ELSEVIER

Chemometrics and Intelligent Laboratory Systems 44 (1998) 99–121

Chemometrics and intelligent laboratory systems

Validation and verification of regression in small data sets

Harald A. Martens [a,b,*], Pierre Dardenne [c]

[a] Norwegian University of Science and Technology, Department of Physics and Chemistry, N-7034 Trondheim, Norway
[b] Denmark Technical University, Department of Biotechnology, DK-2800 Lyngby, Denmark
[c] Station de Haute Belgique (GRAGx), Rue de Serpont 100, B-6800 Libramont, Belgium

Received 2 October 1997; revised 17 March 1998; accepted 1 September 1998

Abstract

Four different methods of using small data sets in multivariate modelling are compared w.r.t. predictive precision in the long-run. The modelling in this case concerns multivariate calibration: $\hat{\mathbf{y}} = f(\mathbf{X})$. The study consists of a Monte Carlo simulation within a large data base of real data; $\mathbf{X}$ = NIR reflectance spectra and $\mathbf{y}$ = protein percentage, measured in 922 whole maize plant samples. Small data sets (40–120 objects) were repeatedly selected at random from the data base, each time simulating the situation of having only a small set of samples available for estimating, optimizing and assessing the calibration model. The 'true' apparent prediction error was each time controlled in the remaining data base. This was replicated 100 times in order to study the statistical performance of the four different validation methods. In each Monte Carlo replicate, the splitting of the available data set into calibration set and test set was compared to full cross validation. The results demonstrated that removing samples from an already limited set of available samples to an independent VALIDATION TEST SET seriously reduced the predictive performance of the calibrated models, and at the same time gave uncertain, systematically over-optimistic assessment of the models' predictive performance. Full CROSS VALIDATION gave improved predictive performance, and gave only slightly over-optimistic assessment of this predictive performance. Further removal of even more of the available samples for use in an independent VERIFICATION TEST SET gave in-the-long-run correct, although uncertain estimates of the predictive performance of the calibrated models, but this performance level had seriously deteriorated. Alternative verification of the model's predictive performance by the method of CROSS VERIFICATION gave results very similar to those of the cross validation. These results from real data correspond closely to previous findings for artificially simulated data. It appears that full cross validation is superior to both the use of independent validation test set and independent verification test set. © 1998 Elsevier Science B.V. All rights reserved.

Keywords: Small data sets; Multivariate modelling; Monte Carlo; Multivariate calibration; PLS; Regression; Small sample statistics

1. Introduction

1.1. The purpose of modelling: predictive ability

It is difficult to estimate a mathematical model's parameters optimally from a small set of empirical data and, at the same time, assessing the predictive ability of the obtained solution from the same empirical data.

This problem is acute in the field of multivariate calibration. On one hand it is important to estimate the calibration model parameters so as to minimize the true predictive error of the calibration model for future 'unknown' objects. This determines the qual-

[*] Corresponding author

PII: S0169-7439(98)00167-1

ity of the calibration obtained. On the other hand it is important to assess what this level of true predictive error is—otherwise, the calibration model cannot be put to sensible use.

But the problem is by no means limited to multivariate calibration. Mathematical modelling may be classified according to scientific ambition level, into causal, predictive and ad hoc modelling. Causal modelling is applicable, e.g., in classical physics and chemistry in cases when the causal understanding of the system is thought to be more or less complete. Predictive modelling is a more statistical approach that is applicable in situations where the knowledge may be incomplete, but where this can be made up for by empirical measurements.

In all causal and predictive modelling, the model parameters estimated based on data from one set of samples are normally intended to be valid also for future, unknown samples of the same general kind (the same statistical 'population'). Ad hoc modelling, describing just one given data set, is seldom of much use in science.

However, even causal or predictive modelling may end up with ad hoc problems. One problem concerns the input data: Causal or predictive models based on irrelevant samples or faulty measurement can be worse than useless. These problems will not be given attention here; the samples are presently drawn at random from one statistical population, and the data have already been used for other practical purposes, which indicates that the measurements are OK.

Instead, this paper addresses the topic of underfitting/overfitting, which presents another type of ad hoc problem in causal and predictive modelling: If too many independent model parameters have been estimated or too many alternative models have been tried and rejected, based on too few data, then much of the measurement noise in these data will be drawn into the model. The result is of more or less ad hoc nature showing wrong causality and low predictive ability, in spite of the opposite intent.

To acquire representative samples and to make good measurements is usually expensive and time-consuming. Consequently, in most practical situations the available number of samples with good data is painfully limited. On the other hand, in a normal calibration project, if there is much more calibration samples with good data than needed, then the calibration manager has not designed the calibration experiments cost-efficiently. Exceptions are some rare projects where very large data sets (like the one presently used) are generated for other reasons.

The purpose of the present study is to compare different methods for estimation of the calibration model parameters and the prediction error under various circumstances, based on data from a limited set of available samples. The actual, 'true' prediction error will then be estimated independently, based on a very large set of 'secret' control samples of the same general type. The comparison requires statistical modelling in several stages.

Section 2 of the paper first defines the chosen model and the parameters to be estimated from a given calibration situation (a 'calibration experiment'). Then a measure of the predictive performance of an obtained model is defined, from which a criterion for final model optimization is developed. Four different methods for assessing this predictive performance are explained. Finally, the Monte Carlo technique for statistical comparison of these methods over many calibration experiments is explained.

Section 3 describes the input data and the resampling strategy used. Section 4 illustrates the use of test sets and of resampling for model optimization and prediction assessment, and compares these methods. It then attempts to explain the observed effects by a separate simulation study of the variance of a variance estimate. Finally, it checks the potential for further improvement in model optimization.

2. Theory

The main purpose of the multivariate calibration is to estimate the predictor parameters from the data of the available samples, in such a way that the future predictions of **y** have as low prediction errors as possible. A second goal in multivariate calibration is to provide an estimate of this prediction error, based on the data from the available samples.

2.1. The linear multivariate model and its parameters

In the present case a linear regression model with one regressand and many regressors is studied. The model is applicable, e.g., for multiple linear regression (MLR) as well as various reduced-rank modifications such as the traditional stepwise modification (SMLR), Ridge Regression (RR), Principal Component Regression (PCR) and Partial Least Squares Regression (PLSR). More details on these calibration methods are given, e.g., in Martens and Naes [1]. It is also the basic linear structure model behind conventional Analysis of Variance (ANOVA).

PLSR is used here, but the conclusions are believed to be relevant for the other regression methods as well as to other types of mathematical modelling at the causal or predictive ambition levels.

The structure model of a regression with many **X**-variables may be written:

$$\boldsymbol{y} = 1b_0 + \mathbf{X}\boldsymbol{b} + \boldsymbol{f} \tag{1}$$

where **y** $(n \times 1)$ is the regressand measured in n samples, e.g., percent protein, measured by a traditional reference method in $n = 80$ samples; **1** $(n \times 1)$ is the conventional column of n 1's, included to satisfy matrix algebra formalism; **X** $(n \times K)$ is K **X**-variables measured in the same n samples, e.g., high-speed spectroscopic measurements at $K = 70$ wavelength channels; b_0 (1×1) and **b** $(K \times 1)$, (the regression coefficients) represent the parameters of the model; f $(n \times 1)$ represents the residuals after the modelling of **y** from **X**.

The calibration model parameters $[b_0, \mathbf{b}]$ are to be estimated from the data $[\mathbf{y}, \mathbf{X}]$ in the n samples. Various optimization criteria and estimation algorithms may be used, based on ordinary least squares, weighted least squares, generalized least squares, iteratively reweighed least squares/maximum likelihood, etc. In the present study, the ordinary least squares version of the PLSR for a single Y-variable is used, as described by Martens and Naes [1], with the input weight for each variable and object equal to 1 (i.e., no a priori standardization of variables). The regression coefficients are here estimated for A-dimensional PLSR models as:

$$\hat{\boldsymbol{b}}_A = \hat{\boldsymbol{W}}_A\left(\hat{\boldsymbol{P}}'_A\hat{\boldsymbol{W}}_A\right)^{-1}\hat{\boldsymbol{q}}'_A \tag{2a}$$

with offset:

$$\hat{b}_{0A} = \bar{y} - \bar{\boldsymbol{x}}\hat{\boldsymbol{b}}_A \tag{2b}$$

where $'$ means 'transposed'; $\bar{\mathbf{x}}(1 \times K)$ is the mean for the **X**-variables; $\bar{y}(1 \times 1)$ is the mean for the y-variable; $\mathbf{W}_{\mathrm{A}}$ $(K \times A)$ are the loading weights for **X** for PLSR components $a = 1,2,\ldots,$A; $\mathbf{P}_{\mathrm{A}}$ $(K \times A)$ are the loadings for **X** for PLSR components $a = 1,2,\ldots,$A; $\mathbf{q}_{\mathrm{A}}$ $(1 \times A)$ are the loadings for **y** for PLSR components $a = 1,2,\ldots,$A.

To make the following explanation simpler, we shall avoid the offset term $b_{0\mathrm{A}}$ (2b), instead including it as part of vector $\boldsymbol{b}_{\mathrm{A}}$, which then increases in size from K to $K + 1$, corresponding to writing Eq. (1) as:

$$\boldsymbol{y} = \mathbf{X}\boldsymbol{b} + \boldsymbol{f} \tag{3}$$

The PLSR model with $A = K$ corresponds to classical MLR. However, in practical cases when the **X**-variables are strongly intercorrelated and/or noisy, a reduced-rank regression model $(A < K)$ is often required, in order to avoid over-fitting. But the rank A then has to be estimated somehow from the data. During the estimation of a multivariate regression parameter vector $\boldsymbol{b}$ in Eq. (1) it is common to split the parameter estimation into two separate problems.

2.1.1. Estimating several alternative models of increasing complexity

The first problem, the estimation of the 'bulk' of the linear model parameters, consists of using an estimation method, e.g., a projection method, to estimate a whole series of alternative calibration models, $A = 0,1,2\ldots,A$Max:

$$[\boldsymbol{b}_0\, \boldsymbol{b}_1\, \boldsymbol{b}_2 \ldots\, \boldsymbol{b}_A \ldots\, \boldsymbol{b}_{A\mathrm{Max}}] \tag{4}$$

(The model $\mathbf{b}_0$ with $A =$ zero components consists only of the y-offset $b_0 = \bar{y}$).

Although all the models have the same physical appearance (a column vector of length $K + 1$), each successive model $A = 0,1,2,\ldots,A$Max reflects an increasing number of underlying parameters *independently* estimated, in order to describe more and more detail in the data.

For instance, in SMLR, the different $\boldsymbol{b}$-vectors could correspond to the increasing number of **X**-variables to be used in the final model (**X**-variables not used have zeros in **b**). In ridge regression it would correspond to decreasing size of the ridge parameter. In bilinear regression methods like PLSR it corresponds to increasing number of bilinear components (latent variables from **X**) used for modelling **y** (and **X**), i.e., including more columns into **W**, **P** and **q** in Eq. (2a).

