1 y_1}^2 = \sigma_{z_1 y_2}^2$(the first eigenvalue is 1) and $|R_{12}| = R^2$. The conditional correlation is then

$$\varrho = \frac{\rho \mp R^2}{1 - R^2}$$

Then the conditional correlation is decided by R^2 and ρ. There might be other constraints on the correlations for other choices of *relpos* as well.

Chapter 4

Results

In some Figures and tables the simulation parameters are written with latin letters. Table 4.1 gives a translation

Table 4.1: *A translation of parameters from greek letter to latin letters*

Parameter	n	p	ρ	ϱ	γ	R_1^2	R_2^2	m	R^2
Latin	n	p	rho_u	rho_b	gamma	R2_y1	R2_y2	m	R2

4.1 Estimation Uniresponse

4.1.1 Estimation with Least Squares

To find out how different simulation-parameter-values affect the estimation of $\boldsymbol{\beta}$ with Least Squares method, we kept all simulation parameters constant, except the one that was varied. The estimation error was calculated as described in section 2.3 with eq. 2.11, using 100 replications(r). The constant values for the simulation parameters were set to the values in table 4.2.

Table 4.2: *Simulation parameter values*

$n = 20$
$p = 10$
$m = 2$
$relpos = (1, 2, ..., m)$
$\gamma = 0.7$
$R^2 = 0.8$

The values are based on a distribution that we often find. The first few components are relevant and the number of observations are not to high and there are some collinearity between the explanatory variables.

The simulation parameters $n, relpos, m, R^2, p$ and γ was investigated. The Least Squares method was fitted with all explanatory variables, no kind of variable selection was done. The Estimation Error was plotted against the selected simulation parameter as shown in Figure 4.1

Form eq. 2.23, we can see that the effect of different n is in the estimation of the eigenvalues. If n is low the eigenvalues of $\boldsymbol{X^t X}^{-1}$ are low. When n increases the eigenvalues increases and the estimation error decreases. That is what we see in Figure 4.1 **a)**.

The simulation parameter $relpos$ will affect the denominator in eq. 2.23. If the relevant components are in the first positions, the estimation error will be larger because the relevant components corresponds to the highest eigenvalues. If the relevant components corresponds to the smallest eigenvalues the estimation error will be lower. This is what we see in Figure 4.1 **b)**.

Different values of m will change the denominator in 2.23. A higher m will increase the number of $\sigma_{z_i y} \neq 0$ which will increase the value of the denominator as m increases. The estimation error will decrease as m

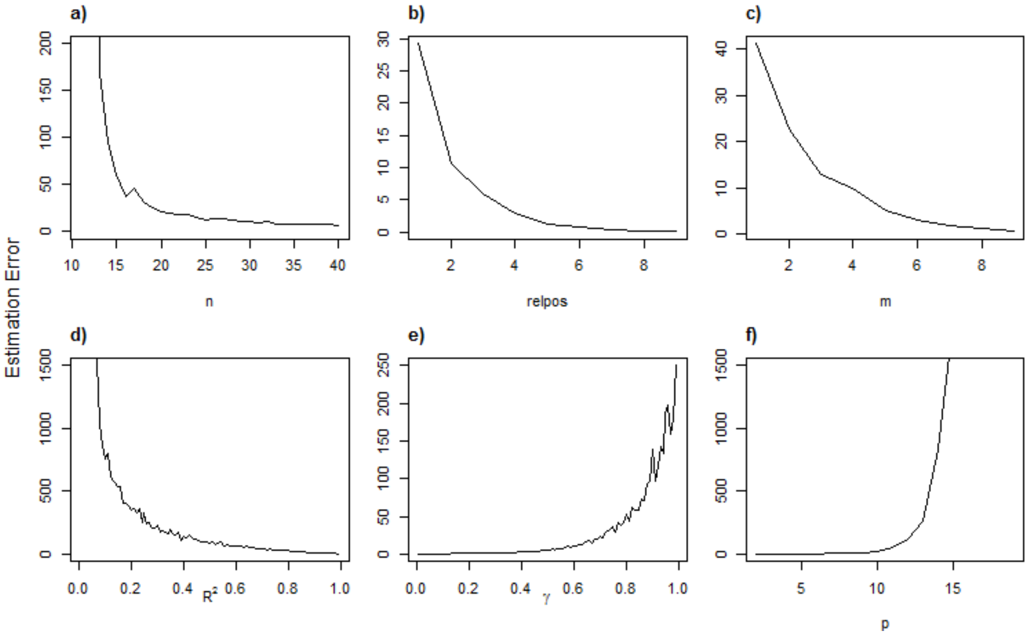

Figure 4.1: *The Estimation Error for β when fitting with Least Squares method with one simulation parameter varying at a time.* ***a)*** *The Estimation Error when n is increasing.* ***b)*** *The Estimation Error for the increasing relpos. If relpos is 2, the relevant components is 2 and 3.* ***c)*** *Estimation Error when the number of relevant components(m) is increasing* ***d)*** *Estimation Error for increasing R^2* ***e)*** *Estimation Error for increasing collinearity (γ).* ***f)*** *Estimation Error for increasing p.*

increases. This is what we see in Figure 4.1 **c)**

When the correlation($R^2 = Corr(\boldsymbol{\beta}^t \boldsymbol{x}, Y)$) between the explanatory variables and the response increases the estimation error decreases. This is seen in Figure 4.1 **d)**. Looking at eq. 2.23 we can see that it is due to the parameter $\sigma^2 = 1 - R^2$ and the denominator. As R^2 increases the denominator will increase. The difference between the denominator and R^2 is that in the denominator we square the eigenvalues. Both the scaling(denominator) and σ^2 will result in a decrease of the estimation error when increasing R^2.

Collinearity is a well known problem when using Least Squares method. And the higher the collinearity the worse LS performs. Looking at Figure 4.1 **e)** it can be observed how the level of collinearity increases the Estimation Error increases.

The simulation parameter p affects the estimation error greatly. When p is larger than 10 the estimation error increases rapidly as seen in Figure 4.1. In eq. 2.23 we see that p will change the sum of the estimated eigenvalues. By increasing p, the number of variables, the number of eigenvalues to estimate increase. The eigenvalues that are added by increasing p gets lower as p increases, which results in a larger estimation error.

Investigating the y-axis in Figure 4.1 we observe that some simulation parameters result in much larger estimation error than others. The parameters that resulted in the highest estimation errors are the parameters R^2, γ, p and n. The estimation of the eigenvalues affect the estimation error heavily and the simulation parameters that affect the estimation of the eigenvalues are $gamma$, n and p.

4.1.2 Estimation with PLS1

We performed a similar simulation study as in section 4.1.1, but using PLS1 to estimate $\boldsymbol{\beta}$ instead of Least Squares method. The same values on the simulation parameters were used (see Table 4.2). The number of components used for each replication was chosen to be the number of components that gave the smallest estimation error. The results can be seen in Figure 4.2.

For the PLS1 estimator we can not find an expression for the estimation error as a function of data as for the Least Squares method, since $E(\hat{\boldsymbol{\beta}})$ and $Var(\hat{\boldsymbol{\beta}})$ are unknown. We can estimate the estimation error as in eq. 2.11. Some of the simulation parameters act the same way as when using

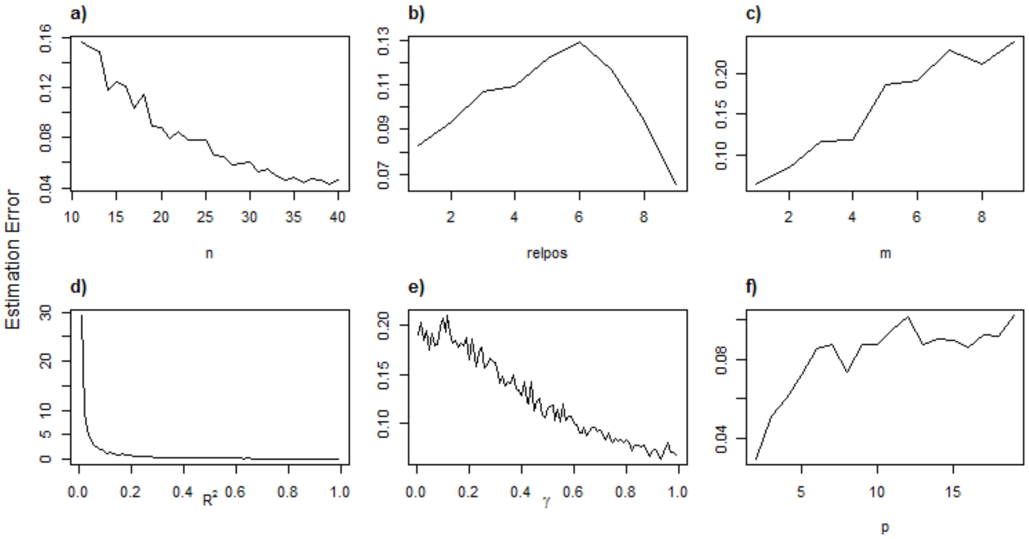

Figure 4.2: *The Estimation Error for β, when using PLS1 as estimator with one simulation parameter varying at a time.* ***a)*** *The Estimation Error when n is increasing.* ***b)*** *The Estimation Error for the increasing relpos. If relpos is 2, the relevant components is 2 and 3.* ***c)*** *Estimation Error when the number of relevant components(m) is increasing* ***d)*** *Estimation Error for increasing R^2* ***e)*** *Estimation Error for increasing collinearity (γ).* ***f)*** *Estimation Error for increasing p*

Least Squares method. The main difference is that all estimation errors are much smaller than for Least Squares method by investigating the y-axis. In Figure 4.2 a) we can see that as n increases the estimation error decreases. It is natural to think that we get better estimation when we have more observations.

In Figure 4.2 b) something strange happens. As *relpos* increases the estimation error increases at first, but at $relpos = 6, 7$ the estimation error decreases. This might be a scaling effect, due to that we divide on $\boldsymbol{\beta}^t\boldsymbol{\beta}$ and as in Least Squares as the *relpos* increases the estimation error decreases.

The effect we see for the first *relpos*'s might be explained by that we have high γ and two relevant components that are the first components. This is a situation where PLS1 is known to estimate well, but as the relevant positions is on the components that are not one of the first components, the estimation error increases. There might be some interaction effects with γ

Looking at Figure 4.2 c) we can see that the opposite happens compared to the Least Squares method. As m increases the estimation error increases. This shows that if we have many relevant components it is hard to estimate. It also might be explained by the values decided on the simulation parameters. We have somewhat high collinearity(high γ) and if we have many relevant components we have components with small eigenvalues which are relevant. For a lower γ the situation might be different. There might be some interaction effect between m and γ or some of the other simulation parameters. Another possibility is that if there are more relevant components the method will require more components included to reach minimum estimation error. This requires more components to be estimated and a higher estimation error.

In Figure 4.2 d) we see that the estimation error decreases as the correlation(R^2) increases. As the correlation between the response and explanatory variables increases we get better estimation.

Looking at Figure 4.2 e) we can see that the opposite happens compared to the Least Squares method. The estimation error decreases as γ increases. High collinearity is better than low collinearity. This might be explained by that $m = 2$ and $relpos = 1, 2$. The relevant components are the two first components and the PLS1 estimator then get better estimation error as there is high degree of collinearity and have many small irrelevant eigenvalues and only a few large relevant eigenvalues. The opposite might happen when the relevant positions is not the two first components, but some of the last components.

We suspect there might be some interaction effects between *relpos*, m and γ. We do not investigate the interactions between simulation parameters for estimation error. As we have seen, some of the effects of the simulation parameters might be explained by that we divide by $\boldsymbol{\beta}^t\boldsymbol{\beta}$. Therefore it might not be such a good idea after all. The prediction error and estimation error are quite similar so many of the effects of the simulation parameters on estimation error might be similar when using prediction error instead.

4.2 Prediction Uniresponse

4.2.1 PLS1 with two-levels of Simulation-parameters

Since we suspect that there might be some interaction effects between the simulation parameter as described in sec. 4.1.2 for estimation error there might also be some for prediction error. We can do some full-scale simulation with different levels on the simulation parameters to potentially find important interactions and the effects of the simulation parameters has on the prediction error. In the first full-scale simulation the parameter values was decided to be as in Table 4.3. One high and one low value on each simulation parameter. In total we are investigating $2^6 = 64$ ω's(see end of section 3.1). For each point in the parameter space 100 replicates were done, and for each replicate a PLS1-model was fitted with 1-14 components. The limit of 14 components was to make the programming a bit easier and faster and that all points in the parameter space would have the same number of total components. So for each replicate, we have 14 $\hat{\theta}^2$ calculated from eq. 2.16, one for each number of components. In total

14(components)×100(replicates)×64(ω's) $= 89600$ $\hat{\theta}^2$'s was estimated.

Table 4.3: *Parameter values used in the first round of simulations*

Parameter	Low value	High value
n	20	100
p	15	50
m	2	10
γ	0.3	0.95
relpos	(1, 2,...,m)	(5, 6,...,m)
R^2	0.5	0.95

4.2.2 Points in parameter space with large prediction error

We have plotted the prediction error as a function of components, with two curves, one for each choice of n. After investigating the plots it was discovered that for some ω's the prediction error ($\bar{\hat{\theta}}^2$) was much greater than 1 for any choice of number of components included (see Figure 4.3). That means that the null model is a better predictor at these points in the parameterspace.

The estimated standard deviation for the estimated prediction error($\hat{\theta}^2$) in each ω was calculated for each number of PLS1-components with eq 2.17. The largest standard error was as large as almost 12000. This indicates that for some combination of number of components and ω the estimation of $\boldsymbol{\beta}$ varies greatly. When only the number of component giving lowest $\bar{\hat{\theta}}^2$ for each ω is chosen, the maximum standard error is reduced to about 80. This is still very large, when we consider that the prediction errors(θ^2) for useful models should be between $1 - R^2$ and 1(discussed in section 2.4.1). Therefore the null-model is a better predictor for these points in the parameter space.

