Evaluation of empirical models for calibration and classification

Kurt VARMUZA

Vienna University of Technology

Institute of Chemical Engineering

and

Department of Statistics and Probability Theory

www.lcm.tuwien.ac.at, kvarmuza@email.tuwien.ac.at

Collaboration: Peter Filzmoser and Bettina Liebmann

2nd Summer School 2012 of the Marie Curie ITN "Environmental ChemoInformatics" and Meeting of the International Academy of Mathematical Chemistry,
11 - 15 June 2012, Verona, Italy

Lecture 14 June 2012

Contents

Acknowledgment. This work was supported by the Austrian Science Fund (FWF), project P22029-N13, "Information measures to characterize networks", project leader M. Dehmer (UMIT, Hall in Tyrol, Austria).

Common situation in science

available data

$x_1, x_2, \ldots, x_m$

= vector $\boldsymbol{x}^T$

Measured or calculated x_j

desired data

y *(e.g., property)*

Cannot be determined directly or only with high cost

model

$$\hat{y} = f(x_1, x_2, \ldots, x_m) = f(\boldsymbol{x}^T)$$

f: mathematical equation or algorithm,

derived from data (**empirical model**) or from knowledge (theoretical model)

Common situation in science 1/3

available data

$x_1, x_2, \ldots, x_m$

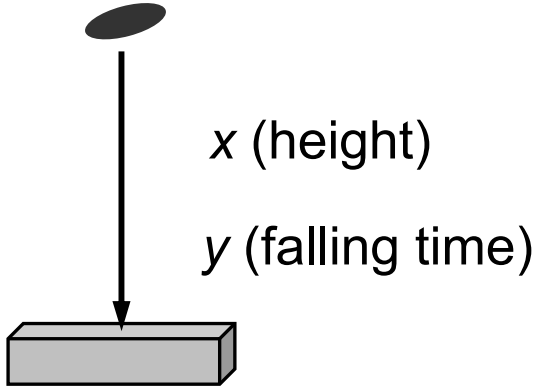
= vector $\boldsymbol{x}^T$

Measured or calculated x_j

desired data

y *(e.g., property)*

Cannot be determined directly or only with high cost



Fundamental (scientific) law,
first principle

$\hat{y} = (2x / g)^{0.5}$

model parameter g: gravity constant

Common situation in science 2/3

available data

$x_1, x_2, \ldots, x_m$
= vector $\boldsymbol{x}^T$
Measured or calculated x_j

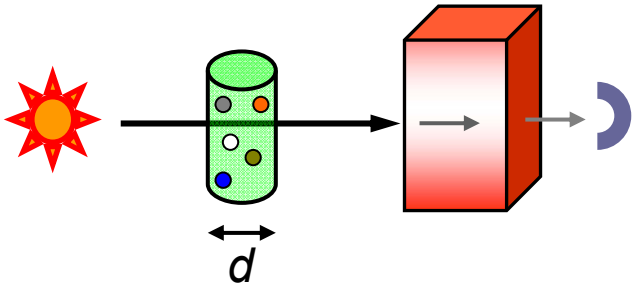
desired data

y *(e.g., property)*

Cannot be determined directly or only with high cost

$x_1, x_2, \ldots, x_m$ (N)IR absorbances

y concentration of a **compound**



Lambert-Beer's law,
reasonable relationship between x and y (parameter unknown)

$$\hat{y} = \log (I_0 / I) / (\alpha d)$$

$$\hat{y} = \sum_j \beta_j \log({}_j I_0 / I_j I) + \beta_0$$

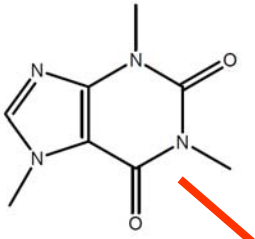
Common situation in science 3/3

available data

$x_1, x_2, \ldots, x_m$

= vector $\boldsymbol{x}^T$

Measured or calculated x_j



set of numbers, molecular descriptors, $\boldsymbol{x}^T$

property y

desired data

y *(e.g., property)*

Cannot be determined directly or only with high cost

Only an assumption:

y (property) is simply related with x (variables),

"very empirical" ("dangerous")

$\hat{y} = \boldsymbol{x}^T \boldsymbol{b} + b_0$ (linear model)

Empirical linear models

$\hat{y} = f(x_1, x_2, ..., x_m)$

y	● continuous	multivariate calibration
	● discrete, categorical	multivariate classification pattern recognition

Linear model

$$\hat{y} = \boldsymbol{x}^T \boldsymbol{b} + b_0 = b_1 x_1 + b_2 x_2 + ... + b_m x_m + b_0$$

b_0: intercept

$\boldsymbol{b}$: vector with regression coefficients

$\boldsymbol{x}$: vector with variables (features, descriptors)