2.1.2. Estimation of optimal model complexity

The second problem in the parameter estimation consists of using a quite different estimation technique to choose which one of these alternative models $\boldsymbol{b}_A$, $A = 0,1,2,\dots,A\text{Max}$, to use in Eqs. (2a) and (2b). In other words, this part concerns choosing the balance between underfitting (leaving valid structure in the data unmodelled) and overfitting (bringing in ad hoc measurement noise, etc., into the model).

In the case of bilinear regression the optimal model choice is here called '*A*Opt'. In PLSR it corresponds to the number of bilinear components $A = 0,1,2,\dots,A\text{Opt}$ to be used in Eqs. (2a) and (2b) in the final calibration model.

2.2. A measure of the predictive performance of an obtained model

Unless we know what predictive performance to expect from an estimated calibration model, the estimated model cannot be put to critical use in practical situations.

An estimated calibration model parameter $\hat{\mathbf{b}}_A$ may later be applied to an observed data vector $\boldsymbol{x}_i$ of the same general kind, in order to predict the value of y_i:

$$\hat{y}_{iA} = \boldsymbol{x}_i \hat{\boldsymbol{b}}_A \tag{5a}$$

If we had known the value of the reference value y_i after also using the traditional measurement method in the new sample, i, we could have assessed the actual prediction error:

$$\hat{f}_{iA} = y_i - \hat{y}_{iA} \tag{5b}$$

(apart from the unknown measurement errors in y_i).

In almost all practical cases such 'known' reference values for y_i are, of course, unknown; the purpose of a calibration project is usually to replace the cumbersome traditional reference analysis of **y** by a quicker (and often more precise) prediction from **X**!

Still, the main quality criterion for a practical calibration method is that it, again and again, results in calibration models that give low prediction errors when y_i is predicted in new samples $i = 1,2,\dots,n$ New of the general type calibrated for. So, it is important to be able to estimate this quality criterion from the available data.

In a real set of samples there will always be variations between the individual samples. Therefore, we are not only interested in the uncertainty of an individual sample's prediction, $\hat{y}_{iA}$. We want to have a low 'average' prediction error for the type of samples calibrated for.

A commonly used uncertainty measure is defined as the root-mean-square (rms) error of prediction (RMSEP) over a set of new samples $i = 1,2,\dots,n\text{New}$:

$$\text{RMSEP}(A) = \sqrt{\frac{\hat{\boldsymbol{f}}'_A \hat{\boldsymbol{f}}_A}{n\text{New}}} \tag{6}$$

where $\hat{\mathbf{f}}_A$ here represents the y-residual $\hat{f}_{iA}$ (Eq. (5b)) for a whole set samples, $i = 1,2,\dots,n\text{New}$.

RMSEP is a vector with one value for each model rank used ($A = 0,1,\dots,A\text{Max}$). Once we have chosen the optimal model rank *A*Opt, our final estimate of the prediction error is a single scalar value, and therefore here written in italics: Final RMSEP = RMSEP(*A*Opt).

Normally, the term RMSEP indicates that it is based on data from *n*New *independent* samples (i.e., samples not used for estimating parameters **b**). In contrast. the traditional corresponding measure based directly on the data from the n calibration samples is called the root mean error of calibration, RMSEC:

$$\text{RMSEC}(A) = \sqrt{\frac{\hat{\boldsymbol{f}}'_A \hat{\boldsymbol{f}}_A}{(n - df)}} \tag{7}$$

where $\hat{\mathbf{f}}_A$ here is the $(n \times 1)$ vector of calibration residuals of the n calibration objects, obtained from Eq. (1) after estimating the b-parameters, and df. is the number of degrees of freedom lost due to estimation of parameters $\hat{\mathbf{b}}$ (due to having estimated independent parameters and/or having tested and rejected alternative parameter combinations in the estimation process).

2.3. Criterion for choosing the 'best' model, AOpt

The number of PLSR components to use as A in Eqs. (2a) and (2b) in the final model has to be estimated. This parameter, *A*Opt, is chosen as the model rank that minimizes some criterion Crit for the different models $A = 0,1,2,\dots,A\text{Max}$:

$$A\text{Opt}: \min(\text{Crit}(A),\ A = 0,1,2,\dots,A\text{Max}) \tag{8}$$

Different criteria may be used in Eq. (8). One criterion is to let

$$\text{Crit}(A) = \text{RMSEP}(A) \tag{9a}$$

for $A = 0,1,\dots,A\text{Max}$.

In the present study this was modified slightly in order to make the modelling more robust: When several models, e.g., $A=5,6,7$ have about the same RMSEP, a lower-dimensional model is preferred over a higher-dimensional model. This was implemented as:

$$\mathrm{Crit}(A) = \mathrm{RMSEP}(A) + A * C \tag{9b}$$

for $A=0,1,\ldots,A\mathrm{Max}$, where RMSEP represents the validation RMSEP (irrespectively of how this has been estimated; see more detail below), and

$$C = 0.05 * s \tag{9c}$$

Parameter s provides automatic scaling of the modification, and reflects the maximum RMSEP(A) values between $A=0$ and $A=A\mathrm{Max}$.

This is very similar to the criterion used, e.g., in the UNSCRAMBLER program [2] on the first author's recommendation. The only minor modification is that the criterion was made extra robust for the present, totally unsupervised Monte Carlo simulations, by defining s as the square root of (RMSEP-$(0)^2$ + RMSEP($A\mathrm{Max})^2$), instead of just $s=$ RMSEP(0).

In summary, a general methodology has now been described for estimating $\boldsymbol{b}_A$ and RMSEP(A) for $A=1,2,\ldots,A\mathrm{Max}$, then for estimating AOpt from RMSEP via Crit, and finally, the quality control: Final RMSEP = RMSEP(AOpt). One problem remains: precisely how to estimate RMSEP from the available data. This will determine the actual value of AOpt and final RMSEP.

2.4. Optimizing and assessing a model's predictive performance from the available data

Most practical calibration experiments have to be based on a limited set of available samples. Let us designate the number of available samples as nAvail. Then we must solve all three of the problems; estimating $\boldsymbol{b}_A$, AOpt and RMSEP, from the nAvail samples.

The traditional statistical approach has been to estimate all necessary parameters in one step, based on distributional assumptions about the data. In the authors' experience, this is difficult to do correctly, and it often leads to over-fitting, as discussed, e.g., by Høskuldsson [3]. It implies in Eqs. (9a), (9b) and (9c) replacing RMSEP(A) from Eq. (4) by RMSEC(A) from Eqs. (5a) and (5b), and this requires, among other things, that a good estimator for the number of degrees of freedom is available, which is often not the case, e.g., for PLSR (Here: $df = 1 + A$).

There is thus a need for more pragmatic methods for estimating AOpt and final RMSEP. In order to avoid over-optimism due to over-fitting, it is important that the estimation of AOpt and RMSEP is independent from the estimation of $\boldsymbol{b}_A$. Four methods for doing this will now be explained.

2.4.1. Joint estimation of AOpt and RMSEP

In chemometrics there are presently two main practical principles for internal 'quality control' of a calibration model.

2.4.1.1. Independent validation test set. The first internal validation principle, and probably the intuitively simplest one, is to set aside some of the nAvail samples (nTVal samples) to be used as an independent validation test set, leaving the remaining nTCal $= n$Avail $-$ nTVal samples:

Estimate $\boldsymbol{b}_A$, $A=0,1,2,\ldots,A\mathrm{Max}$ (Eq. (4)) from the nTCal calibration samples (Eqs. (2a) and (2b)).

Use the nTVal independent test samples for estimating the validation prediction error RMSEP(A) for $A=0,1,\ldots,A\mathrm{Max}$ according to Eqs. (5a), (5b) and (6); call the result RMSETVal.

Estimate optimal model rank AOpt (sometimes called 'AOptT') from RMSETVal (Eqs. (8), (9a), (9b) and (9c)).

Final RMSEP = RMSEPTVal(AOptT).

2.4.1.2. Cross validation within the calibration set. The second internal validation principle is well established in statistics and is called cross validation. This method, based on a systematic resampling of the available data, was originally pioneered by M. Stone [4]. It consists of using all the nAvail samples both for estimating $\boldsymbol{b}_A$ and RMSEP(A) and then for estimating AOpt and final RMSEP. The clue here is to *repeat* the estimations in such a way that the danger of over-optimism is minimized.

Cross validation may done in many different ways. The cross validation used here is the full-model leave-one-out version described, e.g., by Efron and Tibshirani [5] and Martens and Naes [1].

Cross validation is here abbreviated 'X-Val'. The whole calibration modelling is repeated nAvail times, each time in turn leaving one new object out.

In each cross validation segment $i = 1, \dots, n$Avail: (10)

```
Take out sample $i$ as temporary 'test set'
For each model $A = 0,1,2,\dots,A$Max
Estimate $\boldsymbol{b}_{A,-i}$ (Eq. (4)) on the remaining $n$Avail $-$ 1 samples from Eqs. (2a) and (2b)
Predict $y_i$ from Eq. (3): $\hat{y}_{i,A} = \hat{\mathbf{x}}_i \hat{\mathbf{b}}_{A,-i}$
Estimate and store $\hat{f}\mathrm{Val}_{i,A} = y_i - \hat{y}_{i,A}$
```

Estimate validation RMSEP(A), $A = 0,1,2,\dots,$ AMax according to Eq. (5a), with $\hat{\mathbf{f}}_A = \hat{\mathbf{f}}\mathrm{Val}_A$, and call this RMSEPXVal.

Use the validation RMSEPXVal to estimate AOpt ($=$'AOptX') (Eqs. (8), (9a), (9b) and (9c)).

Final RMSEP = RMSEPXVal(AOptX).

This full-model version of cross validation is the one implemented, e.g., in the chemometric Unscrambler program, but different from the local-component cross validation used in several other chemometric programs, which only cross validates each component locally to find AOpt, and therefore does not attempt to give estimate of RMSEP.

The same robustification method 0.05 $* s$ (Eq. (9b)) was used for test set validation and for cross validation.

2.4.2. Separate estimation of AOpt and final RMSEP

In the conventional test set as well as in cross validation as described above, AOpt and final RMSEP are estimated at the same time. To what extent is this a problem?

In computationally heavy modelling methods like neural nets and genetic algorithms, where a very high number of alternative models are tested, each with several independent parameters, it is probably dangerous. One must expect the choice of an AOpt that minimizes RMSEP(A) in a test-set to be over-fitted to the particular noise structure in the test set and therefore under-estimate the true (but unknown) prediction error.

It has not been clear if this represents a problem with the simpler regression methods like, e.g., RR, PCR or PLSR. Therefore, the present study also includes separate estimation of AOpt (here termed 'validation', since that word is traditionally used in 'cross validation') and estimation of final RMSEP ('verification'). This will be done both in the case of using test set and using cross validation for validation:

2.4.2.1. Independent verification test set. In the case of test set, the nAvail samples are split in three (instead of just in two) sets—one calibration set with nTCal samples (for estimation of $\boldsymbol{b}_A$), and two test sets, a validation set with nTVal samples used for estimating AOpt, and a verification set with nTVer samples used for estimating final RMSEP:

Estimate $\boldsymbol{b}_A$, $A = 0,1,2,\dots,A$Max (Eq. (4)) from the nTCal calibration samples (Eqs. (2a) and (2b)).