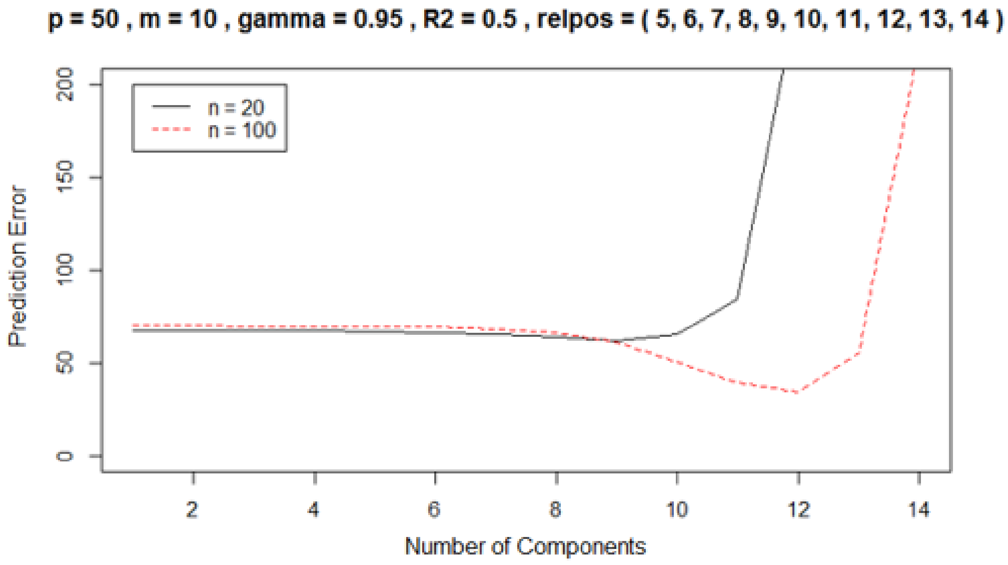

Figure 4.3: *The average prediction error against the number of components for two different ω's that gave high prediction error(higher than for the null model) for all choices of components.*

The 14 average $\hat{\theta}^2$ was computed for each ω(one for each number of components included). The $\omega's$ with no $\bar{\hat{\theta}}^2$ below 1 was removed. The removed ω's is presented in Table 4.4. The standard deviation was estimated for the number of component included with lowest $\bar{\hat{\theta}}^2$ for the ω's left and the maximum standard error was now below 1.

When investigating the removed $\omega's$ in Table 4.4 the most common combination among them was $m = 10$ and $\gamma = 0.95$. In other words many relevant components and high collinearity causes a problem. In this situation we have a few large relevant eigenvalues and many small relevant eigenvalues. It might be the number of small relevant eigenvalues that causes the prediction error to be large. Even though we have a few large relevant eigenvalues, it does not seem to compensate for the many small eigenvalues. The question

Table 4.4: *The 19 ω's removed because of to high $\bar{\hat{\theta}}^2$ for any choice of number of components included. The value of comp is the number of PLS1-components that gave the lowest $\hat{\theta}^2$. If relpos is 1 it means that the first m components are relevant. If relpos is 5 it means that the first relevant component is the 5th component.*

$\bar{\hat{\theta}}^2$	comp	n	p	m	γ	*relpos*	R^2
1.07	4	20	15	2	0.95	5	0.50
1.09	2	20	15	10	0.30	1	0.50
1.13	2	20	50	10	0.30	1	0.50
1.21	3	20	50	10	0.30	5	0.50
1.47	3	20	15	10	0.30	5	0.50
8.42	10	20	50	10	0.95	1	0.95
9.61	10	20	15	10	0.95	1	0.95
9.76	10	100	15	10	0.95	1	0.50
10.49	10	100	50	10	0.95	1	0.50
16.11	14	100	50	10	0.95	5	0.95
29.45	13	100	15	10	0.95	5	0.95
34.66	12	100	50	10	0.95	5	0.50
42.07	7	20	50	10	0.95	1	0.50
46.13	8	20	15	10	0.95	1	0.50
51.48	11	100	15	10	0.95	5	0.50
59.92	9	20	15	10	0.95	5	0.50
62.11	9	20	50	10	0.95	5	0.50
73.51	12	20	50	10	0.95	5	0.95
101.74	11	20	15	10	0.95	5	0.95

is if there are any interactions between γ and m that causes the prediction error to be large or if it is some other effect that causes these ω's to have large prediction errors.

4.2.3 Analysis of effects of simulation parameter values on prediction error

An analysis of variance of the main effects was done with prediction error as response and the simulation parameters as factors with two levels of each. This was first done without removing the ω's in Table 4.4. For each ω we choose the number of components having lowest average prediction error. An analysis of variance was run and the estimated effects can be seen in Table 4.5. The parameterization of the effects was done as described in section 2.10, with the model

$$\hat{\theta}^2_{ijklstu} = \mu + n_i + p_j + \gamma_k + m_l + R^2_s + relpos_t + \epsilon_{ijklstu}$$

and with the restriction that sum of all effects of a factor to be zero for all six factors.

Looking at the estimated effects we see that all of them are quite high. All are above 1. That means that by changing the level on any of the factors keeping the others the same the estimated prediction error will be above 1 and the Null Model would be a better suggestion. These strange estimated effects may be an indication of some interaction effect. The simulation parameters that has the most effect on the prediction error is m and γ. This is what we expected since high value on m and γ gave prediction errors that was so large that they were later removed from the dataset.

Investigating the two-factor interaction effects, with all ω's included, we can see that the most significant interaction is between m and γ. This is

Table 4.5: *The estimated main effects of the simulation parameters and their p-values with all ω's included.*

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	9.0338	0.3130	28.87	0.0000
relpos(1)	-4.7478	0.3130	-15.17	0.0000
R2(0.5)	1.4122	0.3130	4.51	0.0000
p(15)	0.9675	0.3130	3.09	0.0020
n(20)	3.9862	0.3130	12.74	0.0000
gamma(0.3)	-8.5724	0.3130	-27.39	0.0000
m(2)	-8.6765	0.3130	-27.72	0.0000

illustrated in Figure 4.4. The two lines are not parallel, and therefore there is an interaction between m and γ. High degree of collinearity(large γ) is only a problem when we have many relevant components(large m). This is what we saw in sec. 4.2.2.

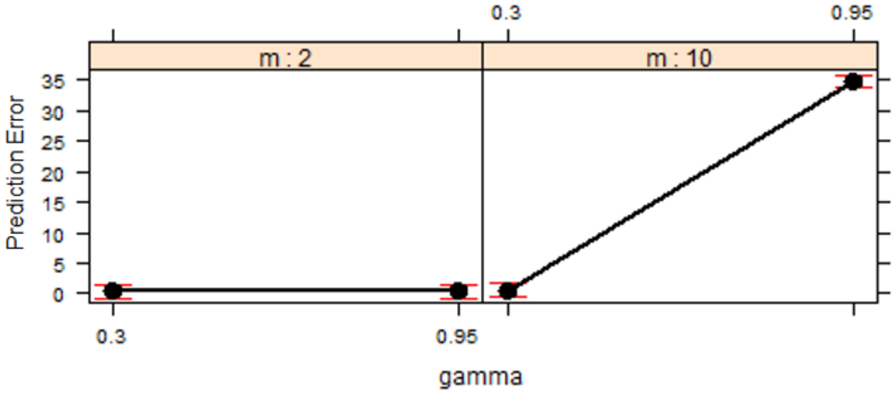

Figure 4.4: *Effect plot of the interaction between m and γ.*

We removed the ω's as mentioned in 4.2.2(because of high prediction errors and high variation in prediction error) and did some analysis on the data left. We can see the estimated main effects in Table 4.6. All the estimated effects of the simulation parameters are below 1 and higher than -1. The most significant simulation parameter is R^2 as expected. If R^2 is large, σ^2 is small and the prediction error is small. This corresponds with a positive effect of R^2 at level 0.5 in Table 4.6. Looking at some of the other effects we can notice that the effect of *relpos* is negative. Meaning that if the relevant components corresponds to the first and largest relevant eigenvalues, the prediction error is smaller compared to the relevant components starting at position 5. The effect of p is negative, meaning that by increasing the number of explanatory variables gives better prediction. This is a bit weird, because it means that by adding variables that are uncorrelated to the response we get better prediction. The effect of low n is positive, meaning that by increasing the number of observations the prediction error decreases. The effect of γ is positive, meaning that if the eigenvalues decreases faster the prediction error increases. But we know that if the first components are relevant and the number of relevant components are few the PLS1 predictor predicts well when there is high degree of collinearity(γ is large). Therefore we suspect that

Table 4.6: *The estimated main effects of the simulation parameters and their p-values after removing ω's with $\hat{\theta}^2 > 1$.*

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.4879	0.0049	98.58	0.0000
relpos(1)	-0.0304	0.0040	-7.50	0.0000
R2(0.5)	0.2389	0.0042	57.08	0.0000
p(15)	0.0148	0.0040	3.68	0.0002
n(20)	0.0511	0.0042	12.20	0.0000
gamma(0.3)	-0.0750	0.0044	-16.90	0.0000
m(2)	-0.1404	0.0048	-29.15	0.0000

there still are some interaction effects between the simulation parameters.

One can argue that we should not remove ω's from our study because of high prediction error. By removing ω's we don't have a 2^6 factorial design, making it difficult to look at some interactions. Therefore it was decided to do a second simulation, changing the high level of m from 10 to 4. It turned out that 4 ω's had $\bar{\hat{\theta}}^2 > 1$ for the number of components with average lowest prediction error. The highest at 1.78. Estimating the standard deviation as in eq. 2.17, we find that the largest is 0.916 and the smallest is 0.001. It is still a big difference, but it is not as bad as for the first simulation. Therefore we decide to not remove any of the ω's even if not all $\bar{\hat{\theta}}^2$ is below 1.

Performing the main effect ANOVA as for the first simulation gave similar effects as for the analysis with removed ω's. We decided to go further to investigate any interaction effects. Running an analysis of variance on two-factor interactions of all the simulation parameters gave the estimated effects in Table 4.7. Still the effect of R^2 is the most significant. The intercept is the average prediction error for the ω's included in the study. Investigating the different interactions we see that the interaction between *relpos* and γ is the most significant. In Figure 4.5 we can see the interaction between these two simulation parameters.

It's the combination of high collinearity and the first relevant component at position 5 that causes the prediction error to increase. That happens if we have a small eigenvalue with a corresponding relevant component.

Table 4.7: *The estimated effects of the simulation parameters included effects of two-factor interaction effects on the prediction error.*

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.4305	0.0033	128.97	0.0000
relpos(1)	-0.0975	0.0033	-29.19	0.0000
R2(0.5)	0.2767	0.0033	82.89	0.0000
p(15)	0.0442	0.0033	13.25	0.0000
n(20)	0.0988	0.0033	29.58	0.0000
m(2)	-0.0744	0.0033	-22.28	0.0000
gamma(0.3)	-0.0818	0.0033	-24.50	0.0000
relpos(1):R2(0.5)	-0.0096	0.0033	-2.88	0.0040
relpos(1):p(15)	-0.0406	0.0033	-12.18	0.0000
relpos(1):n(20)	-0.0587	0.0033	-17.59	0.0000
relpos(1):m(2)	0.0460	0.0033	13.78	0.0000
R2(0.5):m(2)	-0.0151	0.0033	-4.52	0.0000
p(15):m(2)	-0.0186	0.0033	-5.56	0.0000
n(20):m(2)	-0.0426	0.0033	-12.75	0.0000
m(2):gamma(0.3)	0.0590	0.0033	17.68	0.0000
relpos(1):gamma(0.3)	0.0722	0.0033	21.62	0.0000
R2(0.5):p(15)	-0.0028	0.0033	-0.83	0.4061
R2(0.5):n(20)	0.0233	0.0033	6.96	0.0000
R2(0.5):gamma(0.3)	-0.0141	0.0033	-4.21	0.0000
p(15):n(20)	0.0234	0.0033	7.00	0.0000
p(15):gamma(0.3)	-0.0281	0.0033	-8.42	0.0000
n(20):gamma(0.3)	-0.0488	0.0033	-14.63	0.0000

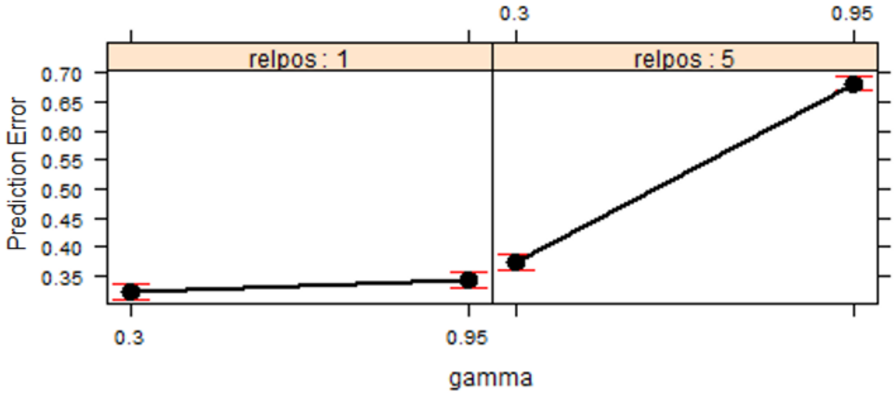

Figure 4.5: *Effect plot of the interaction between relpos and γ.*

Another interesting interaction is that between *relpos* and n. The effect plot shown in Figure 4.6 shows that the decreases in prediction error when increasing n is dependent on were the first relevant component is. Especially when *relpos* is 5, increasing n is very effective.