$\hat{y}$: calculated (predicted) property

Empirical linear models

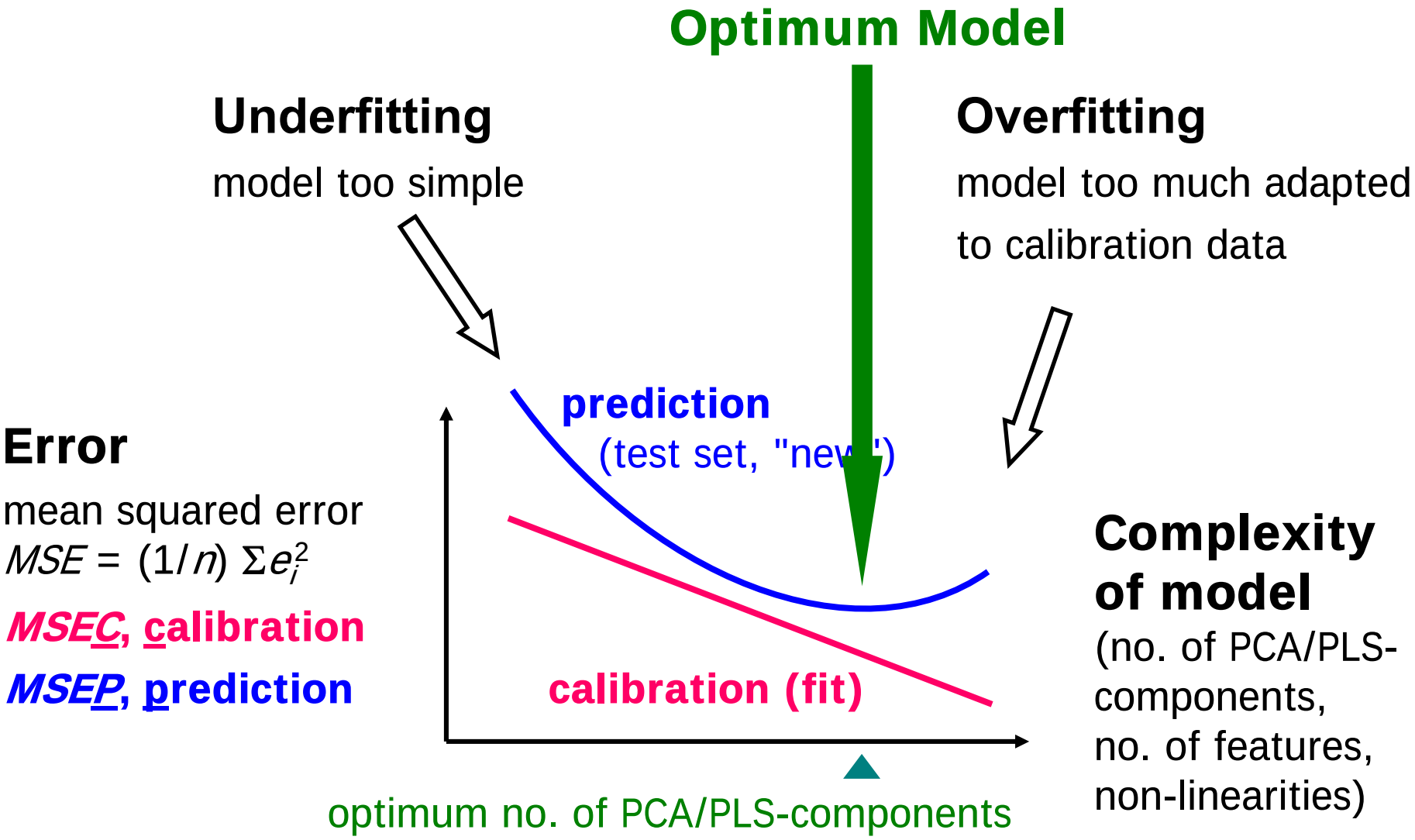
Creation of a model:
estimation of the
model parameters ($\boldsymbol{b}$, b_0)
from given data, $\boldsymbol{X}$ and $\boldsymbol{y}$
(calibration set)

Guiding principle

NOT best fit of the calibration data is important,

BUT **optimum prediction for new cases**
(**test set** data, never used in model creation)

Empirical models



Empirical models

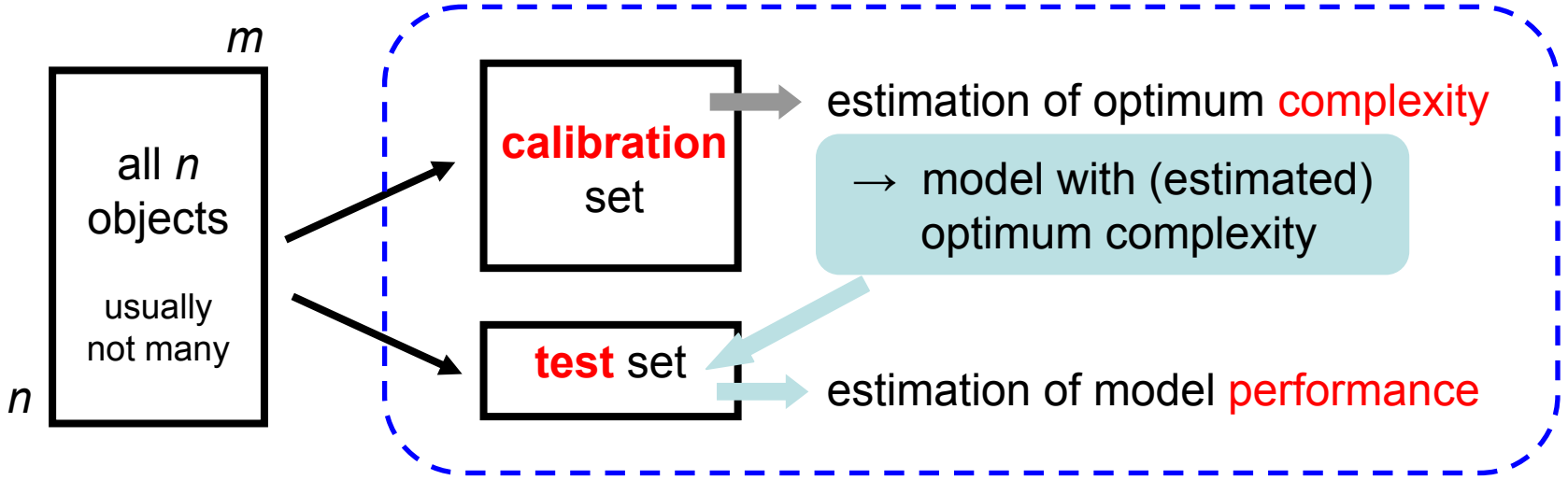
1. Optimum **model complexity**
   estimated from **calibration data**