Use the nTVal independent test samples for estimating the validation prediction error RMSEP(A) for $A = 0,1,\dots,A$Max according to Eqs. (5a), (5b) and (6); call the result RMSETVal.

Estimate optimal model rank AOpt ($=$'AOptT') from RMSETVal (Eqs. (8), (9a), (9b) and (9c)).

Use the nTVer independent test samples for *independently* estimating the verification prediction error RMSEP(A) for $A = 0,1,\dots,A$Max according to Eqs. (5a), (5b) and (6); call the result RMSETVer.

Final RMSEP = RMSEPTVer(AOptT).

2.4.2.2. Cross verification within the calibration set. In the case of resampling, all the nAvail samples are used both for estimating $\mathbf{b}_\mathbf{a}$, AOpt and final RMSEP. The resampling principle from conventional cross validation is used for separating $\mathbf{b}_\mathbf{a}$ estimation from AOpt and final RMSEP. In addition, a similar principle is at the same time used for separating the AOpt estimation from the final RMSEP estimation. This is here is termed 'cross verification'. The cross verification method extends the conventional 'leave-one-out cross validation' to a 'leave-two-out cross validation/verification; one for validation and one for verification':

In each cross validation segment $i = 1, \dots, n$Avail: (11)

```
Define sample $j = f(i)$
Take out the *two* of the samples, $i$ and $j$ $(i \neq j)$
For each model $A = 0,1,2,\dots,A$Max (Eq. (3))
```

```
Estimate $\boldsymbol{b}_{A,-i,j}$ on the remaining $n$Avail − 2 samples, from Eqs. (2a) and (2b)
Predict $y_i$ from Eq. (3): $\hat{y}_{i,A} = \hat{\mathbf{x}}_i \hat{\mathbf{b}}_{A,-i,j}$
Estimate and store $\hat{f}\text{Val}_{i,A} = y_i - \hat{y}_{iA}$
Predict $y_j$ from Eq. (3): $\hat{y}_{j,A} = \hat{\mathbf{x}}_j \hat{\mathbf{b}}_{A,-i,j}$
Estimate and store $\hat{f}\text{Ver}_{j,A} = y_j - \hat{y}_{j,A}$
```

Estimate validation RMSEP(A), $A = 0,1,2,\ldots, A$Max according to Eq. (5a) with $\hat{\mathbf{f}}_A = \hat{\mathbf{f}}\text{Val}_A$; call this RMSEPVal.

Estimate AOpt (= 'AOptX') from RMSEPVal (Eqs. (8), (9a), (9b) and (9c)).

Estimate verification RMSEP(A), $A = 0,1,2, \ldots, A$Max according to Eq. (5a) with $\hat{\mathbf{f}}_A = \hat{\mathbf{f}}\text{Ver}_A$; call this RMSEPVer.

Final RMSEP = RMSEPVer(AOptX).

The following definition of sample j was used here:

Define sample $j = f(i)$: If $i < n$Avail then $j = i + 1$, else $j = 1$

In the present study the samples had been randomly ordered, so no particular pattern was expected between adjacent samples. (Other rules for selecting verification samples might of course have been used instead, and more than one verification sample might also have been taken out at a time.)

This cross verification method appears to be a simplified version of the double cross validation methods proposed by Stone [4] and Hardy et al. [6]; but a detailed comparison has not yet been made.

In summary, the present study of validation techniques compares the performance of the four combinations:

estimating both AOpt and final RMSEP in a separate validation test set;

estimating both AOpt and final RMSEP in cross validation;

estimating AOpt in a validation test set, and estimating final RMSEP in a verification test set;

estimating AOpt in cross validation, and estimating final RMSEP in cross verification.

2.5. Statistical comparison of different validation and verification methods

Due to random effects in selection of the 'available' samples and in the measurements, one cannot draw powerful methodological conclusions based on one individual data set. Therefore, the job of having to calibrate and assess the final RMSEP on a small data set and checking the final RMSEP in a big control set was replicated many times in a Monte Carlo simulation. (Initial studies showed that $M = 100$ such modelling replicates gave sufficient precision in the conclusions) The results may be summarized by conventional averaging over the replicates. Final RMSEP was for instance averaged over M calibration modelling experiments (Monte Carlo replicates) by the Mean Final RMSEP:

$$\text{Mean Final RMSEP} = \sqrt{\frac{\sum_m^M \text{RMSEP}_m^2}{M}} \tag{12}$$

To summarize the variations from experiment to experiment, the between-replicate S.D. was calculated:

$$\text{S.D. of Final RMSEP} = \sqrt{\frac{\sum_{m=1}^{M} (\text{RMSEP} - \text{Mean Final RMSEP})^2}{M-1}} \tag{13}$$

The Monte Carlo simulation was repeated several times with different input parameters: different number of replicates M, different number of **X**-variables K, different number of available samples, nAvail.

2.6. Previous work

A similar study based on purely artificial data was recently published [7]. The data were generated according to a linear mixture model with random normally distributed noise added to **X** and **y**. That study indicated that the use of test set for validation was less successful than the use of cross validation, and that the use of a second test set for verification in some respects just made things even worse.

It also showed that the present leave-one-out cross validation and cross verification methods behaved very similarly for the artificial data.

Finally, the study indicated that a small set of randomly chosen samples always has a probability of not including certain variation phenomena that are present in the population at large. Consequently, any internal validation or verification method using the small set of available samples, be it based on test set or resampling, will tend to underestimate the 'true' final RMSEP slightly.

The present study is intended to check if these conclusions also hold for real data. Contrary to the work in Ref. [7], where validation test set was used with and without verification test set, validation and verification test sets will for simplicity always be used simultaneously in the following study.

2.7. Terminology

The term 'RMSE' is used when a common reference to both RMSEP and RMSEC is needed.

The abbreviations 'Cal', 'Val', 'Ver' and 'Con' are used for representing calibration, validation, verification and control-set check.

Examples: RMSEPVal, RMSEPVer, RMSEPCon

When deemed necessary for clarity, the letters 'T' and 'X' will be used for representing Test set and cross validation ('X-Val').

Examples: AOptT, AOptX, RMSEPTVal, RMSEPXVal, RMSEPTVer, RMSEPXVer, RMSEPTCon.

3. Experimental

3.1. Chemical and instrumental data

A total of 902 samples of maize hole plants were supplied from several European breeding companies and from Belgian field experiments, and analyzed as described in more detail by Sinnaeve et al. [8].

Each sample was analyzed w.r.t. total protein percentage in dry matter ($\mathbf{y}$) by standard Kjeldahl method (Mean: 8.9% protein, S.D. 1.2%). NIR reflectance spectra were measured in an NIRSystems 5000 scanning instrument (1100–2500 nm in steps of 2 nm). The instrument reflectance R was converted to ($\mathbf{X}$) optical density by taking the conventional $\log(1/R)$. Sinnaeve et al. reported an RMSEP of 0.41 percent protein, i.e., a squared correlation of $r^2 = 1 - (0.41/1.2)^2 = 0.88$, when using all these data directly in multivariate calibration. The expected measurement uncertainty level of the reference Kjeldahl measurements $\mathbf{y}$ (reproducibility S.D. over 126 days) was found to be 0.11%, indicating that there is a residual truly unmodellable RMSEP of $\sqrt{(0.41^2 - 0.11^2)} = 0.39\%$ protein.

Although Sinnaeve et al. showed some advantage in spectral preprocessing of the NIR data, this was, for simplicity, not used here. To save computer simulation time, averages of 20 nm wavelength wide spectral segments were taken, yielding 70 $\mathbf{X}$-variables to be used in most of the experiments. A slight further loss of predictive ability may be expected due to the loss of spectral resolution. Lower number of wavelength channels will also be reported for comparison.

3.2. Estimation methods

3.2.1. Monte Carlo simulation of independent calibration experiments

Replicate sets $m = 1,2,\ldots,M$ were drawn randomly from the total set of 902 objects.

Each replicate data set m, of nAvail $= 80$ objects, was modelled as if it were the only data available. Then the predictive performance of the various calibration models was checked using the remaining nControl $= 822$ 'secrect' objects.

Finally, the results were averaged over the M replicated experiments, as shown in Eqs. (12) and (13).

Initial studies demonstrated that $M = 100$ independent replicate Monte Carlo experiments were sufficient for all practical purposes, as will be illustrated by some results for larger simulation sets.

3.2.2. Data analysis within each of the calibration experiments

Four levels of estimation was performed in each replicate m, according to the theory described in Section 1:

Calibration: Estimation of parameters in AMax + 1 alternative models (estimating $[\mathbf{b}_0\ \mathbf{b}_1\ \mathbf{b}_2\ \mathbf{b}_3 \ldots \mathbf{b}_{A\mathrm{Max}}]$);

Validation: Model optimisation/choice of one of the alternative models (estimation of RMSEPVal(A), $A = 0,1,\ldots,A$Max, plus choosing AOpt and consequently the final model, $\hat{\mathbf{b}}$);

Verification: Internal quality control (estimation of RMSEPVer(A), $A = 0,1,\ldots,A$Max) and, Model control: Checking the performance of AOpt, $\hat{\mathbf{b}}$, RMSEPVal and RMSEPVer (estimation of the 'true' prediction error in the control set, RMSEPCon).

For each replicate data set $[\mathbf{X}\ \mathbf{y}]_m$ this was done twice.

(1) Test set: For checking the test validation, the set of nAvail = 80 samples was split into $n = 40$ calibration objects, nTVal = 20 validation test samples and nTVer = 20 verification test objects. The remaining nControl = 822 samples were used as 'secret' control set.

(2) Resampling: For checking the full resampling with cross-validation and -verification, the data from the same nAvail = 80 samples were used for both calibration, validation and verification. Again, the same nControl = 822 remaining samples were used as 'secret' control set.

Results with other number of available samples (nAvail = 40 and 120) will also be reported, for comparison.

Other ways of splitting the nAvail samples into validation and verification test sets were also checked (e.g., more validation samples than verification samples); they gave the same general conclusions and will therefore not be reported here.

A simple extra simulation study was made for studying the uncertainty of a final RMSEP estimate as a function of number of objects used in the estimation (Matlab pseudocode):

```
nTests = [5 10 20 50 100]
for i = 1:length(nTests)
nTest = nTests(i);
for Rep = 1:10000
fHat = randn(nTest,1);
RMSE(Rep,i) = sqrt(fHat' * fHat/nTest);
end % for Rep
end % for nTest
```

The cumulative distribution of RMSEP over the replicates was plotted for the different numbers of test samples, nTest.

3.3. Software

The computations were performed in MATLAB [9], using the first author's software.