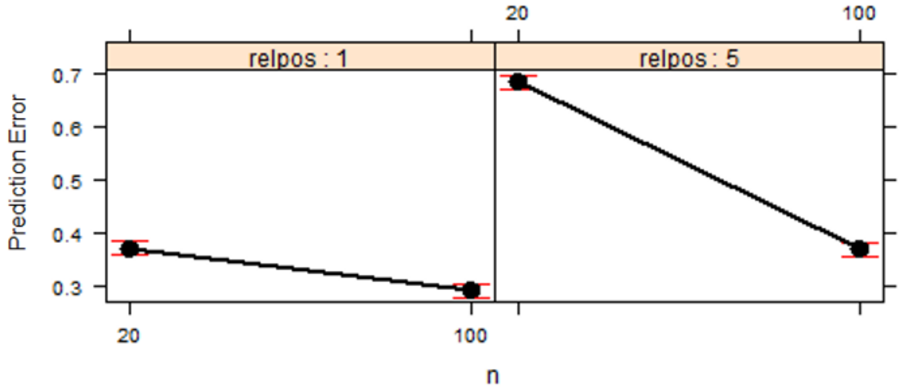

Figure 4.6: *Effect plot of the interaction between relpos and n.*

When estimating the 3-factor interaction effects(see table A.1 in Appendix), we saw that interaction was most significant between *relpos*, m and γ. In Figure 4.7 we have the 3-factor effect plot between these simulation parameters. We can see that it is the combination of high level of all 3 simulation parameters that causes the prediction error to increase. This is a point were we have many small eigenvalues with relevant components.

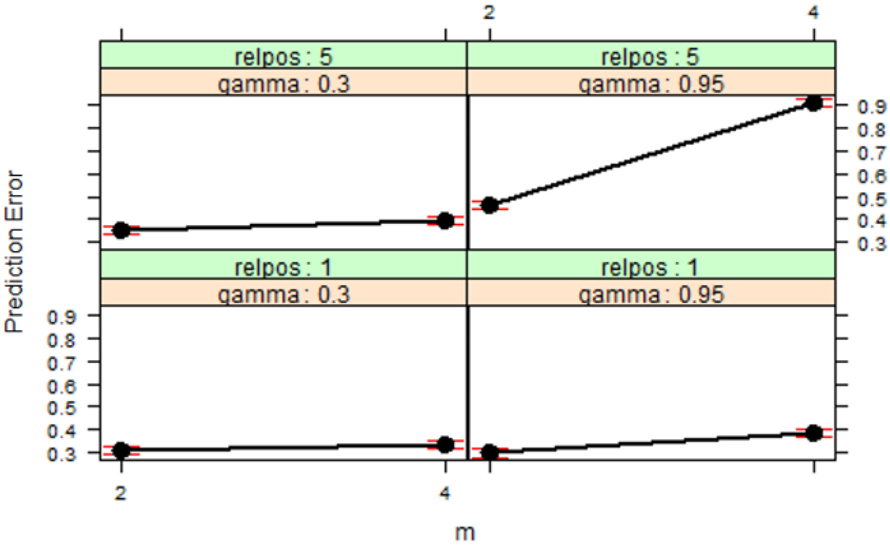

Figure 4.7: *Effect plot of the 3-factor interaction between relpos, m and gamma.*

Other 3-factor interactions that were highly significant were between *relpos*, n, γ, and n, m, γ. There are four simulation parameters that are repeated here. We estimated the 4-factor interaction effects between these parameters and made an effect plot shown in Figure 4.8. Investigating all the 4-factor interaction effects(see table A.1 in appendix) we see that the combination of these four simulation parameters is the most significant. From figure 4.8 we see that it is the combination of $m = 4$, $\gamma = 0.95$, $relpos = 5$ and $n = 20$ that causes the highest prediction error.In this situation it is very effective to increase n. We have several possible explanations for the large prediction error. Either the PLS1 predictor does not work well with the combination of small eigenvalues with relevant components, or that the number of components required to reach minimum prediction error is high.

It is also possible that the simulated data varies a lot, meaning that the estimated values varies much from the true values(decided with the simulation parameters). The last option we look at in section 4.2.4

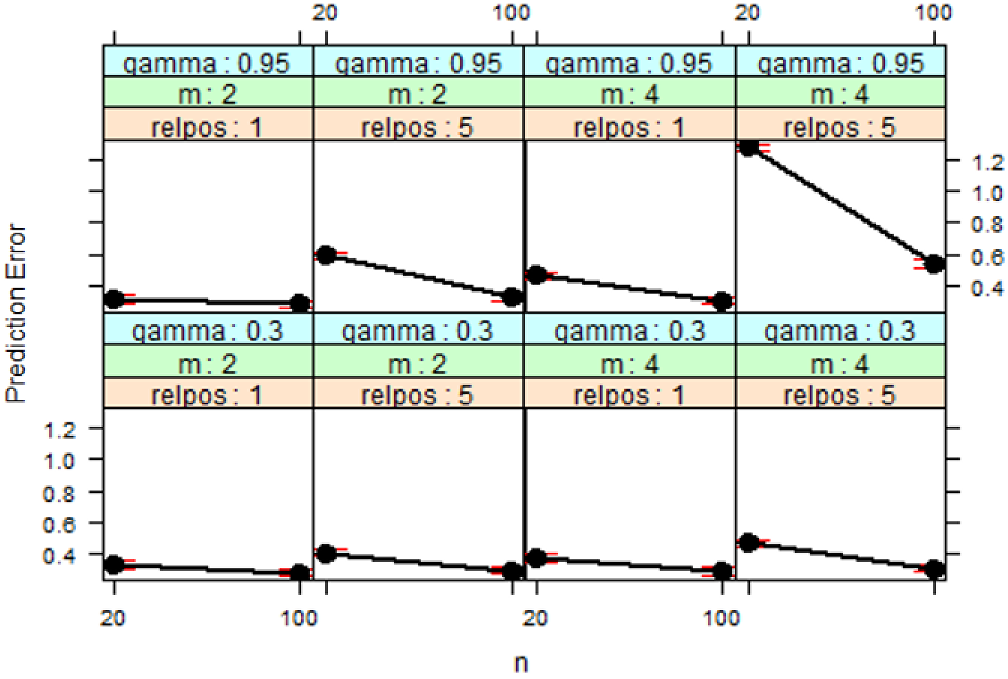

Figure 4.8: *Effect plot of the 4-factor interaction between relpos, γ, m and n.*

Another highly significant 4-factor interaction is that between *relpos*, γ, m and p. The interactions are shown in Figure 4.9. Here we observe that the effect of p is almost the same for all combinations of *relpos*, γ and m except for when *relpos* $= 5$, $\gamma = 0.95$ and $m = 4$. Then the prediction error is dependent on the level of p. It looks as if it pays to have more explanatory variables that are uncorrelated to the response when we have many small eigenvalues that corresponds to relevant components.

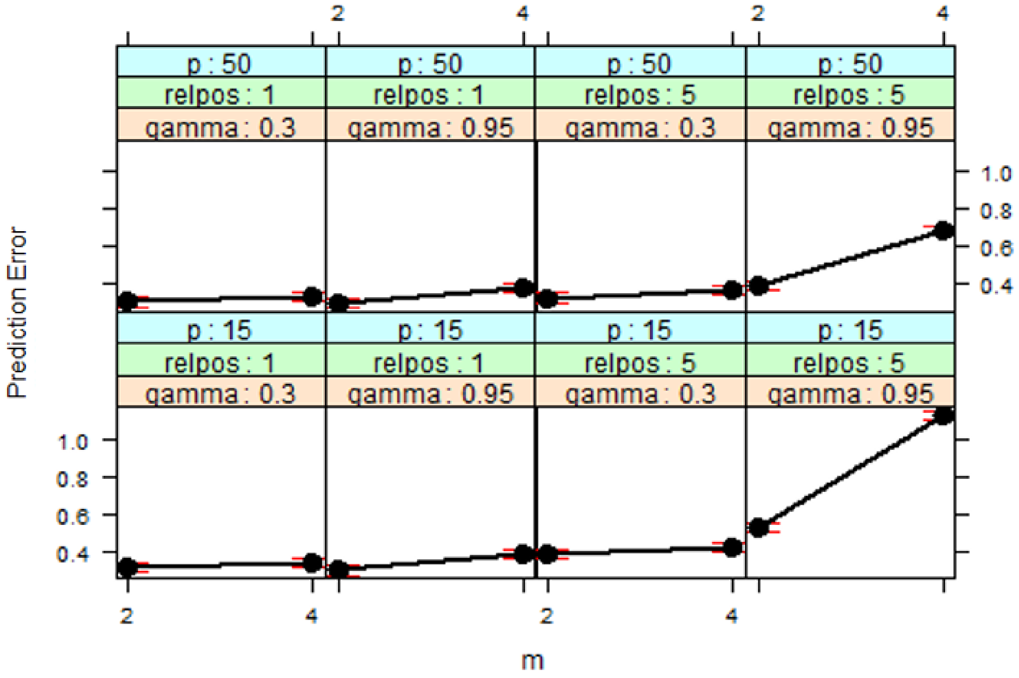

Figure 4.9: *Effect plot of the 4-factor interaction between relpos, γ, m and p.*

4.2.4 The estimated world vs the true world

With our simulation parameters we decide the true parameters and draw observations from it's distribution. We can estimate the parameters with the observations. The estimations might be similar to the true distribution or it might not be any similarities at all. From a dataset we can estimate the eigenvalues and find which principal components that seems to be relevant, but this might not give the same result or even closely to the same result as the true world. The practitioner does not know how many relevant components there are or which components that are relevant. Therefore it is necessary to estimate them. Some graphics might help. Figure 4.10 shows the eigenvalues and covariances between component and the response estimated from a dataset. This is compared to distribution of the true world. With true world we mean the true parameter values and with estimated world we mean the estimated parameter values. We see in Figure 4.10 that it is a large difference between the true world and the estimated world. In the true world (see 4.10 b)) the relevant components are 8 and 10. While from the estimated values in 4.10 it seems to be the 2-3 first components that are most relevant. Drawing new observations from the same distribution and estimating the world gave widely different result each time.

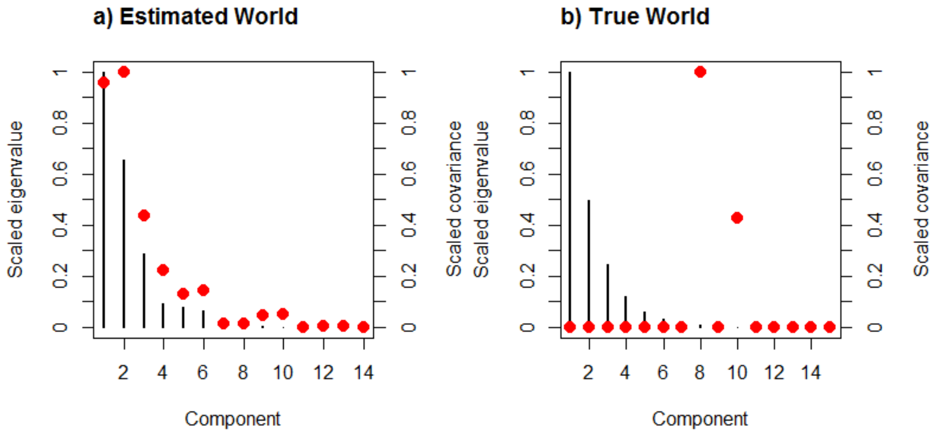

Figure 4.10: *The simulation parameter were set to* $n = 20, p = 15, m = 2, q = 2, relpos = (8, 10), \gamma = 0.7$ *and* $R^2 = 0.9$*. All variables were centred and the eigenvalues and covariance were scaled. The scaled eigenvalues are the black lines and the red dots are the scaled covariances plotted for each principal component. In* ***a)*** *we can see the estimated eigenvalues and covariances from the simulated data. In* ***b)*** *we see the true eigenvalues and true covariances based on the simulation parameters.*

Letting the simulation parameter n be 1000 we could see the similarity between the estimated world and the true world(see Figure 4.11). Still there is some noise, but we are able to see the correct relevant components.

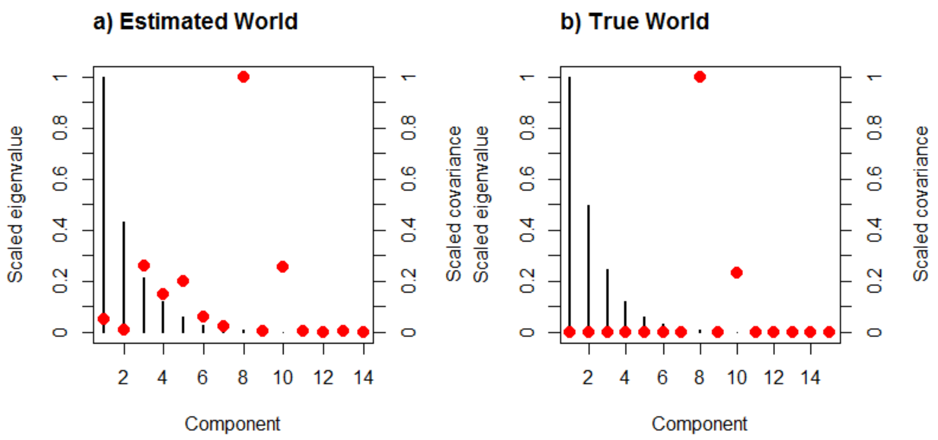

Figure 4.11: *The same situation as in Figure 4.10 except* $n = 1000$*.* ***a)*** *is the estimated world.* ***b)*** *is the True world.*

By looking at both Figure 4.10 and 4.11 we can see that the eigenvalues are estimated for the most part the same. The estimation of relevant positions are quite different for each time.

Small relevant eigenvalues is a situation were we earlier have discovered that it is difficult to predict. Investigating an easier situation as seen in Figure 4.12 we see that there are similarities between the true world and estimated world and we are able to find the correct relevant components. Another thing to observe here is that it seems as if there is a tendency to overestimate the large eigenvalues and underestimate the small eigenvalues.