2. **Performance of model**
   estimated from **test data**

☞ Should be estimated **independently**

Empirical models

Proposed strategy



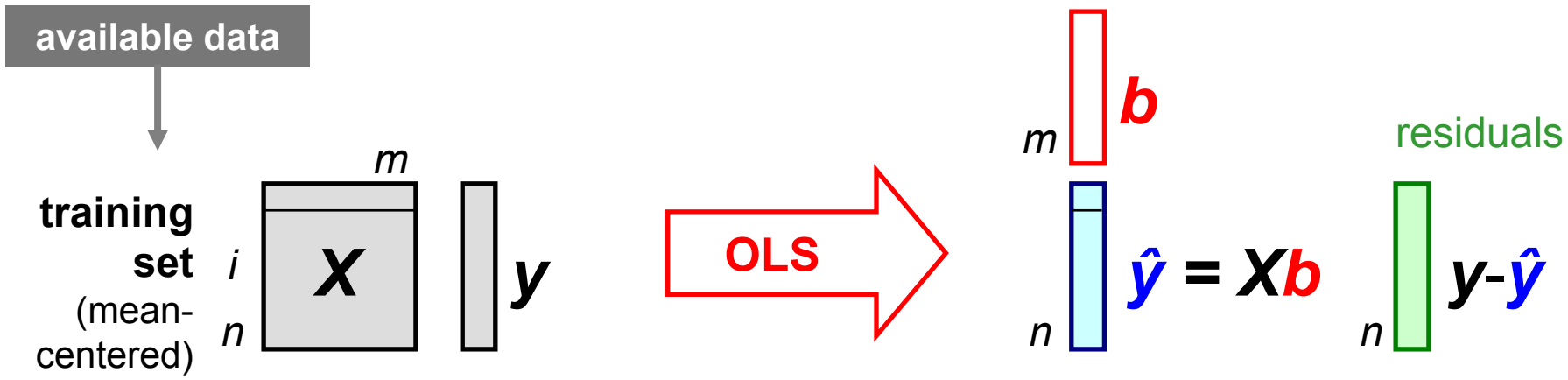
Repeated ! → **variability** of optimum complexity

→ **variability** of performance

Contents

Multivariate calibration (linear)

only a few selected topics



OLS Ordinary Least-Squares Regression

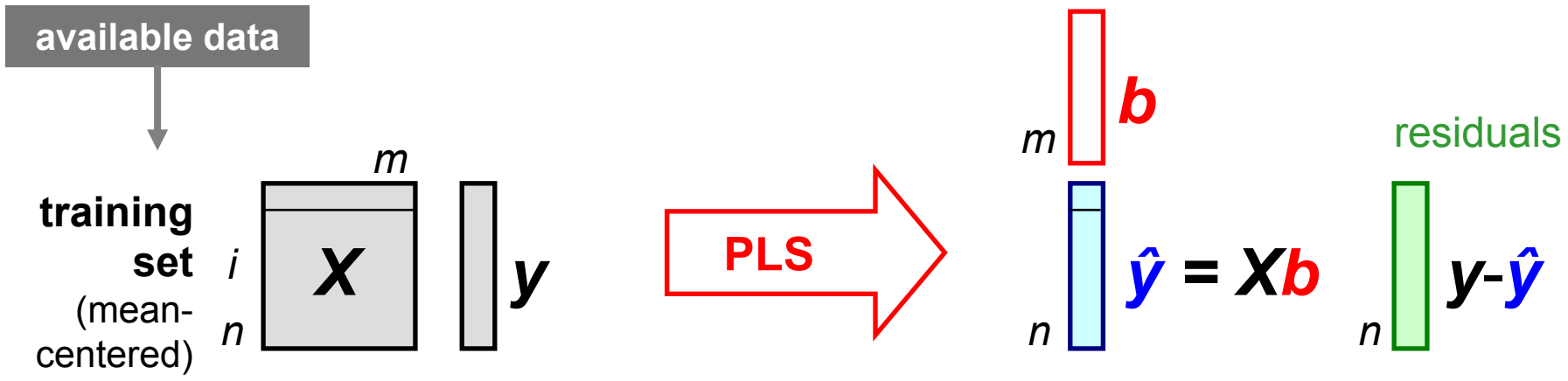
$\sum_i (y_i - \hat{y}_i)^2 \rightarrow \min$ $\qquad \boldsymbol{b}_{OLS} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$

Requirements
- $m < n$
- no highly correlating *x*-variables (columns)

☹ No optimization of model complexity (possibly a variable selection); rarely applicable in chemistry

Multivariate calibration (linear)

only a few selected topics



PLS Partial Least-Squares Regression

simplified

(1) $U_{PLS} = X B_{PLS}$

Intermediate, linear (latent) variables (**components**):
- ☐ maximum covariance with *y*,
- ☐ uncorrelated or orthogonal directions in *x*-space,
- ☐ number of PLS-components is optimized

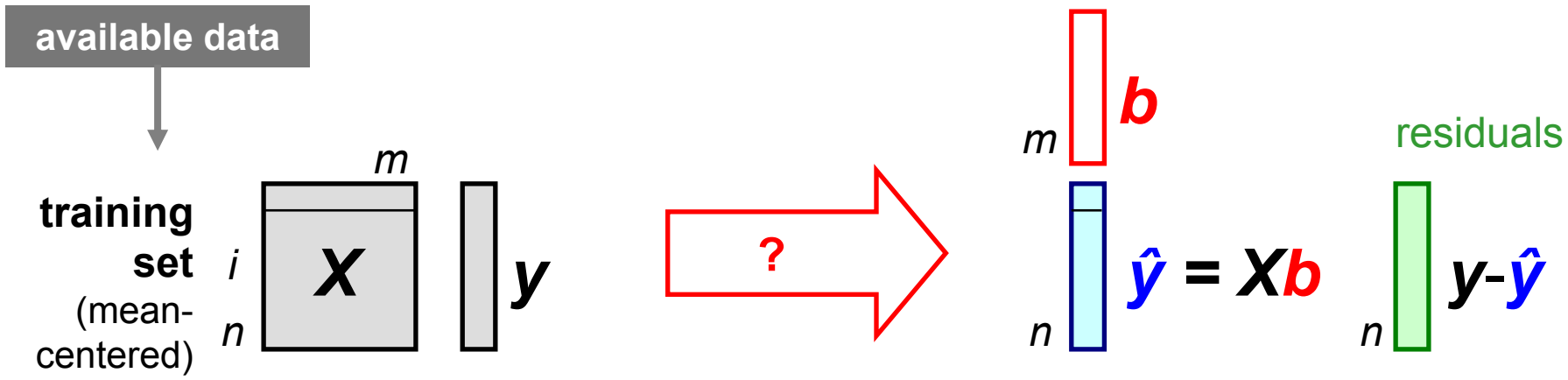
(2) OLS with U_{PLS}

☺
- ☐ applicable if $m > n$,
- ☐ applicable for highly correlating variables,
- ☐ optimization of model complexity !!!