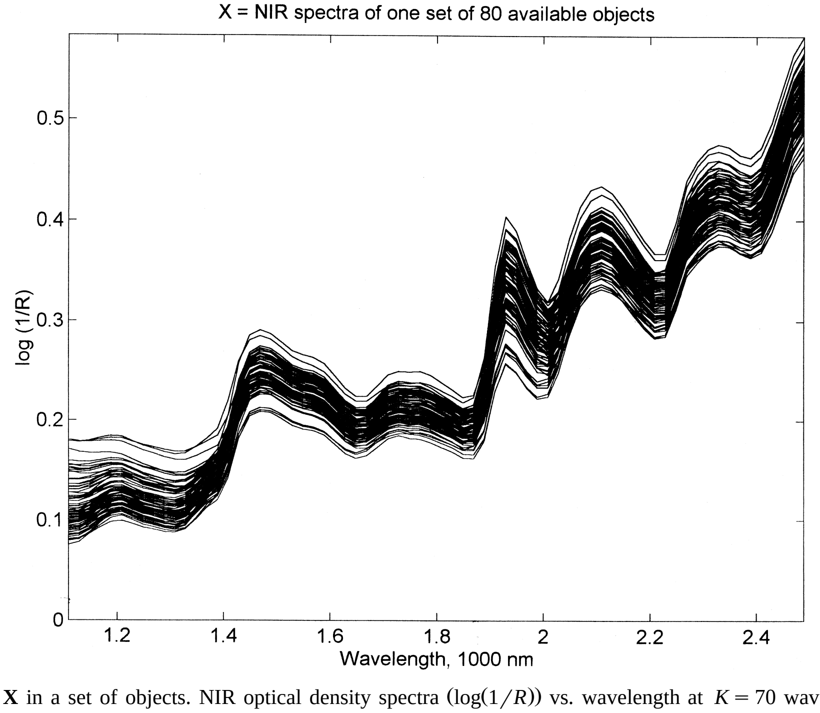

Fig. 1. Available data **X** in a set of objects. NIR optical density spectra (log(1/R)) vs. wavelength at $K = 70$ wavelength channels for a representative set of nAvail = 80 whole maize samples.

4. Results and discussion

4.1. Splitting one available data set into calibration, validation and verification sets (A single replicate)

Fig. 1 shows the **X**-data of one typical experiment among the 100 replicated experiments. The NIR spectra of a small set of (nAvail = 80) maize samples, measured by reflectance analysis of ground whole maize plants, and here represented at 70 NIR wavelength channels. On the basis of these **X**-data and the corresponding reference measurements **y** of protein percentage, a multivariate calibration model is to be estimated and assessed.

Fig. 2a,b,c illustrates in more detail how the available data in this experiment were used in the case of test set validation and verification: The nAvail = 80 objects in Fig. 1 have been split at random into n = 40 objects in a calibration set, nTVal = 20 objects in a validation test set and nTVer = 20 objects in a verification test set.

Multivariate calibration by PLSR was applied to the n = 40 objects in Fig. 2a. Eqs. (8), (9a), (9b) and (9c) were used for automatically choosing the optimal model rank, AOpt, which in this particular case was found to be 13. Fig. 3a shows the predicted $\hat{y}_i$ vs. the measured y_i (percent protein) for the calibration regression model that uses AOpt = 13 components. The corresponding predictions in the validation test set and the verification test set are given in Fig. 3b and c.

In the present study we actually have **X**- and y-data for a much larger set of additional objects (nControl = 822); their spectra are given in Fig. 2d.

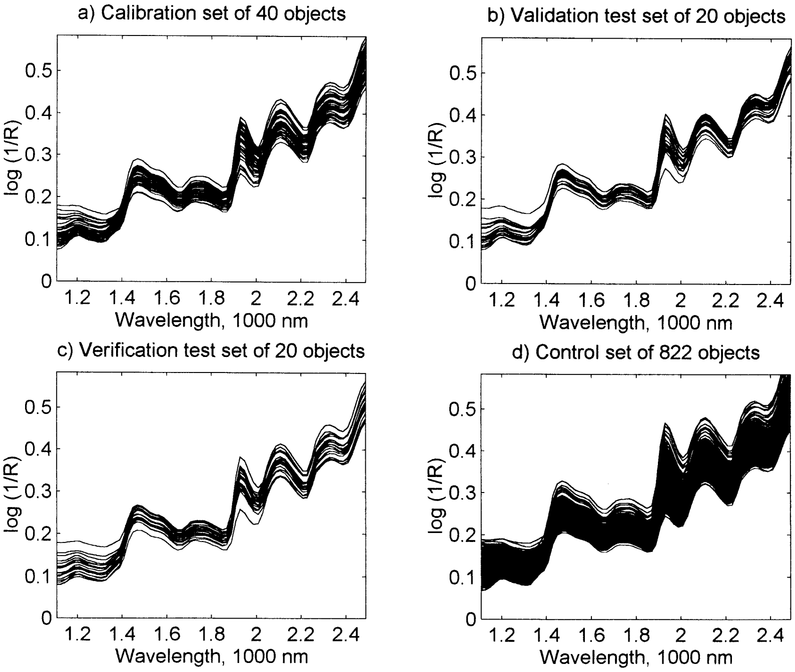

Fig. 2. Test set validation and verification: NIR spectra. The 80 available objects in Fig. 1 were randomly separated into (a) a calibration set of nCal = 40 objects, from which the regression coefficient vectors $\boldsymbol{b}_A$ are estimated for several alternative models with different number of PLSR components (rank), $A = 0,1,2,\ldots,A$Max. (b) A validation test set of nVal = 20 objects from which the optimal model rank AOpt is estimated, and (c) a verification test set of nVer = 20 objects from which final RMSEP for the regression coefficient vector $\boldsymbol{b}_{A\mathrm{Opt}}$ is estimated. (d) A control set of nControl = 822 objects, from which the 'true' prediction error occurring when $\hat{\mathbf{b}}_{A\mathrm{Opt}}$ is applied to the whole population, can be estimated.

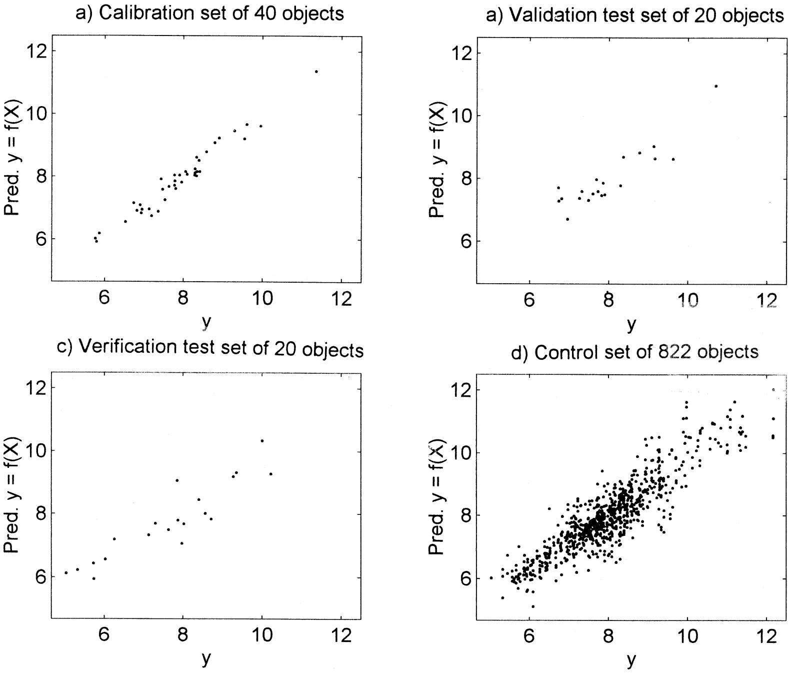

Fig. 3. Test set validation and verification: Predicted vs. measured **y**. Percentage protein measured by Kjeldahl-N (abscissa) and predicted from NIR spectra (ordinate) for the four sets of objects in Fig. 2a–d), using AOpt = 13 PLS components.

Since we are studying the analysis of small available data sets, this larger set is presently regarded as 'secret'—in real situations it would not be available. The prediction performance of the model with AOpt = 13 in this 'secret' control set is shown in Fig. 3d.

Together, the subplots in Fig. 3 indicate that the objects vary quite a bit in how well y_i and $\hat{y}_i$ fit together, and that the fit for the calibration data (Fig. 3a) appears to be better than in the three other sets.

The optimal number of components, AOpt, was determined from the RMSEP(A) summary of the y-predictions for $A = 1,2,\ldots,A$Max in the validation test set of 20 objects (Fig. 3b), modified according to the criterion in Eqs. (8), (9a), (9b) and (9c). Although $A = 15$ components showed the lowest computed validation RMSEP, the correction in Eqs. (9a), (9b) and (9c) led to AOpt = 13. The RMSEP for this test validation set ('TVal') is shown by the circles in Fig. 4b, with the automatically obtained model rank AOpt = 13 marked. Fig. 4a and c show the corresponding RMSEP for the calibration set of 40 objects and the verification set of 20 objects. The apparent true RMSEP from the control set of 822 objects, ($*$) in Fig. 4d, is included in the three other subplots for comparison.

It should be noted that in practical calibration situations with higher risk for non-stationarity problems (later instrument drift or unexpected sample qualities), an even lower number of components should have been used as AOpt, say, 10. However, nonstationarity problems are considered more or less irrelevant for the present study, since it probably affects the different validation methods in the same way. The control data set shows that AOpt = 13 was about right in this particular case; all models with 10–15 components performed well.

Fig. 4 confirms that the fit in the calibration set is too good, compared to the predictive performance in the control set. It also shows that RMSEP in the validation and verification sets changes somewhat erratically.

(It may be noted that the first couple of PLSR components have little or no predictive power for **y**. This is quite common for NIR data; there are sys-

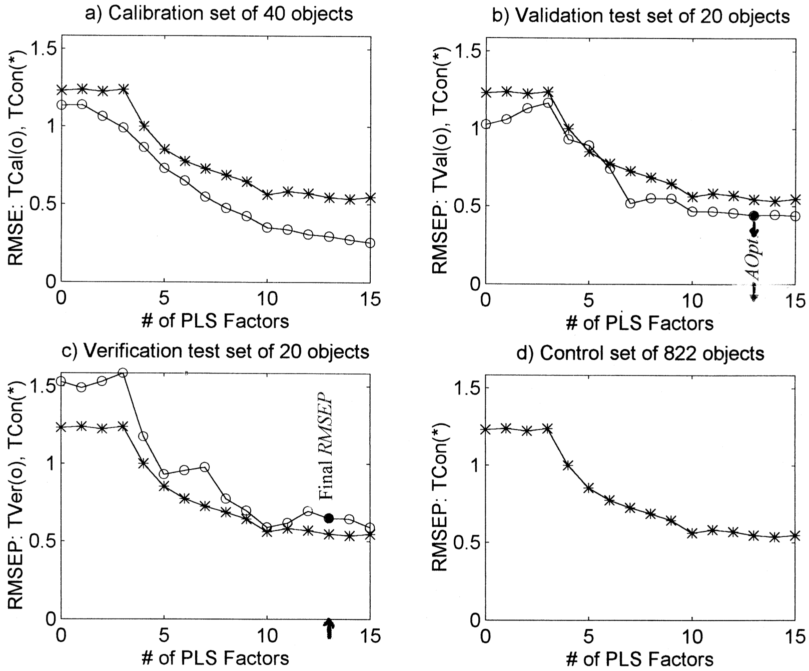

Fig. 4. Test set validation and verification: The effect of PLSR model rank. Prediction error for **y** (Root Mean Squared Error; RMSE) vs. number of bilinear components ('PLS Factors'), for the sets of objects in Fig. 2a–d). The 'true' control set prediction error RMSEP ($*$) from (d) is included also in (a–c) for comparison.

tematic optical variation between the samples, such as diffuse light scattering and surface reflection variations, that dominate the NIR reflection measurements (Fig. 1) with little or no relationship to **y**.)