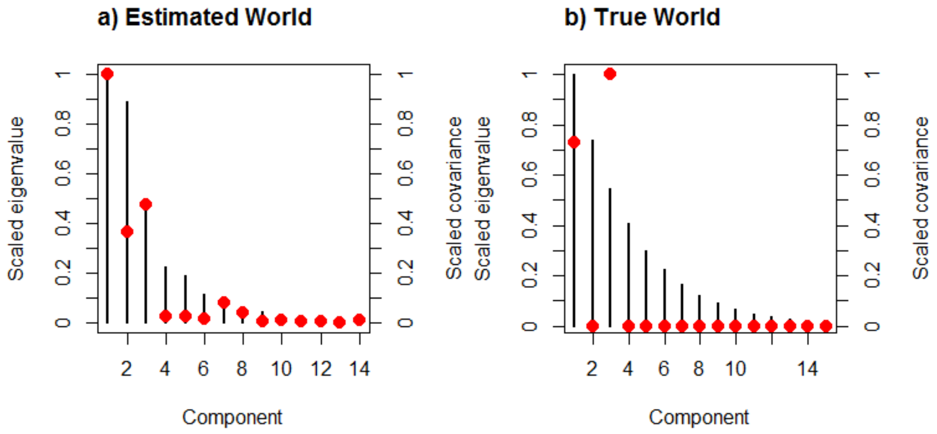

Figure 4.12: *The simulation parameter were set to* $n = 20, p = 15, m = 2, q = 2, relpos = (1, 3), \gamma = 0.3$ *and* $R^2 = 0.9$. *In* ***a)*** *we can see the estimated eigenvalues(black bars) and covariances(red dots) from the simulated data. In* ***b)*** *we see the true eigenvalues and true covariances based on the simulation parameters.*

4.3 Estimation of $\boldsymbol{\beta}$ by PLS2

We estimated the estimation error(eq 2.11) for one point in the parameterspace Φ. This point is relatively similar to the estimated world in the LMP dataset(see section 4.5). The diffrences are a larger p and R^2's are equal. With 100 replication the estimation error was estimated, shown in Figure 4.13. For one component PLS2 gave a better estimator than PLS1 for the first response, but the opposite happened for the second response. However the most striking part is how similar the two methods are. (Notice that the two PLS1 lines are not equal, even when the responses has the same parameter values. This is later discussed in chapter 5)

To investigate how the two methods differ when estimating $\boldsymbol{\beta}$, we did a few simulations without replications. A plot of the estimated $\boldsymbol{\beta}$'s for each

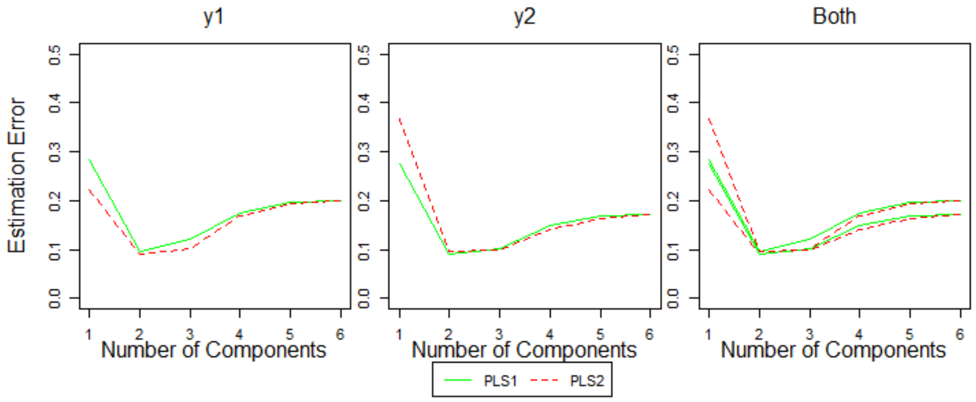

Figure 4.13: *The simulation parameter were;* $n = 40, p = 6, relpos = (3,1) and (3,1), \gamma = 0.5, \rho = 0.95, \varrho = 0.9, R_1^2 = 0.7$ *and* $R_2^2 = 0.7$. *The estimation error estimated with 100 replications.*

component are shown in Figure 4.14.

Again the estimators are extremely similar. For one component included, the third element of $\hat{\boldsymbol{\beta}}_{PLS1,Y_1}$, $\hat{\beta}_{PLS1,Y_1,3}$ comes closer to the true $\beta_{Y_1,3}$. For the second response the opposite happened, $\hat{\beta}_{PLS2,Y_2,3}$ came closer to $\beta_{Y_2,3}$. We can observe this at other elements of the estimated $\boldsymbol{\beta}$'s as well. In Figure 4.15 we have the prediction error for the same simulated dataset. For description of the prediction error for two responses see section 4.4. Comparing the prediction errors it almost looks as if they are the same plot but the PLS1 and PLS2 line has switched places.

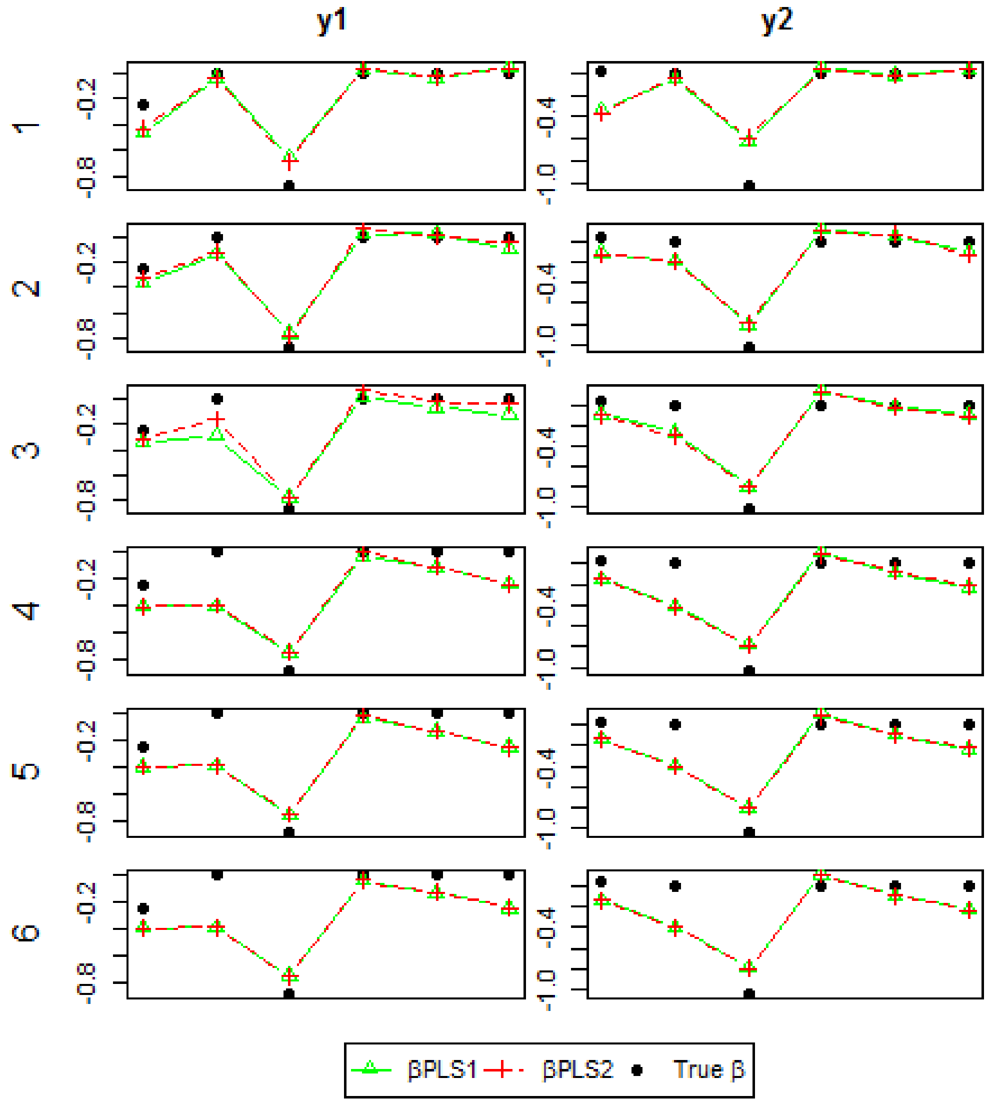

Figure 4.14: *The* $\boldsymbol{\beta}$*'s were estimated without replication. The simulation parameter were set to* $n = 40, p = 6, relpos = (3,1) and (3,1), \gamma = 0.5, \rho = 0.95, \varrho = 0.9, R_1^2 = 0.7$ *and* $R_2^2 = 0.7$. *The numbers 1,2,...,6 refers to the number of components included.*

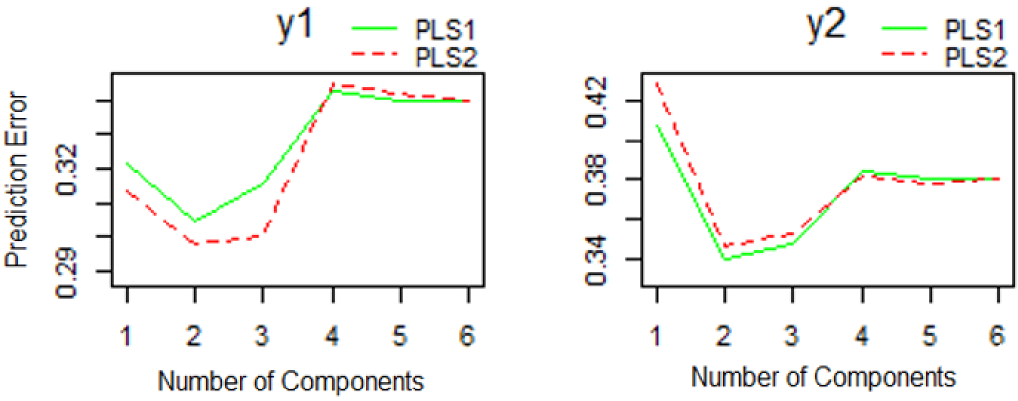

Figure 4.15: *The same $\hat{\boldsymbol{\beta}}$ as used in Figure 4.14. The prediction error was estimated with a testset of 1000 observations.*

To find if $\hat{\boldsymbol{\beta}}_{PLS2}$ is a linear combination or a weighted average of $\hat{\boldsymbol{\beta}}_{PLS1,Y_1}$ and $\hat{\boldsymbol{\beta}}_{PLS1,Y_2}$ we fit the model in eq. 2.25 for each $\hat{\boldsymbol{\beta}}_{PLS2,Y_i}$. We plot the coefficients $\hat{\alpha}_1$ and $\hat{\alpha}_2$ against the number of components included in Figure 4.16. The sixth component is the LS solution and therefore $\alpha_1 = 1$ and $\alpha_2 = 0$ for the first response and opposite for the second response. For the first component $\hat{\boldsymbol{\beta}}_{PLS2,Y_i}$ is exactly a linear combination of $\hat{\boldsymbol{\beta}}_{PLS1,Y_1}$ and $\hat{\boldsymbol{\beta}}_{PLS1,Y_2}$. Comparing the two lines there is a high symmetry along the line $\hat{\alpha} = 0.5$, indicating that it is a weighted average. The confidence intervals are quite wide for many of the components meaning $\hat{\boldsymbol{\beta}}_{PLS2,Y_i}$ is not an exact linear combination of $\hat{\boldsymbol{\beta}}_{PLS1,Y_1}$ and $\hat{\boldsymbol{\beta}}_{PLS1,Y_2}$.

We change the simulation parameters to a different situation, see Figure 4.17. And observe that with one component included $\hat{\boldsymbol{\beta}}_{PLS2,Y_i}$ is a linear combination of $\hat{\boldsymbol{\beta}}_{PLS1,Y_1}$ and $\hat{\boldsymbol{\beta}}_{PLS1,Y_2}$. In this situation the symmetry along the line $\hat{\alpha} = 0.5$ is not as clear as in the first situation. In addition the confidence intervals are relatively wide. They are narrower than for the first situation, but that is due to larger p.

We plotted the loadings for both simulations for the first component in

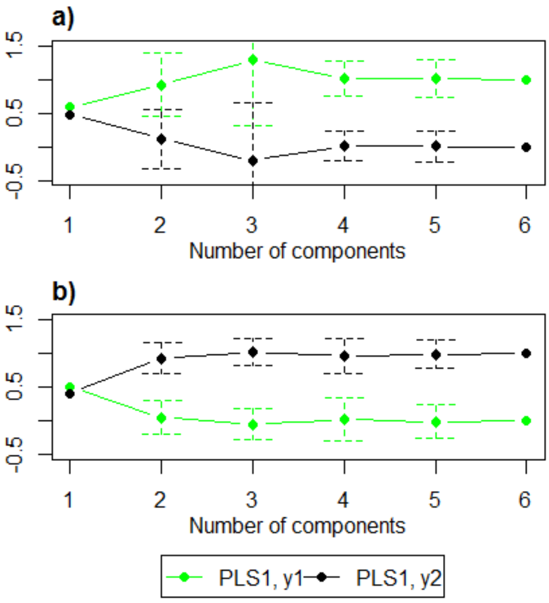

Figure 4.16: *The estimated α_1 and α_2 for the number of component included. With 95 % confidence intervals for all α's. a) is when $\hat{\beta}_{PLS2,Y_1}$ is used as response and b) is when $\hat{\beta}_{PLS2,Y_2}$ is used as response.*

Figure 4.18. We see that in both cases the loading estimated with PLS2 is in the middle of the loadings estimated with PLS1. It strengthens the indication that $\hat{\beta}_{PLS2,Y_i}$ is an exact linear combination or a weighted average of $\hat{\beta}_{PLS1,Y_1}$ and $\hat{\beta}_{PLS1,Y_2}$ for the first component.