☹ Various different approaches and algorithms

Multivariate calibration

only a few selected topics



Some other regression methods in chemometrics

PCR Principal Component Regression (similar to PLS)

Lasso includes variable selection

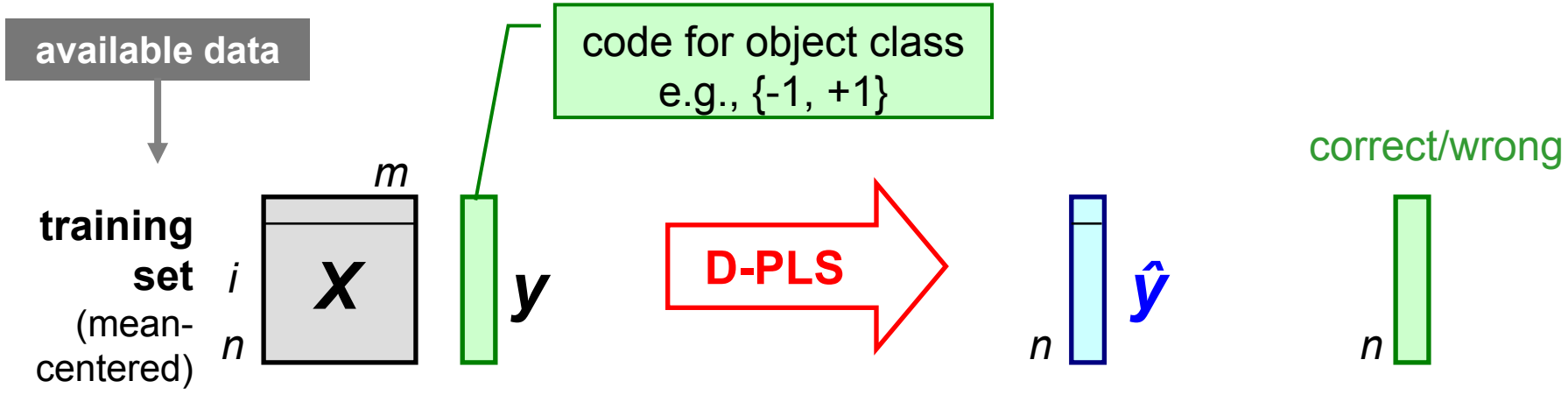
Ridge similar to PCR (weighting of all PCA scores)

ANN Artificial Neural Networks (nonlinear)

PLS2 PLS for more than one *y*-variable

Multivariate classification (linear)

only a few selected topics



D-PLS Discriminant PLS *

Binary classification (2 classes): y = -1 and +1 for class 1 and 2, resp.

PLS is used as a regression method (resulting in a discriminant vector $\boldsymbol{b}$).
Optimization of model complexity: number of PLS-components.

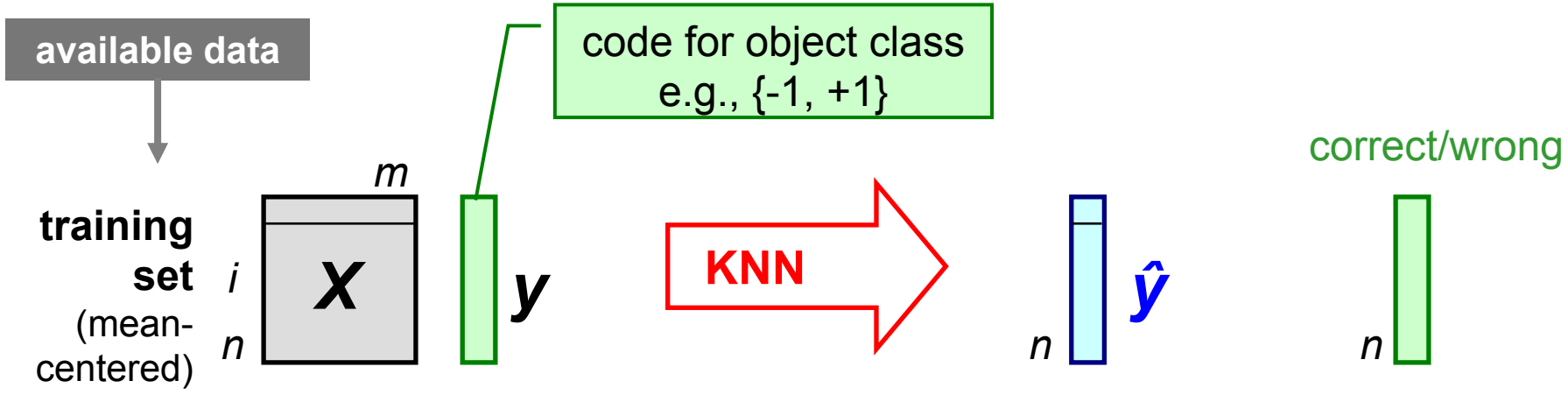
Class assignment: $\hat{y} = \boldsymbol{x}^T\boldsymbol{b}$; if $\hat{y} < 0$ assign to class 1, else to class 2.

Often used instead of LDA (linear discriminant analysis, equivalent to OLS)
because of the advantages of PLS

* D-PLS is in general not recommended for >2 classes

Multivariate classification

only a few selected topics



KNN (*k*-nearest neighbor) classification

An algorithm; nonlinear; no discriminant vector.