Was this splitting of the available data set into three separate data sets an optimal usage of the 80 available objects' data? What is the probability that the different variation phenomena are sufficiently represented in each of these three data sets?

4.2. Calibration with validation and verification test sets, in 100 replicate experiments

Fig. 5 shows the results from $M = 100$ replicated analyses of the type illustrated in Fig. 4. The figure shows that while RMSEP of the replicates behave reasonably similar in the calibration set of 40 objects (Fig. 5a) and the control set of 822 objects (Fig. 5d), it varies strongly in the validation test set and the verification test set, (Fig. 5b and c), of 20 objects each.

Fig. 6 shows the Mean Final RMSEP(A) of the results in Fig. 5, computed for components $A = 0,1,\dots,A\text{Max}$ (in analogy to Eq. (12), defined at $A = A\text{Opt}$). The control set results (Mean RMSEP 'TCon', $*$ in Fig. 6d have been included also in Fig. 6a–c for comparison. The figure shows that while the RMSEC 'TCal' in Fig. 6a on the average underestimates the true apparent prediction error, the validation test set (Fig. 6b) and the verification test set (Fig. 6c) RMSEP on the average give results very similar to the mean "true" average RMSEP (Fig. 6d). Thus, the RMSEP averaged over, e.g., 100 replicates was more stabile than RMSEP of the individual replicates, as expected.

In each of the 100 individual replicates, the optimal number of components, AOpt, was automatically selected from the minimum of the criterion in Eqs. (9a), (9b) and (9c), and the final RMSEC and RMSEP values estimated for this replicate at AOpt components.

Fig. 7 summarizes the results from the various estimators for the final RMSEP: Final RMSEC ('Cal'),

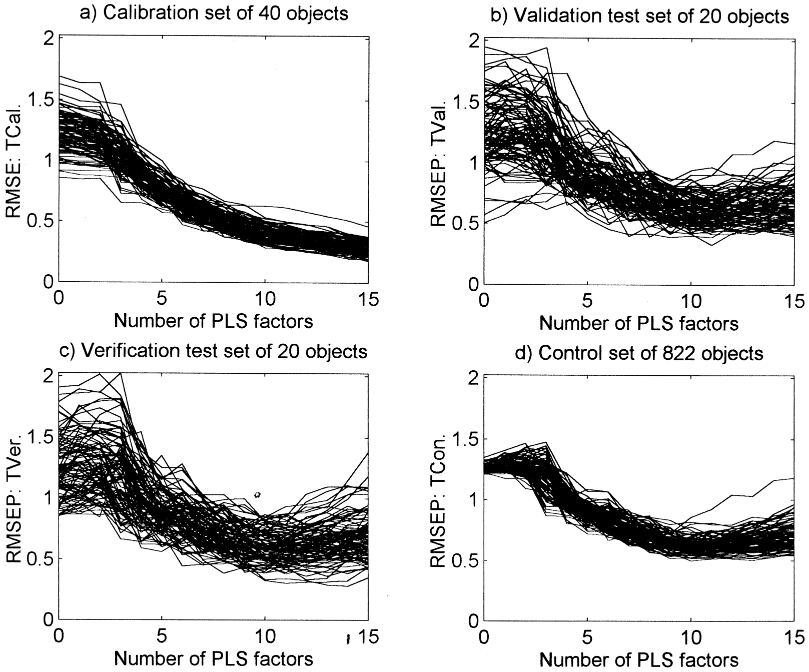

Fig. 5. Test set validation and verification: Individual replicates. RMSE for **y** vs. number of PLS factors A, for $M = 100$ replications of randomly selecting a set of nAvail = 80 'available objects' from the total set of 902 objects, splitting this in 40 calibration, 20 validation and 20 verification objects, and using the remaining 822 for control, as exemplified for one replicate in Figs. 2–4.

final RMSEPVal('Val'), final RMSEPVer('Ver') and the 'truth'—the final RMSEPCon ('Control'). For each of these estimators the circle represents the main Monte Carlo results—the Mean Final RMSEP, summarizing the estimated RMSEP$_m$ at AOpt$_m$ over modelling replicates $m = 1,2,\dots,M$ by Eq. (12), when test set was used. (The cross symbols will be explained later). The vertical bar represents ± one uncertainty standard deviation of the M replicate RMSEP$_m$ results over the replicates Eq. (13). This reflects the uncertainty of RMSEP$_m$ estimates from each single replicate 'experiment'. (The uncertainty of the Mean Final RMSEP results from Monte Carlo simulations, basis for the subsequent method comparisons, is on the order of $1/\sqrt{M} = 1/10$ smaller).

The figure confirms that final RMSEC (abscissa #1) underestimates the 'true' RMSEP (#7), as is well known; the present estimator of degrees of freedom probably renders it useless as estimator for final RMSEP.

More significantly, the figure also shows the validation test case estimator RMSEPVal (#3) to underestimate the 'true' RMSEP (#7), and therefore calls for the use of a second test set for independent verification.

The figure also shows that the RMSEPVer (#5) on the average in fact gives a very good estimate of the corresponding 'true' control RMSEP (#7).

4.2.1. Does estimating AOpt cause some overfit?

Considering that the RMSEPVal(A) and RMSEPVer(A) results for $A = 1,2,\dots,A$Max in Fig. 6 behaved almost identically, how can they cause so different final RMSEP estimates in Fig. 7? The answer must lie in the fact that the validation test set was used for estimating AOpt: The chosen model will tend to fit to the measuring errors etc. in this validation test set, and therefore the final RMSEPVal underestimates the true apparent RMSEP. A further support for this hypothesis is that the variation between replicates (shown as vertical bar ±1 S.D.) is seen to be lower for RMSEPVal than RMSEPVer; the estimation of AOpt seems to have removed some residuals in the validation set that ideally should not

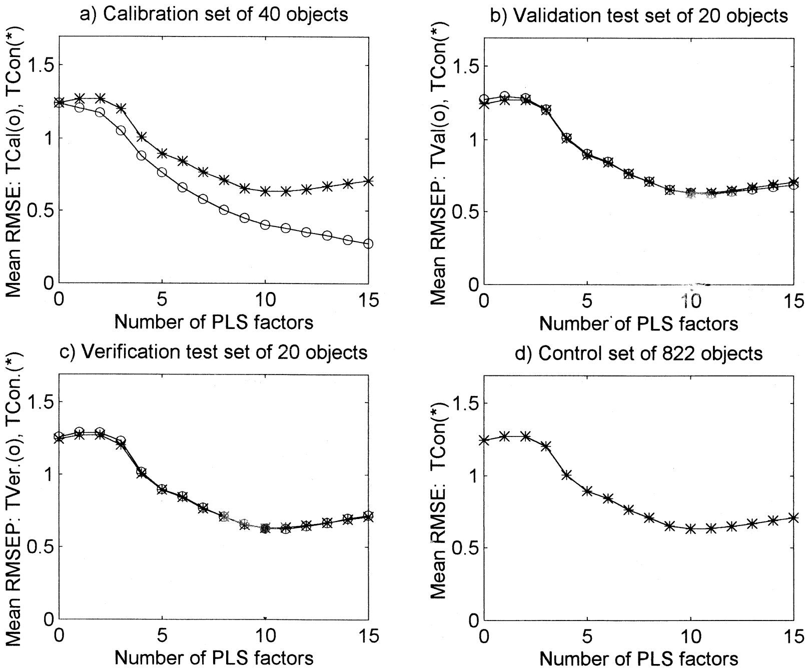

Fig. 6. Test set validation and verification: Mean over 100 replications. RMSE for **y** vs. number of PLS regression factors A, averaged over the $M = 100$ replications in Fig. 5. The 'true' control set prediction error RMSEP ($*$) from the control set (d) is included also in (a–c) for comparison.

have been drawn into the model. Hence, estimating AOpt can cause over-fitting.

It should be noted that although RMSEPVer averaged over 100 replicates may be an 'unbiased' estimator of the 'true' RMSEP in the long run, its variation between individual replicates is high. RMSEPVer estimated from the available data in an individual calibration experiment was sometimes much too high or much too low.

4.3. Calibration with resampling, in the same 100 experiments

Fig. 8 shows the results with resampling instead of using test sets, for the data in the same 100 replicate experiments of nAvail = 80 objects and 822 remaining control objects as those used previously with test sets. Compared to Fig. 5 it shows that the validation (b) and verification (c) RMSEP now behaves much less erratically. This is to be expected, since now there are 80 objects behind each curve, instead of just 20.

Fig. 9 summarizes the data in Fig. 8, in analogy to Fig. 6 for the use of test sets. The 'true' cross validation RMSEP (solid curve, Fig. 9d) is included in Fig. 9a–c for comparison.

The figure shows that both the cross validation (Fig. 9b) and the cross verification (Fig. 9c) RMSEP, on the average, give curves very similar to the 'true' RMSEP curve (Fig. 9d).

Fig. 9 also shows that this 'true' prediction error RMSEPVal, obtained with cross-validation (Fig. 9d solid curve), is lower than that obtained with the use of test set validation (Fig. 6d, included in Fig. 9a–d as a dotted curve).