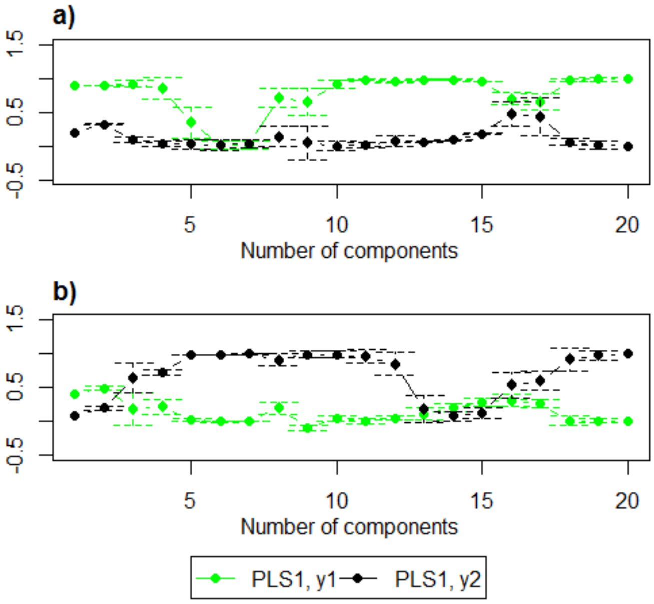

Figure 4.17: *The estimated α_1 and α_2 for the number of components included. With 95 % confidence intervals for α's. a) is when $\hat{\boldsymbol{\beta}}_{PLS2,Y_1}$ is used as response and b) is when $\hat{\boldsymbol{\beta}}_{PLS2,Y_2}$ is used as response. Simulation parameters were set to $n = 40, p = 20, relpos = (1, 2, 3)$ and $(3, 7, 8), \gamma = 0.9, \rho = 0.5, \varrho = 0.9, R_1^2 = 0.5$ and $R_2^2 = 0.9$*

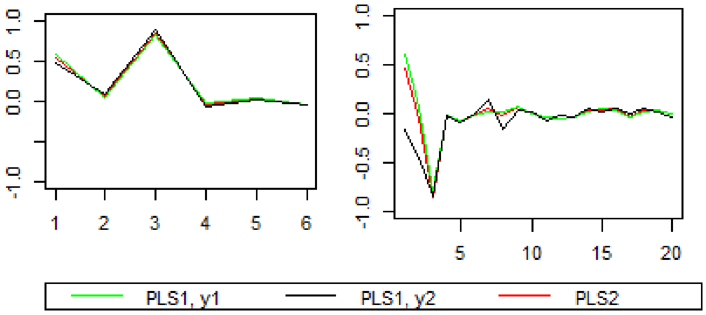

Figure 4.18: *First loading weights for both simulations.*

4.4 Prediction PLS2 compared to PLS1

The prediction error was estimated as discussed in section 2.4.2 with eq. 2.16. Where $\hat{\theta}^2_1$ is the prediction error for the first response and $\hat{\theta}^2_2$ is the prediction error for the second response. For each replication we fit a PLS1 and PLS2 model with 1 to k components included. For each component included we are estimating four different $\hat{\theta}^2$'s; $\hat{\theta}^2_{PLS1,Y_1}$, $\hat{\theta}^2_{PLS1,Y_2}$, $\hat{\theta}^2_{PLS2,Y_1}$ and $\hat{\theta}^2_{PLS2,Y_2}$. For each parameter point, ϕ, we can estimate $4 \times k$ prediction errors.

The $4 \times k$ $\hat{\theta}^2$'s was estimated with a 100 replications for a variety of different ϕ's. In Figures 4.19 and 4.20 the most typical results out of several different ϕ are shown. Figure 4.19 shows an example of a ϕ where PLS1 requires less components to reach the minimum prediction error for the second response. While for the first response 4 components is needed to reach the minimum prediction error for both methods. It was often seen that PLS2 needed more components than PLS1 to reach minimum prediction error.

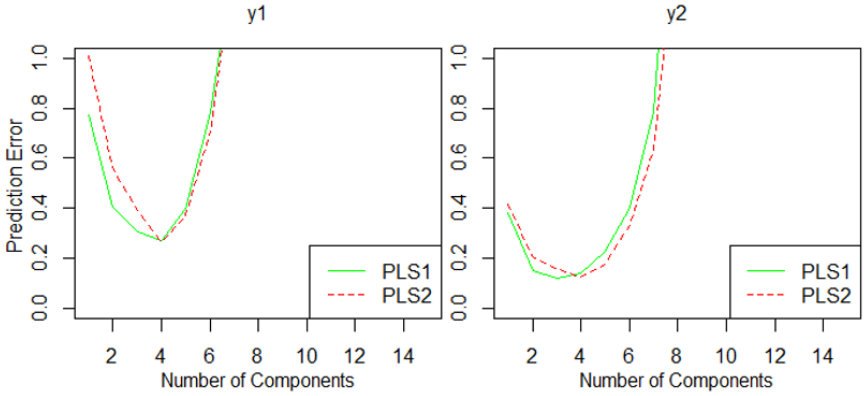

Figure 4.19: *The simulation parameter were* $n = 20, p = 15, relpos = (3,4)$ *and* $(1,3), \gamma = 0.9, \rho = 0.8, \varrho = 0.9, R^2_1 = 0.8$ *and* $R^2_2 = 0.9$. *The prediction error for each response was estimated with 100 replications. PLS2 requires more component than PLS1 to reach minimum prediction error.*

For other ϕ's the prediction error were approximately equal until minimum prediction error was reached for the two methods(see Figure 4.20). After minimum prediction error is reached, PLS2 have lower prediction error than PLS1.

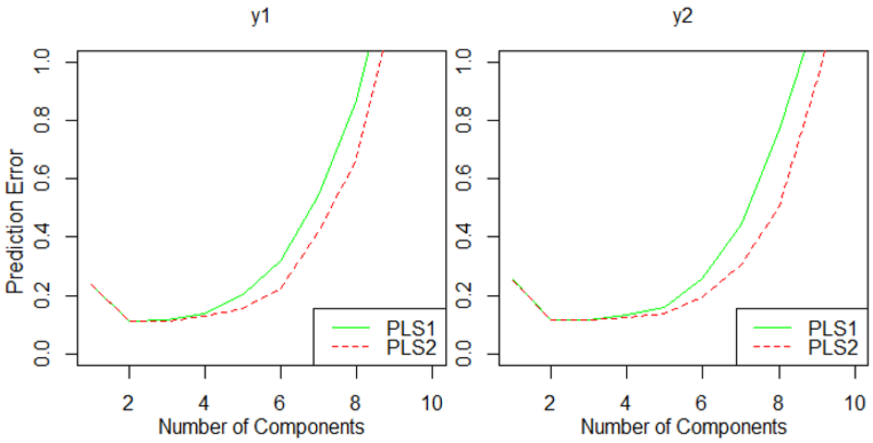

Figure 4.20: *The simulation parameter were* $n = 30, p = 10, relpos = (1, 2)$ *and* $(2, 3), \gamma = 0.9, \rho = 0.9, \varrho = 0.1, R_1^2 = 0.9$ *and* $R_2^2 = 0.9$. *The prediction error for each response was estimated with 100 replication. PLS1 and PLS2 has approximatly the same prediction error until minimum prediction error is reached.*

At one point in the simulation parameter space it was discovered that PLS2 predicts better than PLS1 for one of the responses(see Figure 4.21). Notice that both responses has only one relevant component(5) that usually makes poor prediction. PLS2 predicts better for the first response only. One of the differences between all ϕ's is the choice of *relpos*.

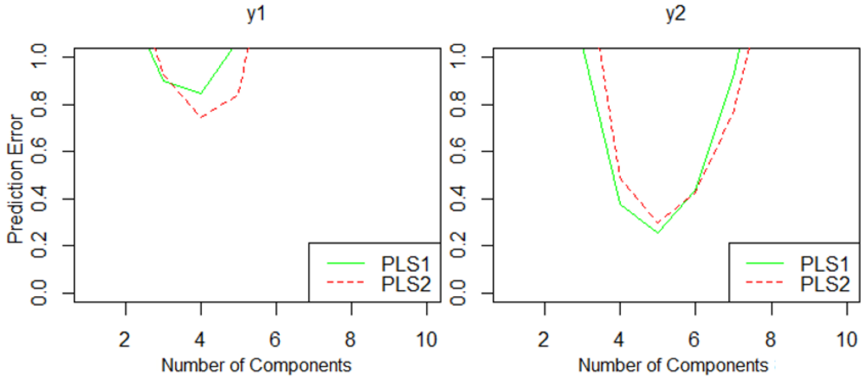

Figure 4.21: *The simulation parameter were* $n = 20, p = 10, relpos = (1, 2, 5)$ *and* $(2, 5), \gamma = 0.9, \rho = 0.5, \varrho = 0.9, R_1^2 = 0.6$ *and* $R_2^2 = 0.9$. *The prediction error for each response was estimated with 100 replication. PLS2 predicts better than PLS1 for the first response.*

4.4.1 Analysis of the difference in prediction error.

For the parameter settings chosen in Figure 4.21 we found that PLS2 predicts better than PLS1. We suspected that this might be due to the choice of *relpos*, having only one common ”difficult” relevant component(a component with small relevant eigenvalues). To investigate this further we did an analysis of variance as in section 4.2.3 with two levels on each simulation parameter. To simplify *relpos* was chosen to be, as in Figure 4.21, (1,2,5) and (2,5). The other simulation parameter values are shown in Table 4.8

Table 4.8: *The values of the simulation parameters.*

Parameter	n	p	ρ	ϱ	γ	R_1^2	R_2^2
Low value	15	15	0.5	0.6	0.3	0.5	0.6
High value	100	50	0.8	0.9	0.9	0.8	0.9

The number of replications was 100. Within each replication we fitted a PLS1 model and a PLS2 model for the same simulated dataset, hence we can use the difference in the prediction error between PLS1 and PLS2 as a response in the analysis of variance. We decided to do an analysis of variance for each response. The main effect ANOVA model is

$$(\hat{\theta}^2_{PLS1} - \hat{\theta}^2_{PLS2})_{ijlstuvw} = \mu + n_i + p_j + \rho_l + \varrho_s + \gamma_t + R^2_{1u} + R^2_{2v} + \epsilon_{ijlstuvw}$$

for each response. All factors had two levels. The parametrization is as described in section 2.10 with the sum of effects equal to zero. If the effect is positive the PLS2 predictor has lower prediction error than the PLS1 predictor.

We decided the number of components to be included to be the number of components giving the average lowest prediction error for each ϕ, meaning that $\hat{\theta}^2_{PLS1}$ and $\hat{\theta}^2_{PLS2}$ might have different number of components in the same ϕ. In front of the analysis we checked if there were any $\bar{\hat{\theta}}^2$'s(average prediction error), for the number of components with minimum prediction error, were above 1. There were some $\bar{\hat{\theta}}^2$'s above 1, but they were not too large. The advantages by keeping a 2^7 factorial design outweighed the fact that some ϕ's gave to large prediction error.

The main effects for the two responses are shown in Table 4.9 and Table 4.10. The intercept is the average difference in prediction error between PLS1 and PLS2 over all parameter values chosen. For the first response it is positive, meaning that on average PLS2 has lower prediction error than PLS1. For the second response the opposite happens and PLS1 predicts better than PLS2 on average.

Table 4.9: *The main effects of the first response, having relevant components 1,2 and 5.*

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.0089	0.0016	5.60	0.0000
gamma(0.3)	-0.0054	0.0016	-3.39	0.0007
n(100)	-0.0052	0.0016	-3.24	0.0012
p(15)	0.0019	0.0016	1.21	0.2254
R2_y1(0.5)	0.0058	0.0016	3.63	0.0003
R2_y2(0.6)	-0.0068	0.0016	-4.26	0.0000
rho_b(0.6)	0.0017	0.0016	1.09	0.2763
rho_u(0.5)	-0.0032	0.0016	-1.98	0.0475

Table 4.10: *The main effects of the second response having relevant components 2 and 5.*

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-0.0140	0.0014	-9.68	0.0000
gamma(0.3)	0.0034	0.0014	2.37	0.0178
n(100)	0.0143	0.0014	9.89	0.0000
p(15)	-0.0011	0.0014	-0.73	0.4633
R2_y1(0.5)	-0.0099	0.0014	-6.84	0.0000
R2_y2(0.6)	0.0011	0.0014	0.76	0.4482
rho_b(0.6)	0.0047	0.0014	3.26	0.0011
rho_u(0.5)	-0.0022	0.0014	-1.54	0.1245

From the result in Table 4.9, we see that it is advantage for PLS2 when we have low n, high γ, low R_1^2 and high R_2^2. The simulation parameters p, ϱ and ρ does not seem to effect the difference in prediction error(p-value above 0.01). There might however be some interactions where these parameters are included.

We did an analysis of all possible interactions and plotted the effects plot shown in Figure 4.22. It illustrates all interaction effects. In other words the average in difference in prediction error for all ϕ's chosen. We can determine the ϕ where the difference is largest between PLS1 and PLS2. This is when $n = 15, p = 15, \rho = 0.8, \varrho = 0.9, \gamma = 0.9, R_2^2 = 0.6, R_1^2 = 0.5$. We can see that when $n = 100$(the left half) there is almost no difference between PLS1 and PLS2. The same happens when $\gamma = 0.3$(bottom half). For observing large difference between PLS1 and PLS2 we need high collinearity and few observations simultaneously. From the upper right corner it is seen that there are some interaction effects between the rest of the simulation parameters.

When we had 'difficult' relevant components, it was discovered that the prediction error had high variation(see sec. 4.2.2). Since we have included a difficult relevant component, and the difference between PLS1 and PLS2 is largest when γ is high and n is small, the difference in prediction error might have high variation. Estimating the standard deviation($\hat{\sigma}$) for the residuals, we find it to be 0.18. This is large compared to the estimated effects.