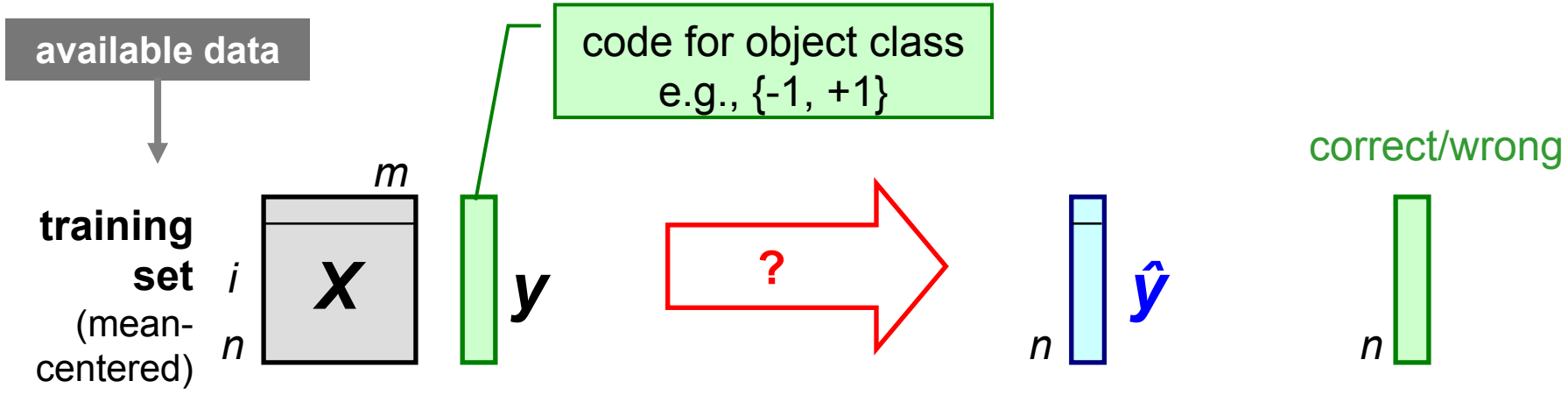
Usually the **Euclidean distance** between objects (in *x*-space) is used to find the nearest neighbors (objects with known class membership) to a query object.

A majority voting among the neighbors determines the class of the query object.

Optimization of model complexity: *k* (number of neighbors)

Multivariate classification

only a few selected topics



Some other classification methods in chemometrics

SVM Support Vector Machine (nonlinear)

CART Classification tree (nonlinear, evident)

SIMCA PCA models for each class (nonlinear, outlier detection)

ANN Artificial Neural Networks (nonlinear)

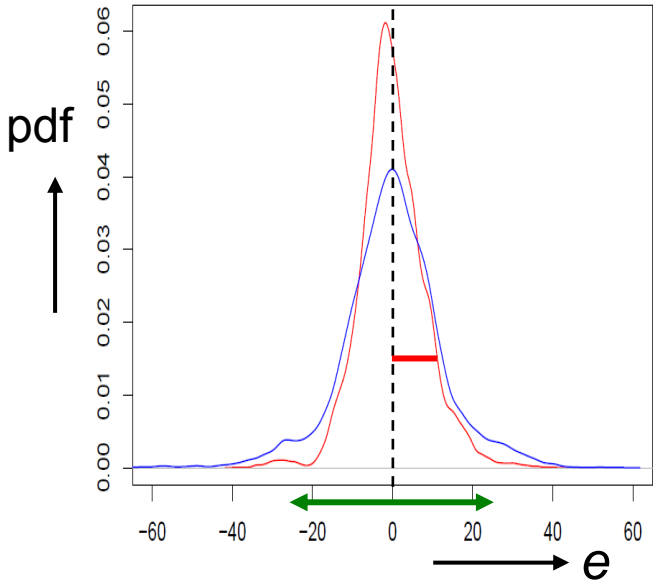
Contents

Performance measures in calibration

y_i	reference ("true") value for object *i*
$\hat{y}_i$	calculated (predicted) value (**test set !**)
$e_i = y_i - \hat{y}_i$	prediction error for object *i* (residual)
$i = 1 \ldots z$	*z* is the number of objects used (*z*>*n* possible)

Specify: ☞ which data set (calibration set, test set)
☞ which strategy (cross validation, ...)

Distribution of prediction errors



bias = mean of prediction errors e_i

– SEP = standard deviation of prediction errors e_i
= **S**tandard **E**rror of **P**rediction

SEC = **S**tandard **E**rror of **C**alibration

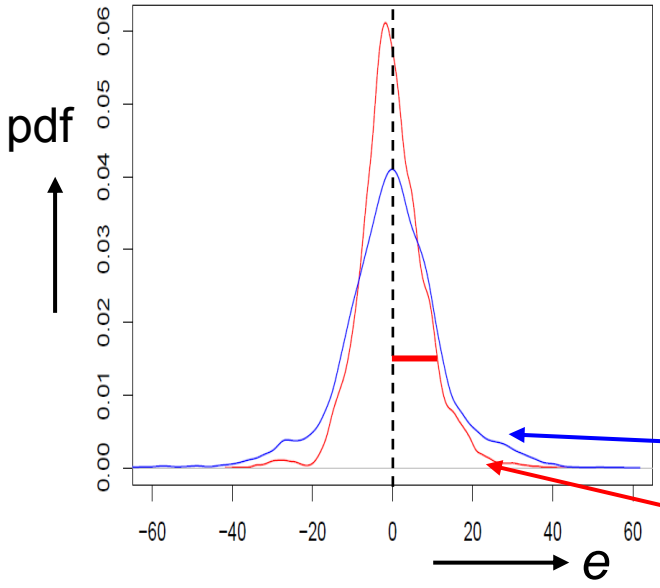
CI = confidence interval, $CI_{95\%} \approx \pm 2*SEP$

All in units of *y* ! Result: $\hat{y} \pm 2*SEP$

Performance measures in calibration

y_i	reference ("true") value for object *i*
$\hat{y}_i$	calculated (predicted) value (**test set !**)
$e_i = y_i - \hat{y}_i$	prediction error for object *i* (residual)
$i = 1 \ldots z$	*z* is the number of objects used (*z*>*n* possible)
	Specify: ☞ which data set (calibration set, test set)
	☞ which strategy (cross validation, ...)