4.4. Comparison of test sets *vs.* resampling

The cross-validation results are compared to the test set results in more detail in Fig. 7. The crosses represent the Mean Final RMSEP (Eq. (12)) obtained by cross-validation estimation of $A\text{Opt}_m$, and the vertical bar the variation range of ±1 uncertainty standard deviation of the individual modelling replicates (Eq. (13)). Again, the uncertainty of the Monte

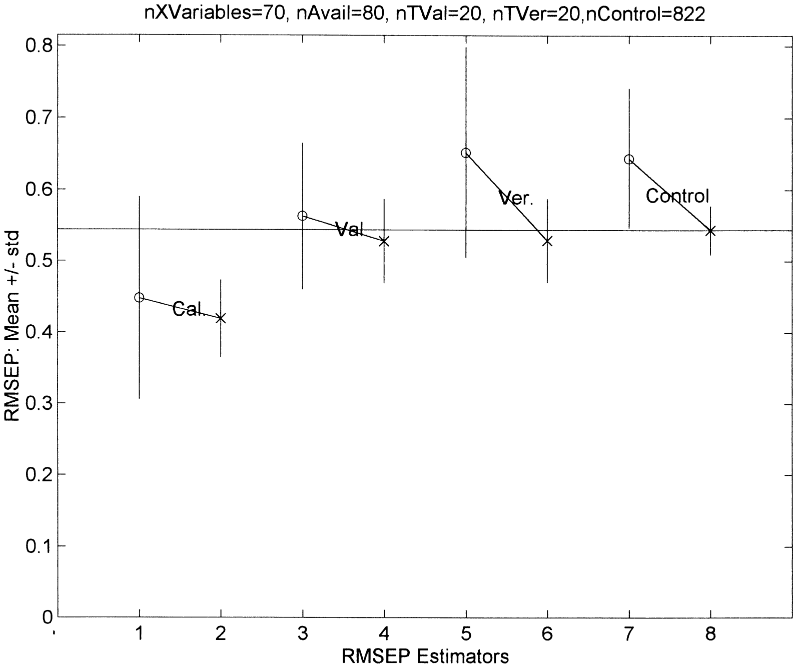

Fig. 7. Comparing final RMSEP of $\hat{\mathbf{b}}_{A\mathrm{Opt}}$ from the different ways to use the available samples. Prediction error RMSEP for **y**, estimated by four methods (Calibration, Validation, Verification and Control), in each case at the PLSR rank *A*Opt estimated by either test set validation (*A*OptT, 'o') or by cross validation (*A*OptX, 'x'): Mean (rms, Eq. (12)) ± S.D. (Eq. (13)) of the 100 replicates. Same conditions as in Fig. 5Fig. 6. Vertical lines represent ± 1 S.D. of the final RMSEP. Solid horizontal line corresponds to the 'true' RMSEP when using cross validation/verification.

Carlo results Mean Final RMSEP is smaller by a factor of about $1/\sqrt{M} = 1/10$.

A horizontal line has been drawn though the 'true' (control) RMSEP for the cross-validation case, to facilitate the visual comparisons.

Fig. 7 first of all confirms that the 'true' control RMSEP in this case of cross validation (#8) is, on the average, considerably lower than that obtained by test set validation (#7), (0.55 vs. 0.65). The uncertainty range (Eq. (13)) of this 'true' RMSEP is also much lower than that obtained with test set validation (0.09 vs. 0.18).

Secondly, Fig. 7 shows that the cross-validation RMSEP (#4) in the long run slightly underestimates the corresponding 'true' RMSEP (#8), although much less so than in the test set validation case (#3 vs. #7). The variation of the cross-validation RMSEP (#4) between replicates is much smaller than that for the test set RMSEP (#3), but higher than the variation of the 'true' RMSEP (#8).

Thirdly, Fig. 7 shows the cross-validation RMSEP (#4) is marginally lower than that of cross-verification RMSEP (#6), on the average (rms mean) 0.528 vs. 0.529. A difference was indeed expected, since *A*Opt was estimated from the cross validation data. The differences is in fact surprisingly small in comparison to the corresponding difference between test set validation and test set verification; this needs further investigation.

Fig. 10 displays the distributions of the estimated and 'true' RMSEP for 1000 individual replicates, under the same condition shown in the previous figures. *A*Opt was in each replicate estimated by test set validation in Fig. 10a and b and by cross-validation in Fig. 10c and d. The left figures show validation results, the right figures verification results.

The figure shows that the 'true' RMSEP (abscissa) has more narrow distributions than the corresponding estimated final RMSEP (ordinate) in all four subplots.

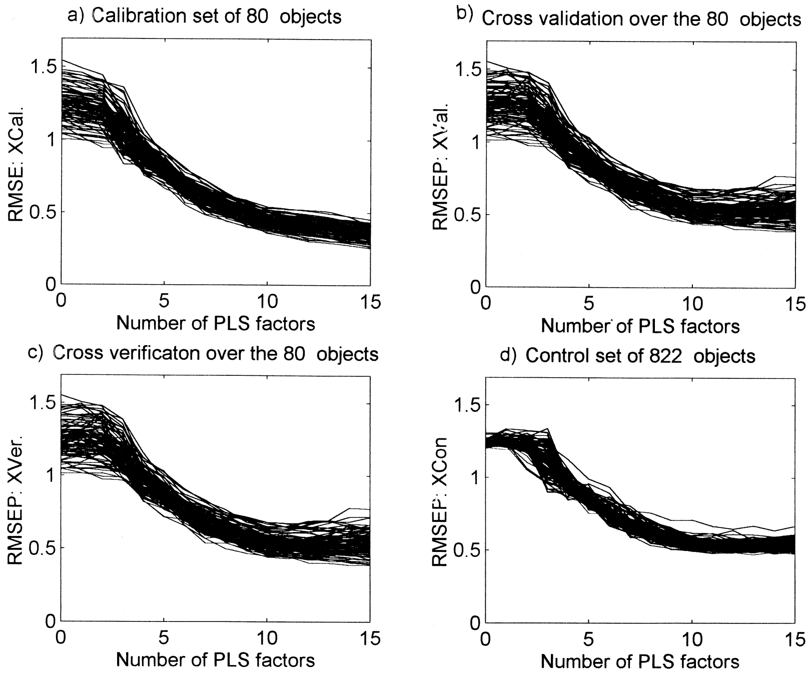

Fig. 8. Cross-validation and cross-verification: Individual replicates. Root Mean Squared Error (RMSE) vs. number of PLS factors *A*, for the same 100 replicates of randomly selecting a set of 80 'available objects' as in Fig. 5.

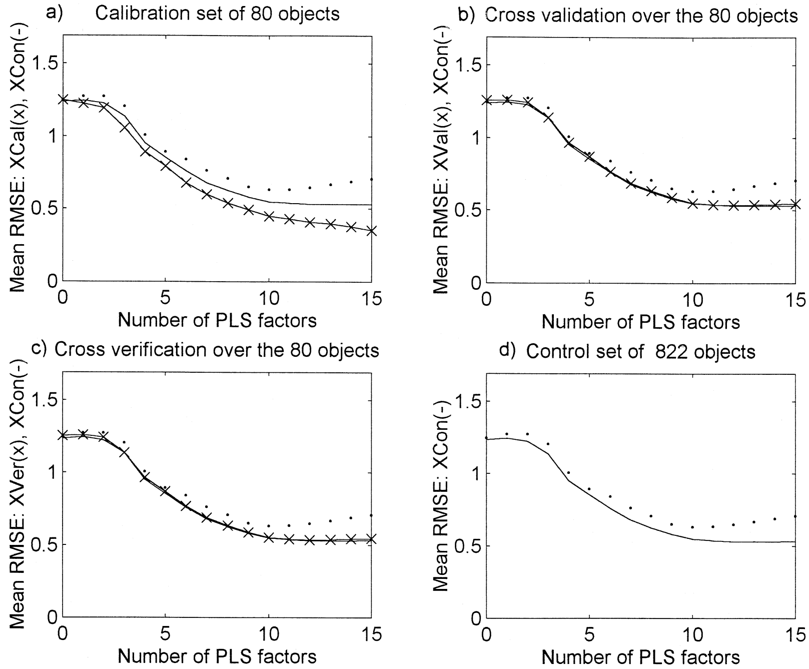

Fig. 9. Cross-validation and verification: Mean over the replications. Root Mean Squared Error (RMSE) vs. number of PLS factors *A*, averaged over the 100 repetitions in Fig. 8. The 'true' control prediction error RMSEP (solid line) from the control set (d) is also included in (a–c). The corresponding 'true' control set RMSEP from Fig. 6d) is included (dotted line) in all four subplots.

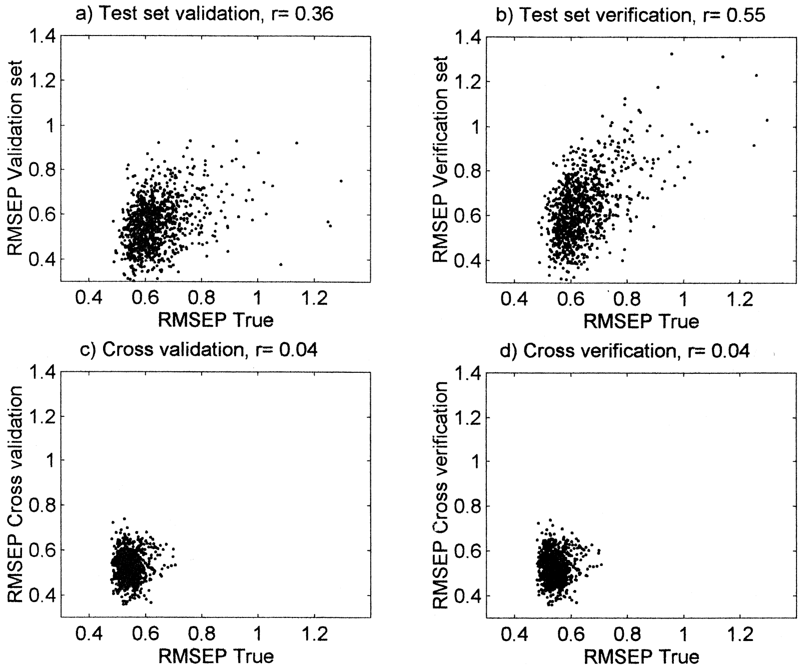

Fig. 10. Comparison of test set and cross-validation and -verifications: Distribution of mean prediction error over 1000 replicate of 80 'available objects'. Upper figures: Test set; Lower figures: Resampling. Left side: Validation RMSEPVal(*A*Opt), Right side: Verification RMSEPVer(*A*Opt). Abscissa: 'True' RMSEP(*A*Opt) from control set.

There is little or no correlation between estimated and 'true' RMSEP in the lower plots(r = 0.04), but there is a clear correlation in the upper plots ($r = 0.36$ and 0.55). The reason for this difference is not yet clear.

Comparison of the upper vs. lower subplots shows that the estimation of *A*Opt by test set validation gives higher and more widely varying final RMSEP than the use of cross validation, both for the 'true' and the estimated values, and both during validation and verification. This is probably due to the difficulty in obtaining precise estimates of squared summaries like a variance or an MSE, and hence of RMSEP in small data sets. This uncertainty in estimating RMSEP from small test sets will now be demonstrated by a simulation using separate, artificial data.

4.5. An estimated uncertainty variance has itself a large uncertainty variance

Fig. 11 shows the estimated cumulative distribution of 10 000 replicate estimates of the final RMSEP in independent verification test sets. The RMSEP values are based on artificial data, and were generated as explained in the Experimental part. The simplifying assumptions are (a) that the true RMSEP is known exactly (= 1), (b) the **X**-data are error-free, and (c) the calibration model has no estimation errors (corresponding to infinitely many calibration objects). The only remaining variation source is independent, identically normal distributed variation $N(0,1)$, simulating, e.g., measurement noise in **y** and true variations in **y** not predictable from **X**. This error will have to be sampled in the test set, and this sampling will be more or less satisfactory, depending on the size of the test set.