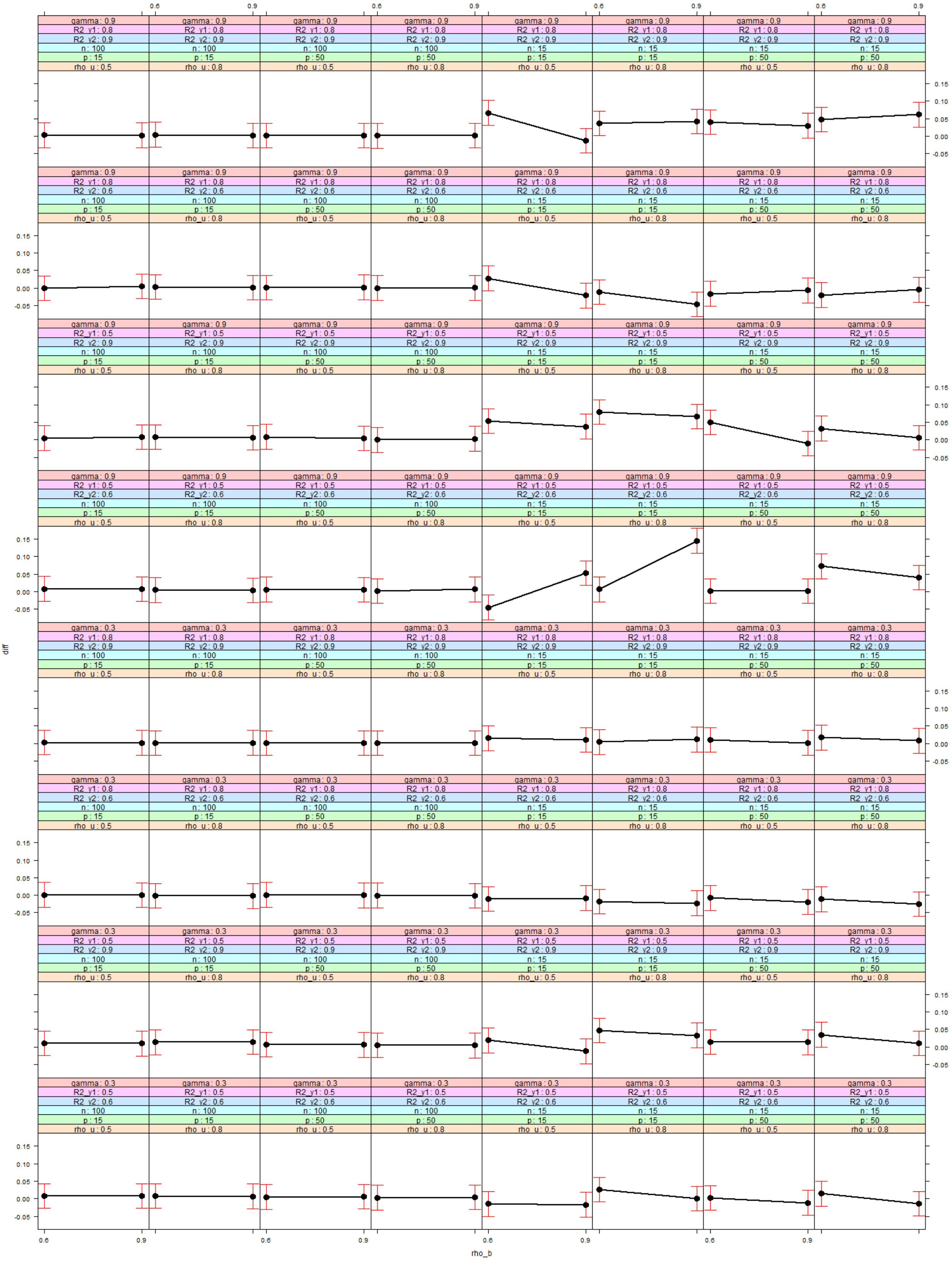

Figure 4.22: *Effect plot with all possible interactions.*

We used the ϕ where the difference in prediction error was found to be largest in Figure 4.22. And plotted the prediction error(estimated with 100 replications) against the number of components included for the first response. This was done twice and shown in Figure 4.23. This shows us that, even with an average of 100 replications, there is still high variation for the difference in prediction error at the chosen ϕ. Notice that for one of them there seem to be almost no difference between PLS1 and PLS2 and the prediction error is above 1. For the second simulation PLS2 performs better than PLS1 and the prediction error is below 1. This indicates that the variation in prediction error is large and therefore if we repeat the analysis above it might give a quite different result.

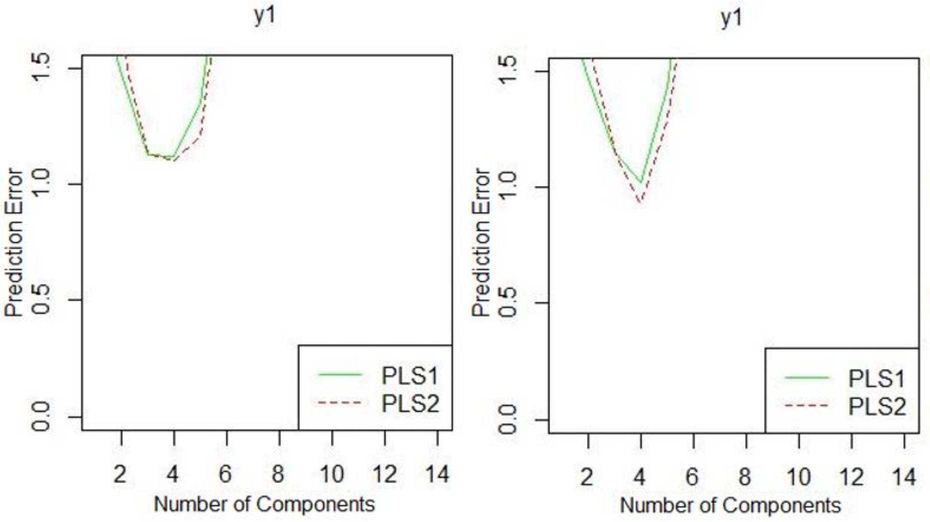

Figure 4.23: *Prediction error from two simulations with same ϕ with a 100 replication each*

To get a better understanding of when PLS2 performs better than PLS1 we did some simulation without replications and plotted the prediction error and the relevant components of both the true world and estimated world for both responses. We only have one replication, and therefore we decided to estimate the prediction error with a testset instead. We used a 100 new observations for estimating the prediction error. Figure 4.24 gives an example of when PLS2 wins and in Figure 4.25 is an example of where there is no difference between PLS1 and PLS2.

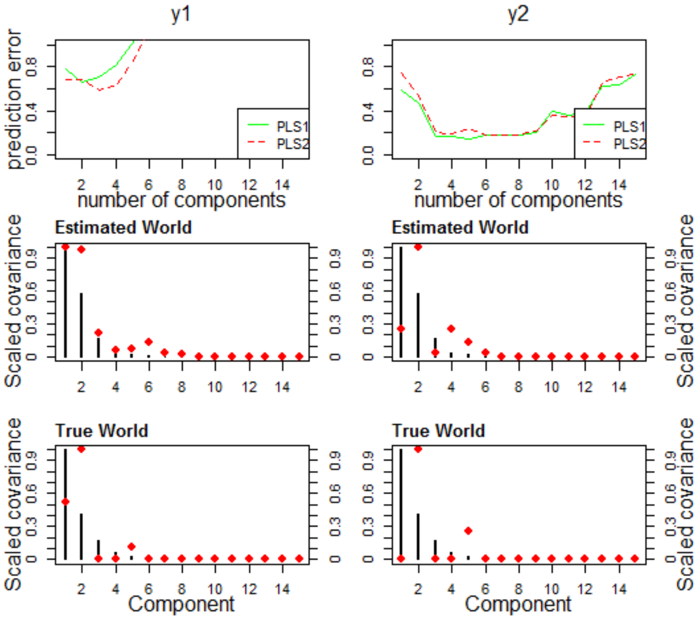

Figure 4.24: *The simulation parameters were set to* $n = 20, p = 15, \rho = 0.8, \varrho = 0.9, \gamma = 0.9, R_2^2 = 0.9, R_1^2 = 0.5$*. An example of when PLS2 has lower prediction error than PLS1*

It is difficult to determine exactly why PLS2 predicts better than PLS1 in Figure 4.24 than in Figure 4.25. What we do not control in the true distribution is the covariances. It is controlled indirectly by some of the correlations, but they are within some limitations decided at random. From Figure 4.24 and 4.25 we cannot see any pattern in the covariances when PLS2 predicts better than PLS1 compared to when there is no difference. Further investigation is needed.

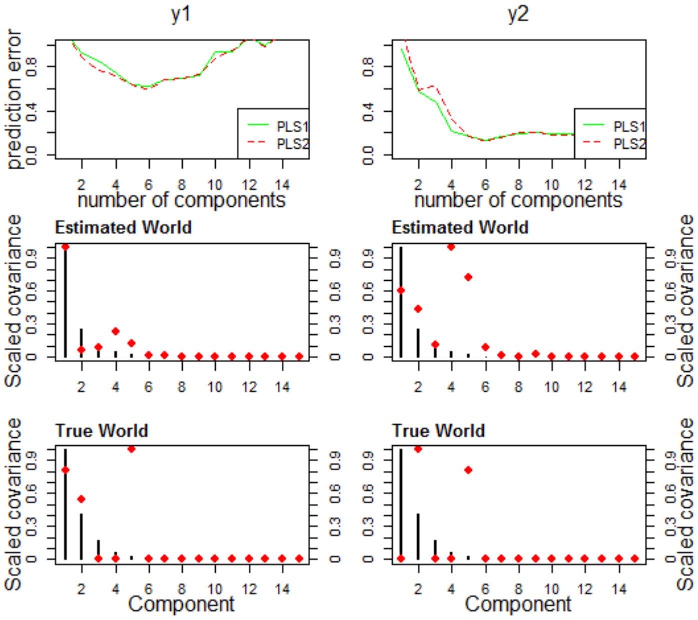

Figure 4.25: *The simulation parameters were set to* $n = 20, p = 15, \rho = 0.8, \varrho = 0.9, \gamma = 0.9, R_2^2 = 0.9, R_1^2 = 0.5$. *An example when there is no difference between PLS1 and PLS2.*

4.5 PLS2 and PLS1 on Real Data

4.5.1 Presentation of the datasets

Lean Meat Percentage(LMP)

The dataset for meat percentage in fattening pigs was made available by Animalia as part of the research project 'Determination of meat percent, and automatic multivariate classification of tissue in live pigs and pork(PigComp)'. In this dataset there are different measurement on pigs. The response are two different measurement of Lean Meat Percentage(LMP). One is the LMP measured by manual dissection(MD) and the other is based on CT-scan(CT). There are 86 observations. The explanatory variables measured are thickness of the subcuntaneous fat, thickness of the sirloin, thickness of the interior fat layer and the weight of the pig.

NIR on corn

This dataset consist of measurements on wheat by near infra red spectroscopy(NIR) with 700 different wavelengths [NIR, 2005]. In addition the responses; proteincontent, moisture, oilcontent and startch is measured for all 80 observations. This means we have 4 responses. We will choose some pairs of the responses to investigate.

4.5.2 Distribution and relevant components

Estimating different correlations and finding what seems to be the relevant components of the datasets will give us an indication of whether or not PLS2 will predict better than PLS1. Before any investigation we centred our data.

LMP data

In Figure 4.26 it seems that both responses has the same relevant components, 1 and 3. The eigenvalues do not indicate high degree of collinearity. The estimated correlation between the two responses is 0.967, which is a larger than what we have simulated with. To estimate the conditional correlation we fitted a LS to both responses and calculated the correlation between the residuals. It was found to be 0.917. From the LS-models we estimate that $R^2_{CT} = 0.632$ and $R^2_{MD} = 0.733$.

In some ways this is very similar to the situation in Figure 4.22 with the highest difference in prediction ability between PLS1 and PLS2. It has relatively low R^2's for both responses. The unconditional and conditional correlation between the two responses are quite high. From Figure 4.26 we

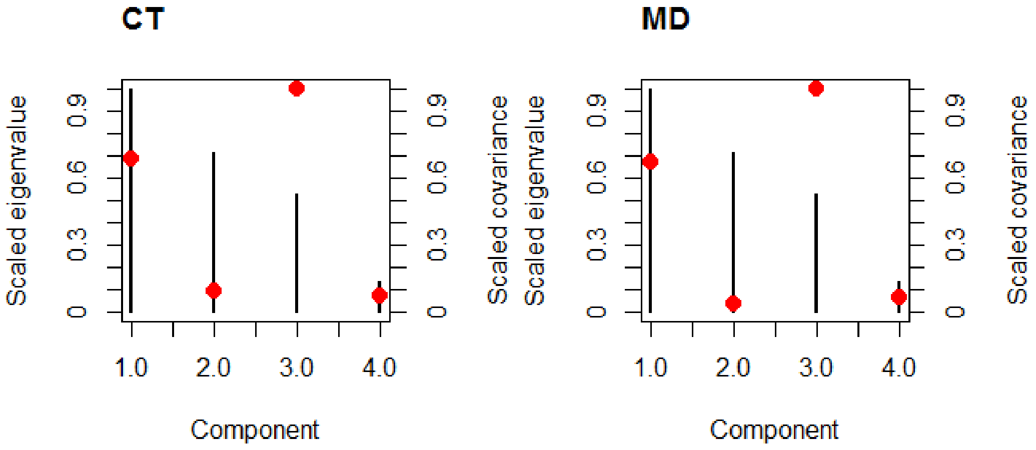

Figure 4.26: *Scaled eigenvalues(black lines) and covariances(red dots) plotted for each principal component for both responses in the LMP data.*

find that both responses have equal relevant components which is not only the first ones. The level of collinearity is not very high, which may lead to PLS1 performing better or the difference is negligible between the two methods.

NIR on corn

We have four possible responses. In Figure 4.27 we see that the different responses have different relevant components and there is high collinearity. The first component has the highest covariance(not shown) for all responses. The R^2's and ϱ(conditional correlation) are not possible to estimate with LS because $n < p$. The estimated unconditional correlation between the responses are found in table 4.11. Protein and Starch seem to have the same relevant components and at least one common 'difficult' relevant component, number 6. Therefore we choose to use Protein and Starch as responses. We also want to test PLS2 where there seem to be no common relevant difficult

Table 4.11: *The estimated correlations between the responses.*

	Moisture	Oil	Protein
Oil	-0.35		
Protein	-0.32	0.29	
Starch	-0.07	0.03	-0.80

component and hence use Moisture and Starch as responses. These responses are weakly unconditionally correlated.