Distribution of prediction errors



Modeling the GC retention index (*y*) for *n* = 208 PAC by *m* = 467 molecular descriptors (*Dragon* software)

Repeated double cross validation (rdCV) with 100 repetitions (*z* = 20 800)

m = 467; SEP = 12.7

m = 13; SEP = 8.2

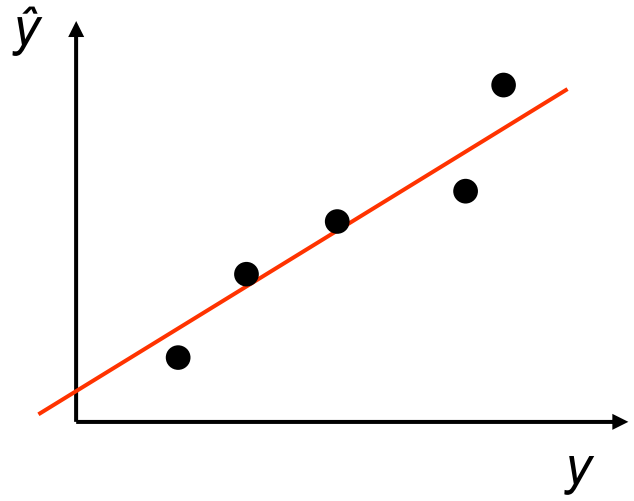
test set objects

Performance measures in calibration

y_i	reference ("true") value for object *i*
$\hat{y}_i$	calculated (predicted) value (**test set !**)
$e_i = y_i - \hat{y}_i$	prediction error for object *i* (residual)
$i = 1 \ldots z$	*z* is the number of objects used (*z*>*n* possible)

Specify: ☞ which data set (calibration set, test set)
☞ which strategy (cross validation, ...)

Predicted versus reference *y*'s



R^2 = **squared (Pearson) correlation coefficient**

$$_{ADJ}R^2 = 1 - (n-1)(1-R^2) / (n-m-1)$$

squared adjusted correlation coefficient

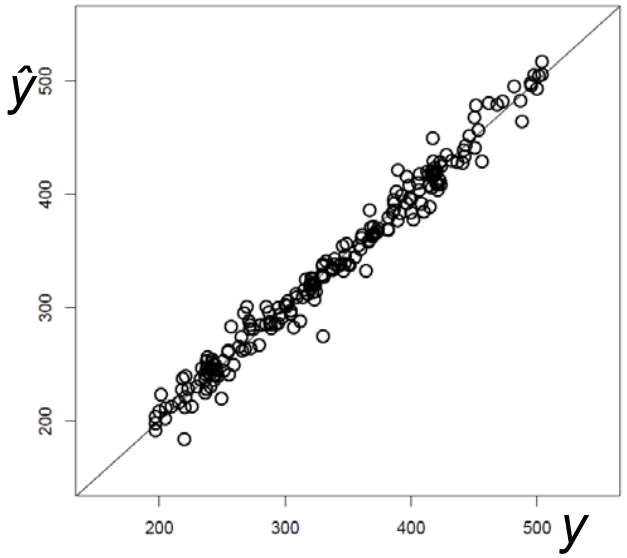
Penalizes models with a higher number of variables (*m*)

Performance measures in calibration

y_i	reference ("true") value for object *i*
$\hat{y}_i$	calculated (predicted) value (**test set !**)
$e_i = y_i - \hat{y}_i$	prediction error for object *i* (residual)
$i = 1 \ldots z$	*z* is the number of objects used (*z*>*n* possible)

Specify: ☞ which data set (calibration set, test set)
☞ which strategy (cross validation, ...)

Predicted versus reference *y*'s



Modeling the GC retention index (*y*) for *n* = 208 PAC by *m* = 467 molecular descriptors (*Dragon* software)

Repeated double cross validation (rdCV)
$\hat{y}$ are means of 100 repetitions

R^2 = 0.979 (test set objects)

Various other diagnostic plots.

Performance measures in calibration

y_i	reference ("true") value for object *i*
$\hat{y}_i$	calculated (predicted) value (**test set !**)
$e_i = y_i - \hat{y}_i$	prediction error for object *i* (residual)
$i = 1 \ldots z$	*z* is the number of objects used (*z*>*n* possible)
	Specify: ☞ which data set (calibration set, test set)
	☞ which strategy (cross validation, ...)

Some other measures

(R)MSE	(root) mean squared error	= (root of) mean of prediction errors e_i
PRESS	predicted residual error sum of squares	= sum of squared errors e_i
Q^2	correlation measure for external test set objects	
AIC	Akaike's information criterion	consider *m*
BIC	Bayes information criterion	
C_p	Mallow's C_p	

Performance measures in classification

Class assignment table (binary classification)

no. of objects

	assigned class 1	2	sum
true class 1	n_{11}	n_{12}	n_1
true class 2	n_{21}	n_{22}	n_2
sum	$n{\to}1$	$n{\to}2$	n

Predictive ability class 1 $P_1 = n_{11}/n_1$

class 2 $P_2 = n_{22}/n_2$

Average predictive ability $P = (P_1 + P_2)/2$

! Avoid: Overall predictive ability $= (n_{11} + n_{22})/n$

Performance measures in classification

Example (warning)

n = 100; n_1 = 95; n_2 = 5

E. g.: All objects from class 1 are correctly classified;
all objects from class 2 are wrong classified.
Result: P_1 = 1; P_2 = 0; P = 0.5 (a bad classifier, OK)

However, $P_{OVERALL}$ = 0.95 ("high for a very bad classifier")