The five curves simulate verification test sets of size *n*Test = 5, 10, 20, 50 and 100. It shows that the smaller the test set, the more widely distributed and hence uncertain are estimates of RMSEP in individual test sets. Approximate confidence intervals for the estimated RMSEP around the true RMSEP = 1 may be read from the figure. For instance, when the test set is very small (*n*Test = 5), at the confidence level

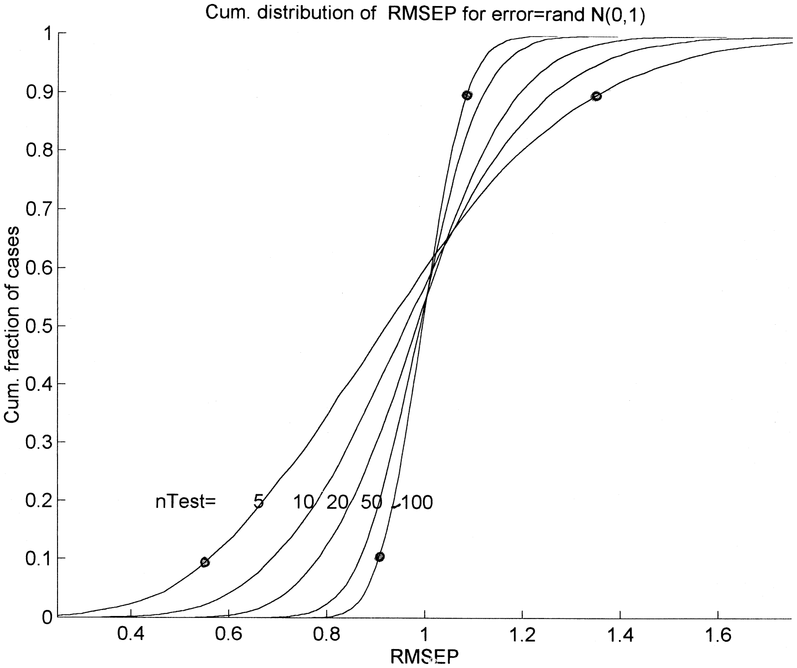

Fig. 11. The uncertainty of an estimated uncertainty. The uncertainty in estimating RMSEP, for a simulated case where the true RMSEP equals 1.0: Abscissa: Estimated RMSEP value; ordinate: Cumulative frequency distribution over 10 000 replications, for test sets of sizes *n*Test = 5, 10, 20, 50 and 100.

as low as 80% (see dots at cumulative fraction of cases = 0.1 and 0.9) the confidence interval is as wide as 0.8 (< 0.5 1.3 >). In contrast, for the largest test set (*n*Test = 100), the 80% confidence interval is only 0.12 (< 0.92 1.1 >).

Hence, part of the large variation found above in RMSEP from the present NIR test sets is probably due to the small size of the test sets themselves, compared to the model complexity.

It is thus important to have enough objects in the test set, when estimating variances and standard deviations (e.g., RMSEP). But in most practical situations, the more objects we put in an independent test set, the less objects we have left for calibration, and the higher will the true predictive error be.

4.6. *The potential for further improvement of the AOpt estimation*

The test set validation/verification was found to perform less satisfactory than the cross validation/verification, and one probable explanation is the difference in number of validation and verification objects. However, could another reason be that the method of estimating *A*Opt (Eqs. (8), (9a), (9b) and (9c)) in the test set validation or the cross validation was suboptimal?

The two plots in Fig. 12 show the potential for improvement in the two rank estimation methods (test set validation and cross validation), for the *M* = 100 replicates. In each figure the abscissa represents the number of PLSR components used, *A*, and the ordinate represents the 'true' RMSEPCon(*A*) in the control set obtained at *A* components Each replicate *m* is represented by two connected points, one for *A* = the estimated *A*Opt(*m*) from the validation, and the other for the *A* that corresponds to the minimum value of the 'true' RMSEPCon itself in replicate *m*. The result at *A*Opt(*m*) is represented by 'o' for test set validation in Fig. 12a, and by 'x' for the cross validation in Fig. 12b

The figure shows that for some replicates *A*Opt was too low and for other replicates it was too high, for both validation method.

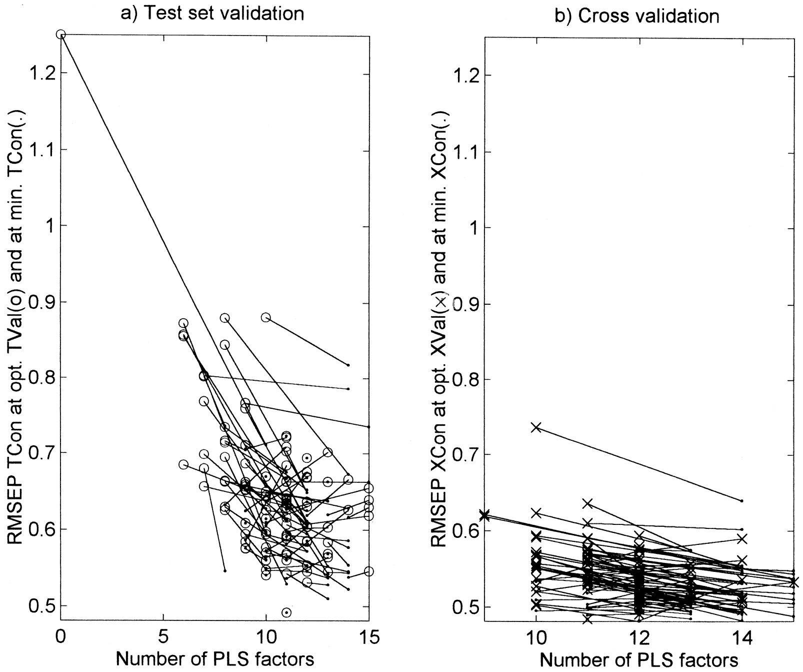

Fig. 12. Potential for further improvement in estimating optimal PLSR model rank *A*Opt. (a) Test set validation: For each of 100 replicates, the final 'true' prediction error in the control set, RMSEPTCon (*A*Opt), with AOpt estimated in validation test set, 'o', is connected to the corresponding RMSEP TCon(*A*OptCon) when *A*OptCon was instead taken at the minimum of RMSEPTCon itself. (b) Cross validation: As for (a), but with prediction error in the control set, RMSEPXCon(*A*Opt) calculated using the calibration model based on all available samples in cross validation, and with *A*Opt 'x' vs. the corresponding minimal RMSEPXCon(*A*OptCon) when *A*OptCon was instead at the minimum of RMSEPXCon itself.

Incidentally, Fig. 12a shows that $A\text{Opt} = 0$ was automatically chosen in one of the 100 replicates, in the case of using test set. Closer inspection of this replicate in Fig. 5c revealed it had very low initial RMSEP in the validation test set. Removing this one abnormal replicate from the average results in Figs. 6 and 7, however, had virtually no effect on the mean final RMSEP and its standard deviation summaries over 100 replicates.

The results for the two validation methods in Fig. 12 are summarized in Table 1. A small underestimation of the average *A*Opt of -0.8 and -0.6 components could be observed for the two validation methods, respectively, with a corresponding deterioration of the 'true' RMSEPCon of about 0.03 percent protein. This systematic underestimation of *A*Opt may be due to the 'punishing' term in Eq. (7) that favours the lower-rank solution among several almost equal solutions. This term was in preliminary simulations found to stabilize the rank estimation and improve predictive ability, however. It is possible that the use of local cross-validation [7] could give somewhat more optimal estimates of *A*Opt (S. Wold, pers. com., 1997), but then probably with much more strongly underestimated cross-validation RMSEP values. It should also be kept in mind that we here assume a homogenous, stationary population of samples.

4.7. Sensitivity to experimental conditions

Table 2 compares the results at the estimated AOpt in Fig. 7, to other experimental conditions. In order to make it easier to see the effects on final RMSEP in the table, the results from one particular 'basis' condition (column 4) have been subtracted from every RMSE result, both w.r.t. rms mean results (0.55 subtracted) and standard deviations (0.03 subtracted).

Test set validation and cross validation results are compared explicitly at the bottom of the table.

Table 1
Control of *A*Opt and the associated 'true' RMSEP

	(a) Test set validation	(b) Cross-validation
Number of cal. objects, *n*	40	80
Number of val. objects, *n*Val	20	80
*A*Opt from validation	10.2 (2.6)	11.3 (1.5)
*A*OptCon from control set	11.0 (1.9)	12.7 (1.7)
*A*Opt difference	−0.8	−0.6
'True' RMSEPCon(*A*Opt)	0.63 (0.10)	0.54 (0.03)
'True' RMSEPCon(*A*OptCon)	0.60 (0.05)	0.52 (0.03)
RMSEPCon difference	0.03	0.03

Model rank *A*Opt from validation is the model rank estimated by (a) test set validation and (b) cross-validation, averaged over $M=100$ replicates (with its standard deviation).
*A*OptCon is the corresponding model rank at the minimum of the 'true' RMSEP from the control set, RMSEPCon.
The actual 'true' mean RMSEP from the control set (Eq. (12)) with the uncertainty standard deviation (Eq. (13)) are given at *A*Opt and *A*OptCon components.
Conditions: 80 available objects, 822 control objects, 70 **X**-variables, 100 Monte Carlo simulation replicates.

4.7.1. Reliability of the Monte Carlo simulation results

Using 100 replicates (column 4) gave almost the same results as 1000 replicates (column 1); hence, the conclusions drawn on 100 replicates are regarded as reliable.

4.7.2. Changing the number of X-variables

Comparison of column 4 with column 2 shows that reducing the number of **X**-variables from 70 wavelength channels (average of 20 nm intervals) to 28 (average of 100 nm intervals) in this case had little or no influence on the results. Apparently, the NIR information was sufficiently well distributed over the spectrum that no loss of rank was experienced, and the multiplexing effect of having 70 **X**-variables in the PLSR model was similar to the noise reduction effect of having averaged over wider wavelength intervals.

4.7.3. Changing the number of available objects, nAvail

Table 2 shows that when number of available objects decreases from 120 (column 3) via 80 (column 4) to 40 (column 5), the 'true' prediction error increases, both for test set and cross validation (RMSEP TCon and XCon). Likewise, the standard deviations of the RMSEP estimates increase, and the estimated *A*Opt value decreases and gets a higher standard deviation.

The damaging effect of splitting the available set of objects thus becomes even more evident when the number of available objects is low: The under-estimation of the 'true' RMSEP increases more with test set validation (RMSEP difference TVal − TCon: −0.05, −0.08, −0.20 percent protein for *n*Avail = 120, 80 and 40, respectively) than with cross validation (RMSEP difference XVal − XCon: −0.02, −0.02, −0.06 percent protein). The mean loss of predictive ability due to splitting the data set increases (RMSEP difference TCon − XCon: 0.06, 0.10 and 0.24% protein).

The full cross verification gave results surprisingly similar to those of the full cross validation under all the conditions tested, the difference being 0.001, 0.001, 0.000, 0.001 and 0.014 percent protein for columns 1–5, respectively. The reason for this is unclear. However, in the present case the good performance of the full cross validation appears to reduce the need for independent verification during cross validation.