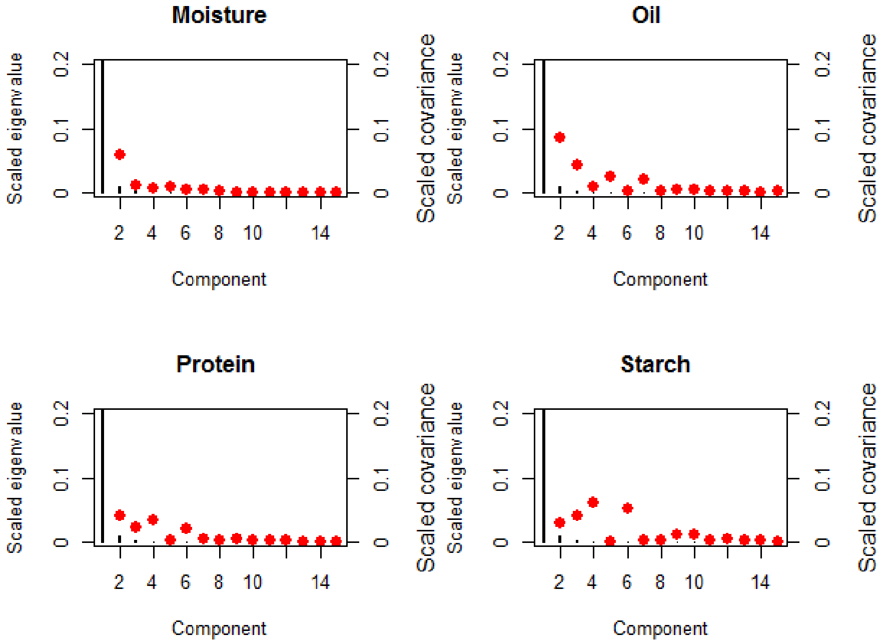

Figure 4.27: *Scaled eigenvalues(black lines) and covariances(red dots) plotted for each principal component for all responses in the NIR data.*

4.5.3 Prediction

LMP data

To estimate the prediction error we randomly draw a test-set of half the observation from the original data. We center the training and test-set by the means of the training data. Then we fitted the model with PLS1 method and PLS2 method and estimated the prediction error by the test-set. The prediction error was plotted against the number of components included in the model. Doing this several times with different test-set gave of course different result each time. But the difference was large and some pattern was detected. In some cases PLS1 and PLS2 performed extremly similar, as shown in Figure 4.28. But in most of the cases PLS2 predicted one of the responses best and PLS1 predicted the other response best(Figure 4.29).

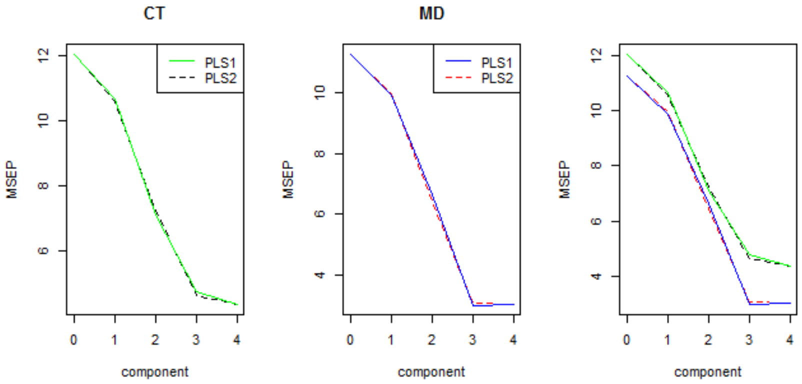

Figure 4.28: *The prediction error for each response plotted against the number of components included in the model for PLS1 and PLS2. In the third figure all four prediction error is plotted against each other. Here there was little visible difference between PLS1 and PLS2*

Choosing different test set each time having the four prediction errors in the same plot the two PLS2 prediction errors almost always ended up in the

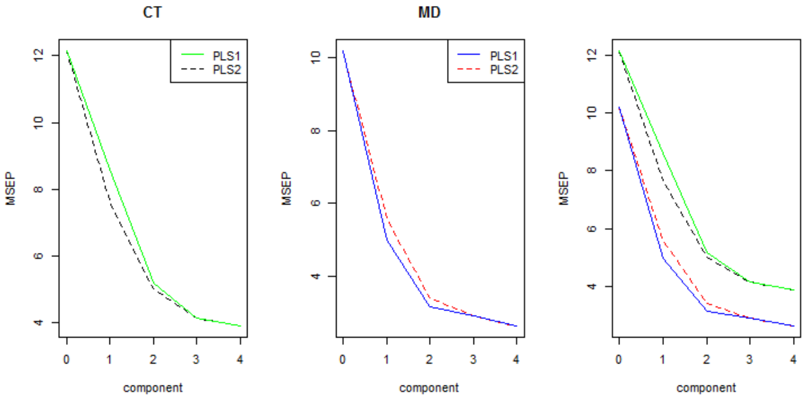

Figure 4.29: *Prediction error for each response plotted against the number of components included in the model for PLS1 and PLS2. In the third figure all four prediction error is plotted against each other. PLS2 predicted better than PLS1 for one of the responses.*

middle or the two PLS1 prediction errors ended up in the middle. PLS2 did not always predict the CT response best. For a different test set it predicted MD best. Very rarely did one of the methods(PLS1 and PLS2) have lowest prediction error for both responses. This indicates that there is some relation between the two methods. We fitted the model in eq. 2.25 with $\hat{\boldsymbol{\beta}}$'s from Figure 4.29 and plotted the $\hat{\alpha}$'s in Figure 4.30 for each component. It is not an exact linear combination, since the confidence intervals are wide. Only for the first component and the last component(LS solution) is the two $\hat{\boldsymbol{\beta}}_{PLS2}$ an exact linear combination of $\hat{\boldsymbol{\beta}}_{PLS1,CT}$ and $\hat{\boldsymbol{\beta}}_{PLS1,MD}$.

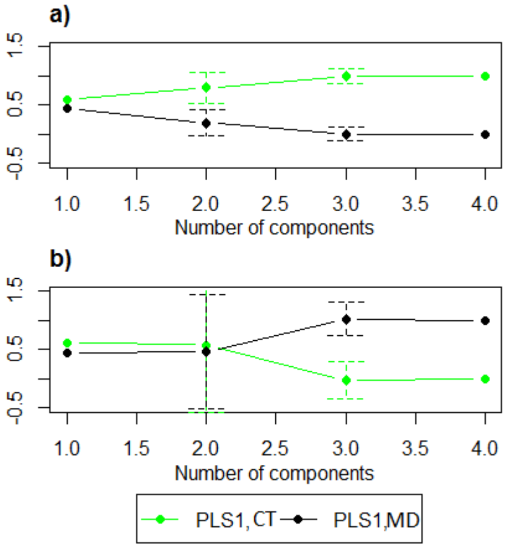

Figure 4.30: *The coefficients $\hat{\alpha}_1$ and $\hat{\alpha}_2$ plotted with 95 % confidence intervals for the α's. a) is when $\hat{\boldsymbol{\beta}}_{PLS2,CT}$ is used as response and b) is when $\hat{\boldsymbol{\beta}}_{PLS2,MD}$ is used as response.*

NIR on corn

We randomly split up the dataset into two groups, each with 40 observations. With training data we fit PLS1 and PLS2 models with Starch and Protein as responses. The test set is used to estimate the prediction error as described in sec 2.5. Result is shown in Figure 4.31. PLS1 and PLS2 produce extremly similar prediction errors. Choosing different test set did of course give different results, but the difference was not as large as with LMP dataset. We fitted the model in eq. 2.25 and plotted the coefficients $\hat{\alpha}_1$ and $\hat{\alpha}_2$ in Figure 4.32. The confidence intervals are not as wide here and it is partly due to large p, since p is equivalent to the number of observations in eq. 2.25. With one component included the two $\hat{\boldsymbol{\beta}}_{PLS2}$ is an exact linear combination of $\hat{\boldsymbol{\beta}}_{PLS1,Protein}$ and $\hat{\boldsymbol{\beta}}_{PLS1,Starch}$ as seen earlier. As more components are included, we approach the LS solution.

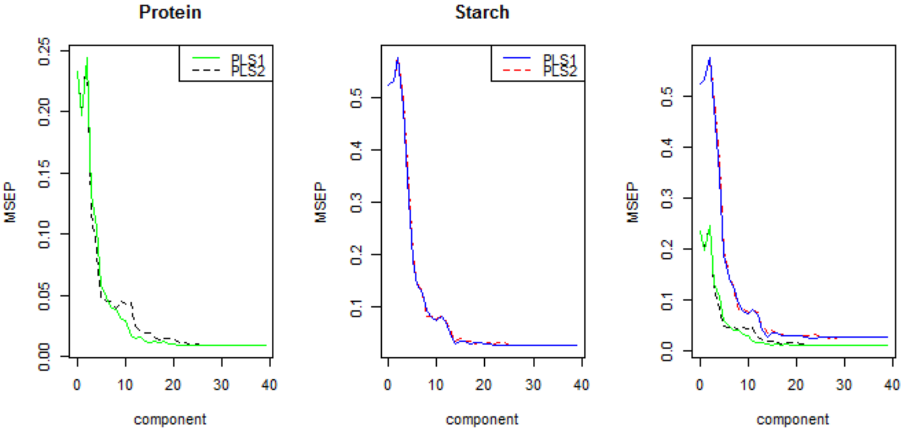

Figure 4.31: *prediction error for each response plotted against the number of components included in the model for PLS1 and PLS2. In the third figure all four prediction error is plotted against each other.*

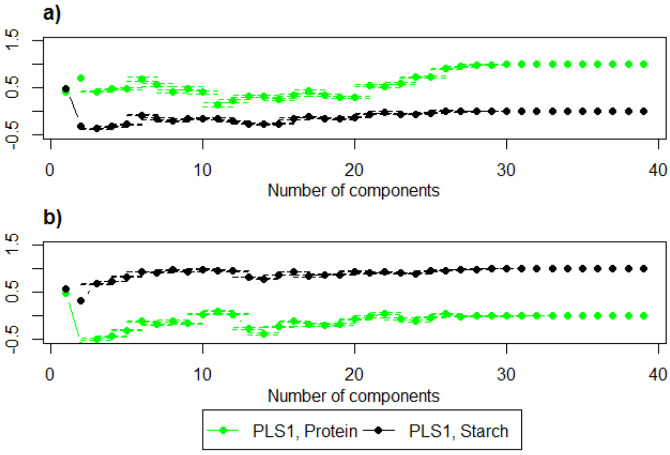

Figure 4.32: *The coefficients $\hat{\alpha}_1$ and $\hat{\alpha}_2$ plotted with 95 % confidence intervals for the α's. a) is when $\hat{\boldsymbol{\beta}}_{PLS2,Protein}$ is used as response and b) is when $\hat{\boldsymbol{\beta}}_{PLS2,Starch}$ is used a response.*

Using Moisture and Oil as responses we plotted the prediction in Figure 4.33. The figure shows that the two methods are very similar. When looking closer we see that PLS1 predicts slightly better than PLS2 as expected.

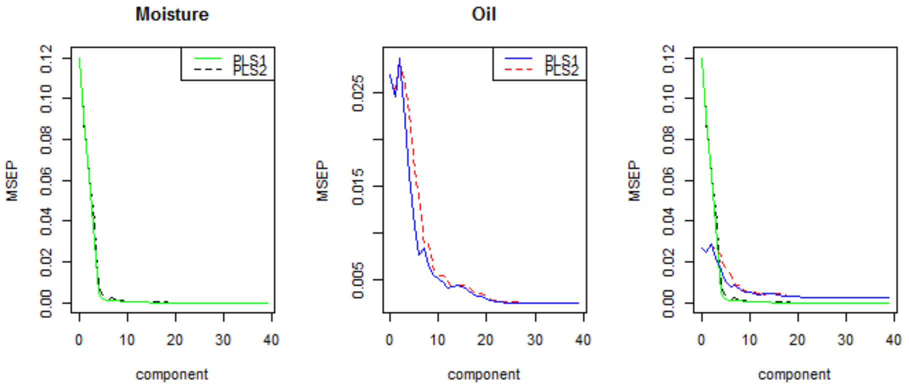

Figure 4.33: *Prediction error for each response plotted against the number of components included in the model for PLS1 and PLS2. In the third figure all four prediction error is plotted against each other.*

When fitting the model in eq. 2.25 and plotting the coefficients $\hat{\alpha}_1$ and $\hat{\alpha}_2$ in Figure 4.34 we see again that there is little symmetry between the two lines and that as more components are included it approaches the LS solution. With one component the two $\hat{\boldsymbol{\beta}}_{PLS2}$ is an exact linear combination of $\hat{\boldsymbol{\beta}}_{PLS1,Protein}$ and $\hat{\boldsymbol{\beta}}_{PLS1,Starch}$.

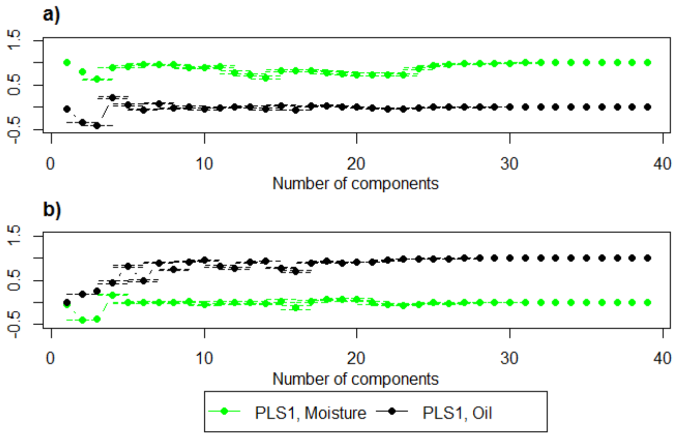

Figure 4.34: *The coefficients $\hat{\alpha}_1$ and $\hat{\alpha}_2$ plotted with 95 % confidence intervals for the α's. a) is when $\hat{\boldsymbol{\beta}}_{PLS2,Moisture}$ is used as response and b) is when $\hat{\boldsymbol{\beta}}_{PLS2,Oil}$ is used a response.*

Chapter 5

Discussion

5.1 PLS1

When we have small relevant eigenvalues the prediction error is large and in addition have large variation. It is clear that small relevant eigenvalues causes a problem. One of the reasons might be the estimation of which components that are relevant. As we saw from simulations without replication in sec 4.2.4 it is difficult to determine the correct relevant components, especially when small relevant eigenvalues are present.