Predictive ability **class 1** $P_1 = n_{11}/n_1$

class 2 $P_2 = n_{22}/n_2$

Average predictive ability $P = (P_1 + P_2)/2$

! **Avoid: Overall predictive ability** $= (n_{11} + n_{22})/n$

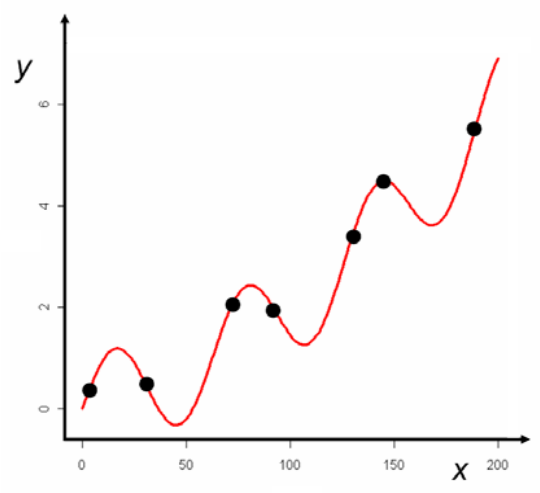
Performance measures in classification

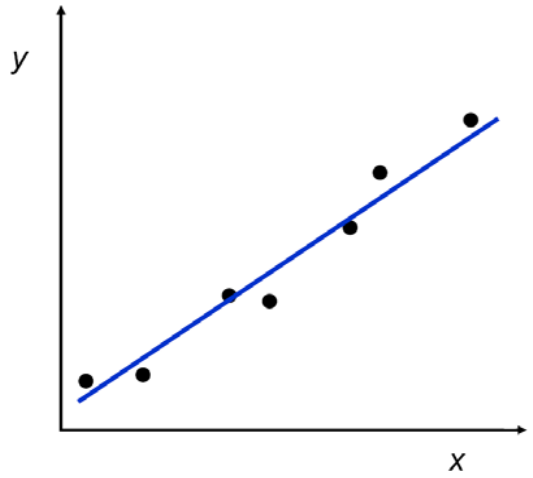
Other measures for classification performance

misclassification rate	for each class separately and summarized
risk of wrong classification	different risks for wrong classification of the different classes can be defined
rejection rate	if no assignment to any class is allowed (dead zone)
confidence of answers	ratio of correct answers (assignment to a specific class 1, 2, 3, ...); depends on ration n_1/n_2 like overall predictive ability

Contents

Strategies (1) Optimum model complexity





fit error = 0

too high complexity of model

overfitted

error for new cases probably large

fit error > 0

perhaps better (optimum) complexity of model

perhaps optimal fitted

error for new cases probably smaller than for overfitted model

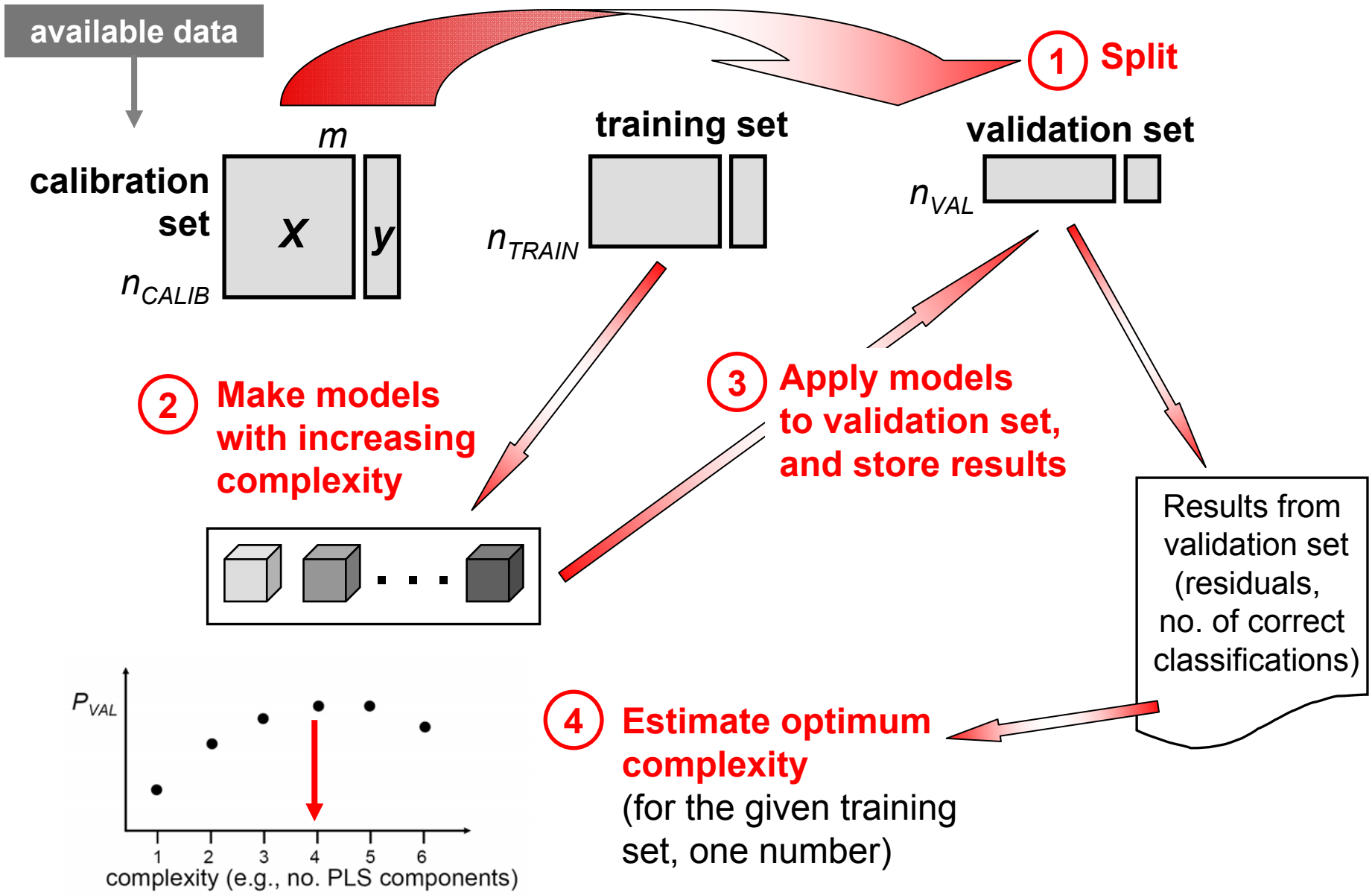
Strategies (1) Optimum model complexity

Optimum complexity of model has to estimated by trial and error.
Usually not a unique solution.