Before drawing final conclusions from the results, the following should be noted.

4.8. Critical parameter: limited number of available samples

This study was performed under the conditions where the number of objects with **X**- and **y**-data available is limited. This is in the authors' experience a common situation, although we have exaggerated it somewhat here in order to illustrate our point. Experienced NIR users know that, e.g., 40 samples is too little to allow good calibration in material as complex as whole maize plants, (with cross-validation the 'true' prediction error corresponded to an r^2 of only 0.82 $(=1-(0.68/1.2)^2)$, down from the original 0.94 $(=1-(0.41/1.2)^2)$.

If there had been unlimited available data, then the different validation and verification methods probably would probably give more similar results. If the full leave-one-out cross validation then is considered too time-consuming, a reduced number of cross vali-

Table 2
Results for different estimation conditions

Column	1	2	3	4	5
Experimental conditions					
# Replicates	**1000**	100	100	**100**	100
# **X**-variables	70	**28**	70	**70**	70
Available objects	80	80	**120**	**80**	**40**
Control set objects	822	822	**782**	**822**	**862**
Use of the available objects with test sets					
Calibration	40	40	**60**	**40**	**20**
Validation	20	20	**30**	**20**	**10**
Verification	20	20	**30**	**20**	**10**
Mean model rank (with standard deviation)					
AOpt TVal	10.3 (2.5)	10.3 (2.4)	10.9 (2.3)	10.2 (2.6)	8.1 (3.5)
AOpt XVal	11.3 (1.6)	11.6 (1.2)	11.9 (1.8)	11.3 (1.5)	10.7 (2.1)
Mean prediction errors: final mean RMSE (with standard deviation), deviations from the 'true' results for cross validation, ×100					
RMSEC TCal	−11 (9)	−8 (10)	−11 (5)	−10 (11)	1 (28)
RMSEC XCal	−13 (2)	−12 (2)	−12 (2)	−13 (2)	−15 (7)
RMSEP TVal	2 (8)	3 (7)	−2 (6)	2 (7)	17 (6)
RMSEP XVal	−2 (3)	−1 (3)	−4 (1)	−2 (2)	7 (9)
RMSEP TVer	10 (11)	13 (11)	4 (7)	11 (11)	38 (24)
RMSEP XVer	−2 (3)	−1 (3)	−4 (1)	−2 (2)	7 (9)
RMSEP TCon	10 (6)	12 (7)	3 (2)	10 (6)	37 (14)
RMSEP XCon	**0 (0)**	0 (0)	−3 (−1)	**0 (0)**	13 (6)
RMSEP XCon =	**0.55 (0.03)** (subtracted from each result above)				
Comparison of validation based on test set and cross-validation, differences between the final mean RMSEP, ×100					
RMSEP TVal − TCon	−8	−9	−5	−8	−20
RMSEP XVal − XCon	−2	−1	−2	−2	−6
RMSEP TCon − XCon	10	12	6	10	24

Comparison of various mean RMSEP(*A*Opt) results from Eq. (12), (with uncertainty standard deviations, Eq. (13)) in simulations: Different number of replicates (column 1 vs. column 4), number of **X**-variables (column 2 vs. column 4) and different number of available objects (column 3 vs. column 4 vs. column 5).
'T' and 'X' represent the use of test set validation and cross-validation, respectively.
'Cal', 'Val', 'Ver' and 'Con' represent calibration, validation, verification and 'true' control set results.

dations based on independent, randomly selected cross validation sets of objects, will probably be just as good. On the other hand, with that many available samples, overfitting is not a problem, so even RMSEC may be expected to be a good estimator for RMSEP.

4.9. The need for a stationary population and representative sample selection

The assumption that the training set data are *representative* for the **X–y** relationships in a certain large *population* of objects is critical. Multivariate calibration gives best results as an interpolation method, not an extrapolation method, especially with randomly selected objects and a forward regression method like PLSR [1]. This means that all **X–y** co-variation types in the population of future objects is supposed to be spanned representatively by the available objects, and that the noise level in **X** is supposed to remain the same (**X** and **y** may be said to reflect a 'stationary process').

If, e.g., in industrial process control, the process calibrated for is non-stationary—if new **X–y** co-variation types arise over time, or the **X**-measurements become more noisy, then the data available at one point in time cannot be representative for the process at a later point in time.

In such cases the use of independent verification test sets at later points in time is recommended, in order to check and correct for drift. However, these verification test sets should not be too small (cf. Fig. 11). Also, this control against non-stationarity over time should not be confused with the concept of taking out independent, more or less randomly selected, validation and verification test sets from the available data set during the actual calibration.

4.10. Comparison to previous work

4.10.1. Monte Carlo simulations

The data in the previous study [7] were generated from a Beer's Law model, with one analyte and two interferants, each with randomly chosen spectra and concentrations, with 20 available objects and 3000 control objects in 1000 replicates. Among other things, various noise levels in **X** and **y** were tested, as well as various numbers of **X**-variables (5 or 20) and the use of one vs. two independent test sets. The results found with the artificial data were very similar to the present ones found with the real NIR and Kjeldahl protein data.

However, there is still a need for further evaluation, using independently programmed software.

4.10.2. Cross verification methodology

Hardy et al. [6] found their double-case cross-validation to give unacceptable predictors in some cases were conventional cross-validation appeared to perform well. This was not found with the present study. There may be several reasons for the difference.

It may in part be that a different criterion for determining the optimal number of components was used. Hardy et al. rightly rejected the *overall minimum* of the Prediction Error Sum of Squares (PRESS, which is proportional to cross-validation RMSEP-Val^2), as criterion. Instead they defined AOpt as the rank of the *first minimum* in the PRESS. In our experience this can be quite misleading; instead, the modification in Eqs. (8), (9a), (9b) and (9c) was used for stabilization against incidentally choosing AOpt too high.

A difference between our results and those of Hardy et al. may also be in the way they implemented their cross-verification, using a different rank $A\text{Opt}(i)$ for each different calibration object $i = 1,2,\ldots,n\text{Cal}$: For each object in the main cross validation, the authors performed a full leave-one-out cross validation on the remaining $n\text{Cal} - 1$ objects to determine $A\text{Opt}(i)$ for this object i, which was then predicted, using the obtained $A\text{Opt}(i)$-dimensional regression model for this cross validation repetition.

Another difference between Hardy et al.'s results and the present results may be in the amount of input data and how they have been analyzed: Hardy et al. used three small data sets and had no control data. We have averaged 100 or 1000 Monte Carlo replicates of small data sets, each time checking the 'true' performance with a control set of more than 800 real objects. The Monte Carlo simulation with artificial data [7] was even more extensive. The present conclusions should therefore be statistically more reliable.

4.10.3. Further improvement of the full cross validation

The full leave-one-out cross validation should be further investigated in order to develop a correction component for the slight under-estimation of the true apparent prediction error.

It has been suggested (Kim Esbensen, Lars Nørgaard, pers. com., 1997) to use *reduced* cross-validation as a way to counter-act the slight over-optimism of the full cross-validation. Reduced cross-validation could be to take out, e.g., 10 objects, instead of one, in each cross-validation segment.

This means intentionally to leave *some available* systematic variability types in the data as unmodelled 'noise' in RMSEPVal; they should presumably simulate *other*, *non-available* systematic variability types in the population. In other words, the distribution of variability types that incidentally are represented by only a few objects in the available data set are taken as simulating the distribution of variability types in the population that are not properly represented in the available data set. This may or may not in some cases be valid. Little is yet known about the *distribution of distributions* of interferences in various kinds of real experimental data; it is probably a fairly deep problem.

5. Conclusions

For the data types and calibration conditions tested here and in the previous study [7], the following can be concluded.

(1) Independent *validation test set* was wasteful and uncertain, and gave over-optimistic estimates of future predictive error.

When setting aside some of the available objects as independent validation test, the ‘true’ prediction error on the average was much *higher*, and showed greater *variability*, than when all the available objects were used for calibration and full cross-validation used for estimating *A*Opt.

To make things worse, test set validation strongly *underestimated* the resulting true apparent prediction error.

(2) Independent *verification test set* gave, on the average, realistic, but uncertain estimates of the predictive error.

Setting aside yet some of the available objects as independent verification test set gave, on the average, *correct estimates of the level* of true apparent prediction error. But the individual estimates had particularly high *variability*. And the *level of this error was unnecessarily high*, compared to that of full cross-validation/cross-verification.

(3) Full *cross-validation* and *cross-verification* gave very similar results.

The cross-verification gave only slightly, although consistently higher RMSEP estimates than the cross-validation.

(4) Even cross-validation gave a *slight* over-optimism.

Full leave-one-out cross-validation and -verification on the average gave a *slight under-estimation* of the true apparent prediction error.

(5) Both *real* and *simulated* data supported the *theoretically expected* conclusions.

The above conclusions are supported by two very different types of data sets, and are close to what was expected theoretically prior to the analyses.

Acknowledgements

Magni Martens, Rasmus Bro and Lars Nörgaard at the Royal Veterinary and Agricultural University, Copenhagen, Agnar Höskuldsson at Denmark Technical University, Kay Steen and Martin Höy at the Norwegian University of Science and Technology, and Tormod Naes at the Norwegian Food Research Institute are thanked for stimulating discussions during the preparation of this study. Two anonymous referees are acknowledged for their sobering, but constructive criticism, and Claus Andersson of the Royal Veterinary and Agricultural University, Copenhagen is thanked for final corrections of the manuscript. Denmark Technical University is acknowledged for financial support of the study.

References

[1] H. Martens, T. Naes, Multivariate Calibration, Wiley, Chichester, UK, 1989.

[2] Unscrambler for Windows, User’s Guide, CAMO AS, Trondheim, Norway, 1996.

[3] A. Høskuldsson, Prediction methods in Science and Technology, Basic Theory, Vol. 1, Thor Publishing, Copenhagen, Denmark, ISBN 87-984941-0-9, 1996.

[4] M. Stone, J. R. Stat. Soc. 36 (1974) 133.

[5] B. Efron, R.J. Tibshirani, An Introduction to the Bootstrap, Chapman & Hall, New York, ISBN 0-412-04231-1, 1993.

[6] A.J. Hardy, P. MacLaurin, S.J. Haswell, S. De Jong, B.G.M. Vandeginste, Double-case diagnostics for outliers identificaction, Chemometrics and Intelligent Laboratory Systems 34 (1996) 117–129.

[7] H. Martens, Determining rank and evaluating performance in regression: independent test sets or cross-validation/cross-evaluation? in: A. Høskuldsson, L. Nørgaard (Eds.), Proceedings, Symp. Applied Chemometrics, DTU, DK-2800 Lyngby, Denmark, January 29, 1997, ISBN 87-98-5941-2-5, 1997, pp. 15–44.

[8] G. Sinnaeve, P. Dardenne, R. Agneessens, A choice for NIR calibrations in analysis of forage quality, J. Near Infrared Spectrosc. 2 (1994) 163–175.

[9] MATLAB 386 User’s Guide, The Mathworks, South Natick, MA, 1989.