If we have small relevant eigenvalues, increasing n or p was a very effective way to achieve a lower prediction error. The increase in n is obvious. The odd thing is that our simulation study showed that we can lower the prediction error by adding more explanatory variables. If it was true that adding more explanatory variables would decrease the prediction error we could have added variables with random numbers to any dataset and it would help. Another explanation could be that it has to do with the simulation. By adding variables(or components) uncorrelated to the response will somehow stabilize the the space orthogonal to the relevant space. And as a consequence it is easier to find the relevant components. This is just a hypothesis and further investigation is needed.

5.2 Comparison of PLS1 and PLS2

5.2.1 Estimation

We did some estimation to get a better understanding of how the two methods differ. From the results we found that when only including one component the two $\hat{\boldsymbol{\beta}}_{PLS2}$ is an exact linear combination of $\hat{\boldsymbol{\beta}}_{PLS1,Y_1}$ and $\hat{\boldsymbol{\beta}}_{PLS1,Y_2}$. This is consistent with both simulated data and real data. Using eq. 2.24 we find that with one component also the loading weight($\hat{\boldsymbol{w}}$) from PLS2 is a linear combination of the loading weights from PLS1. It should be possible to prove that

$$\hat{\boldsymbol{w}}_{PLS2} = c_1\hat{\boldsymbol{w}}_{PLS1,Y_1} + c_2\hat{\boldsymbol{w}}_{PLS1,Y_2}$$

holds for the first component. We have not been successful in doing so. We have searched the literature, but with no results. When adding more components, we saw that the two $\hat{\boldsymbol{\beta}}_{PLS2}$'s were not an exact linear combination of $\hat{\boldsymbol{\beta}}_{PLS1,Y_1}$ and $\hat{\boldsymbol{\beta}}_{PLS1,Y_2}$.

5.2.2 Prediction

By having only one common 'difficult' relevant component(A component with small relevant eigenvalues) it seemed as if PLS2 predicted better than PLS1 for one of the responses. It might be possible that one response works as a support response for the other when using PLS2. If one of the responses had one extra 'difficult' component that the other response did not have, PLS2 did not perform better.

Under the circumstance with only one common 'difficult' relevant component, it was also discovered that it only pays to use PLS2 when we have few observations and when high degree of collinearity is present. Earlier it was discovered that with PLS1 the combination of small relevant eigenvalues and components caused a high variation in prediction error. The upper right quadrant of Figure 4.22, all have parameter settings which results in small relevant eigenvalues. The variation between replications was large. Hence many or all of the effects seen in the upper quadrant is due to noise. Therefore it is difficult to estimate the effects of the other simulation parameters.

Among the simulation parameter setting that was investigated, non indicated that high correlation, both conditional or unconditional, gave better prediction for PLS2.

A possible critique to the analysis of variance in sec. 4.4.1, could be that the levels of R^2 for each response are not equal. If the first response has low level R^2 at 0.5 while the second response has low level at 0.6, it is not only the choice of *relpos* that differs between the two responses and hence the two tables 4.9 and table 4.10. The choice of levels of R^2 might then affect the intercept.

The true parameter values(or the true world as we have called it) is of course not possible to find, when working on real data. In some situations the estimated world was very different from true world. Especially when small relevant eigenvalues are present. The estimated world being quit different from the true world makes it difficult to detect if there is one common 'difficult' relevant component. And therefore it is difficult to recommend PLS2 over PLS1.

When we compared the two methods on real data, we found that for the LMP dataset PLS2 predicts better for one of the responses for some testsets. Since different test set gave different results, it was impossible to determine if PLS2 predicted better than PLS1. The difference between the two methods was small. For the other datasets we tested, PLS1 performed better or equally good as PLS2.

At the end of the study it was discovered that the simulation package does not pick the covariances at random when having two responses. This is seen in Figure 4.13, where both responses has the same R^2 and *relpos*, but the estimation errors are not equal. Because of this we can not be sure if it is an effect of *relpos* that causes PLS2 to predict better, or of it is an effect of covariances as well. Further investigation is needed to investigate if a response with high or low covariance on the 'difficult' relevant component works as a support response.

5.3 Further studies

Due to lack of time some issues have not been studied. Some of them are presented here.

What we have not studied is how the number of components included would effect the models ability to predict, but only chosen the number of components that gave lowest prediction error on average within each parameter setting.

We have found some few situations were PLS2 predicts better than PLS1 on average. There are however some other different PLS-methods which have shown to predict well. One is Canonical PLS(CPLS) [Indahl et al., 2009] which uses canonical correlations. When modeling two responses, the method has many similarities to PLS2. Further studies is needed to find if CPLS with multiple responses predicts better than PLS1.

The PLS algorithm presented in sec. 2.8.3 can be applied in situations with more than two responses. The dataset NIR on corn had four responses. As we have found situations when it pays to use two responses, are there possibilities that there are situations were it pays to use more than two responses?

The original LMP dataset had many missing observations for one of the responses. We have not dealt with missing observations in this study. It could be possible that PLS2, or other multiresponse method, can be used in some situations when there is missing data and even perform better than uniresponse models [Gangsei et al., 2015].

Even if PLS2 do not predict better than PLS1 in many cases it can be a helpful tool in explorative data analysis.

5.4 Conclusion

We have found some few cases where PLS2 on average predicts better than PLS1. This happens if we have only one common 'difficult' relevant component, few observations and high degree of collinearity. However the gain is small and the variation in the prediction error is large for the replications. In addition it is in practice difficult to estimate which components are relevant, if there are small relevant eigenvalues. As far as we have seen we can conclude with that PLS1 is a better option for prediction than PLS2.

Appendix A

Tables

Table A.1: *The estimated effects included all interactions up to 4-factor interactions between simulation parameters for prediction error.*

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.4305	0.0031	138.18	0.0000
relpos(1)	-0.0975	0.0031	-31.28	0.0000
R2(0.5)	0.2767	0.0031	88.80	0.0000
p(15)	0.0442	0.0031	14.20	0.0000
n(20)	0.0988	0.0031	31.69	0.0000
m(2)	-0.0744	0.0031	-23.87	0.0000
gamma(0.3)	-0.0818	0.0031	-26.25	0.0000
relpos(1):R2(0.5)	-0.0096	0.0031	-3.08	0.0020
relpos(1):p(15)	-0.0406	0.0031	-13.04	0.0000
R2(0.5):p(15)	-0.0028	0.0031	-0.89	0.3734
relpos(1):n(20)	-0.0587	0.0031	-18.84	0.0000
R2(0.5):n(20)	0.0233	0.0031	7.46	0.0000

p(15):n(20)	0.0234	0.0031	7.51	0.0000
relpos(1):m(2)	0.0460	0.0031	14.76	0.0000
R2(0.5):m(2)	-0.0151	0.0031	-4.84	0.0000
p(15):m(2)	-0.0186	0.0031	-5.96	0.0000
relpos(1):gamma(0.3)	0.0722	0.0031	23.17	0.0000
R2(0.5):gamma(0.3)	-0.0141	0.0031	-4.51	0.0000
p(15):gamma(0.3)	-0.0281	0.0031	-9.02	0.0000
n(20):m(2)	-0.0426	0.0031	-13.66	0.0000
n(20):gamma(0.3)	-0.0488	0.0031	-15.67	0.0000
m(2):gamma(0.3)	0.0590	0.0031	18.94	0.0000
relpos(1):R2(0.5):p(15)	0.0041	0.0031	1.30	0.1933
relpos(1):R2(0.5):n(20)	0.0044	0.0031	1.41	0.1595
relpos(1):p(15):n(20)	-0.0210	0.0031	-6.75	0.0000
R2(0.5):p(15):n(20)	-0.0068	0.0031	-2.17	0.0298
relpos(1):R2(0.5):m(2)	-0.0069	0.0031	-2.22	0.0268
relpos(1):p(15):m(2)	0.0183	0.0031	5.87	0.0000
R2(0.5):p(15):m(2)	0.0093	0.0031	2.98	0.0029
relpos(1):R2(0.5):gamma(0.3)	0.0038	0.0031	1.22	0.2212
relpos(1):p(15):gamma(0.3)	0.0296	0.0031	9.49	0.0000
R2(0.5):p(15):gamma(0.3)	0.0068	0.0031	2.18	0.0294
relpos(1):n(20):m(2)	0.0227	0.0031	7.27	0.0000
R2(0.5):n(20):m(2)	-0.0012	0.0031	-0.40	0.6921
relpos(1):n(20):gamma(0.3)	0.0421	0.0031	13.52	0.0000
R2(0.5):n(20):gamma(0.3)	-0.0020	0.0031	-0.63	0.5275
relpos(1):m(2):gamma(0.3)	-0.0433	0.0031	-13.88	0.0000
R2(0.5):m(2):gamma(0.3)	0.0073	0.0031	2.33	0.0197
p(15):n(20):m(2)	-0.0059	0.0031	-1.89	0.0583

p(15):n(20):gamma(0.3)	-0.0134	0.0031	-4.30	0.0000
p(15):m(2):gamma(0.3)	0.0196	0.0031	6.28	0.0000
n(20):m(2):gamma(0.3)	0.0327	0.0031	10.50	0.0000
relpos(1):R2(0.5):p(15):n(20)	0.0073	0.0031	2.35	0.0187
relpos(1):R2(0.5):p(15):m(2)	-0.0090	0.0031	-2.88	0.0040
relpos(1):R2(0.5):p(15):gamma(0.3)	-0.0067	0.0031	-2.14	0.0320
relpos(1):R2(0.5):n(20):m(2)	-0.0136	0.0031	-4.35	0.0000
relpos(1):R2(0.5):n(20):gamma(0.3)	-0.0053	0.0031	-1.70	0.0901
relpos(1):R2(0.5):m(2):gamma(0.3)	0.0053	0.0031	1.71	0.0877
relpos(1):p(15):n(20):m(2)	0.0058	0.0031	1.87	0.0609
relpos(1):p(15):n(20):gamma(0.3)	0.0143	0.0031	4.61	0.0000
relpos(1):p(15):m(2):gamma(0.3)	-0.0183	0.0031	-5.87	0.0000
relpos(1):n(20):m(2):gamma(0.3)	-0.0215	0.0031	-6.90	0.0000
R2(0.5):p(15):n(20):m(2)	0.0091	0.0031	2.92	0.0035
R2(0.5):p(15):n(20):gamma(0.3)	0.0076	0.0031	2.45	0.0145
R2(0.5):p(15):m(2):gamma(0.3)	-0.0063	0.0031	-2.04	0.0418
R2(0.5):n(20):m(2):gamma(0.3)	-0.0025	0.0031	-0.82	0.4143
p(15):n(20):m(2):gamma(0.3)	0.0067	0.0031	2.16	0.0305

Appendix B

Software

The thesis is written with Latex.

All calculations and dataplotting is done with R version 3.1.2 (2014-10-31).

The scripts can be found at

https://bitbucket.org/mtalseth/master-thesis.

Bibliography

[NIR, 2005] (2005). NIR of corn samples for standardization benchmarking. http://www.eigenvector.com/data/Corn/index.html.

[Bickel and Doksum, 1977] Bickel, P. J. and Doksum, K. A. (1977). *Mathematical Statistics. Basic Ideas and Selected Topics.* Prentice Hall, Inc., 1st edition.

[Dayal and MacGregor, 1997] Dayal, B. S. and MacGregor, J. F. (1997). Improved PLS algorithms. *Journal of Chemometrics*, 11:73 – 85.

[Frank and Friedman, 1993] Frank, I. E. and Friedman, J. H. (1993). A statistical view of some chemometrics regression tools. *Technometrics*, 35(2).

[Gangsei et al., 2015] Gangsei, L. E., Kongsro, J., Olsen, E. V., Røe, M., Alsvike, O., and Sæbø, S. (2015). Prediction precision for lean meat percentage in norwegian pig carcasses using 'hennesey grading probe 7'. evaluation of methods emphasized at exploiting additional information from computer tomography. Manuscript in preparation.

[Hastie et al., 2001] Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning, Data Mining, Inference, and Prediction.* Springer.

[Höskuldsson and Esbensen, 2003] Höskuldsson, A. and Esbensen, K. H. (2003). Multivariate data analysis: quo vadis? ii. levels of datamodeldata objectives and possibilities. *Journal of Chemometrics*, 17:45 – 52.

[Indahl et al., 2009] Indahl, U. G., Liland, K. H., and Næs, T. (2009). Canonical partial least squares - a unified pls approach to classification and regression problems. *Journal of Chemometrics*, 23:495–504.

[Johnson and Wichern, 2007] Johnson, R. A. and Wichern, D. W. (2007). *Applied Multivariate Statistical Analysis.* Person Education, Inc., 6th edition.

[Lay, 2012] Lay, D. C. (2012). *Linear Algebra and its Applications.* Pearson Education, Inc.

[Mardia et al., 1982] Mardia, K. V., Kent, J. T., and Bibby, J. M. (1982). *Multivariate Analysis.* Academic Press, Inc.

[Martens and Næs, 1989] Martens, H. and Næs, T. (1989). *Multivariate Calibration.* John Wiley & Sons.

[Montgomery, 2013] Montgomery, D. C. (2013). *Design and Analysis of Experiments.* John Wiley & Sons, Inc-, 8th edition.

[Næs and Helland, 1993] Næs, T. and Helland, I. S. (1993). Relevant components in regression. *Scandinavian Journal of Statistics*, 20:239 – 250.

[Sæbø, 2015] Sæbø, S. (2015). *simrel: Linear Model Data Simulation and Design of Computer Experiments.* R package version 1.1-0.

[Sæbø et al., 2015] Sæbø, S., Almøy, T., and Helland, I. S. (2015). Simrel - a versatile method for linear model data simulation based on the concept of a relevant subspace and relevant predictors. *Chemometrics and Intelligent Laboratory Systems*. (accepted for publication).

[Vining, 1998] Vining, G. G. (1998). A compromise approach to multiresponse optimization. *Journal of quality technology*, 30(4):309 – 313.

Norwegian University
of Life Sciences

Postboks 5003
NO-1432 Ås, Norway
+47 67 23 00 00
www.nmbu.no