Optimum complexity: parameter of the method for model generation

Calibration		Classification	
PLS	no. of PLS components	**DPLS**	no. of PLS components
PCR	no. of PCA components	**PCA + LDA**	no. of PCA components
Ridge	complexity parameter λ_R	**KNN**	no. of neighbors
Lasso	complexity parameter λ_L	**SVM**	gamma
ANN	no. of hidden neurons	**SIMCA**	no.s of PCA components
OLS	(no. of variables)	**CART**	tree size
		ANN	no. of hidden neurons

Strategies (1) Optimum model complexity



Strategies (1) Optimum model complexity

Optimum model complexity: estimation, statistics

- ❑ more data are better,
- ❑ more estimations are better

However, usual data sets (in chemistry) are small (number of objects, n = 20 ... 200)

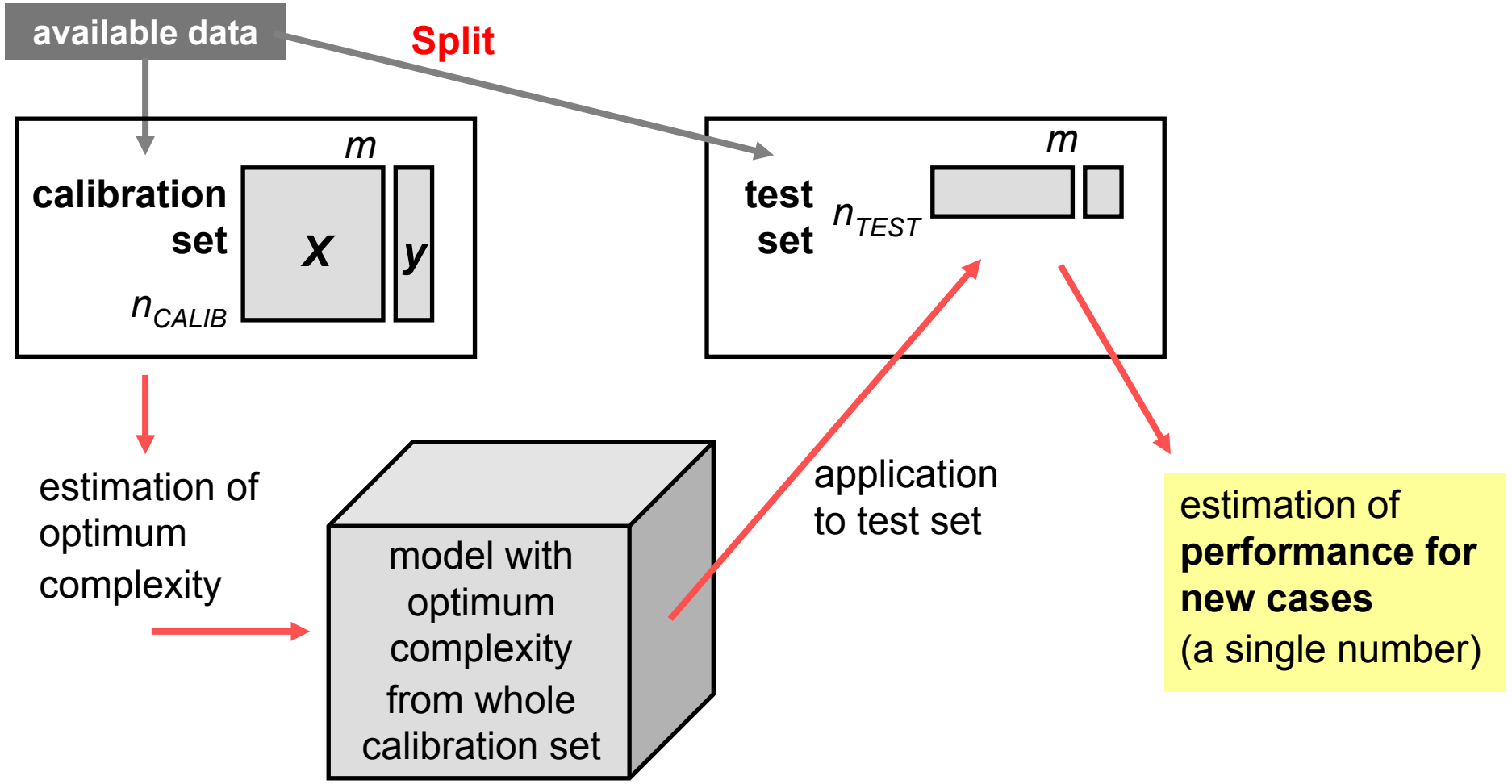
Resampling strategies

- ◆ bootstrap
- ◆ cross validation (CV)

within a calibration set for estimation of optimum model complexity (but not for estimation of model performance)

! Several estimations of the optimum complexity (distribution) !

Strategies (2) Performance for new cases



! Depends on (random) split into calibration set and test set !

Contents

repeated double Cross Validation (rdCV)

For calibration

Filzmoser P., Liebmann B., Varmuza K.: *J. Chemom.*, **23**, 160 (2009).
Repeated double cross validation.

Similar (*cross model validation and permutation*)

Westerhuis J.A. et al.: *Metabolomics*, **4**, 81 (2008).
Assessment of PLSDA cross validation.

Applications of rdCV

- Liebmann B., Friedl A., Varmuza K.: *Anal. Chim. Acta*, **642**, 171 (2009).
  Determination of **glucose and ethanol in bioethanol** production by near infrared spectroscopy and chemometrics.
- Felkel Y., Dörr N., Glatz F., Varmuza K.: *Chemom. Intell. Lab. Syst.*, **101**, 14 (2010).
  Determination of the **total acid number** (TAN) of used gas **engine oils** by **IR** and chemometrics applying a combined strategy for variable selection.
- Liebmann B., Filzmoser P., Varmuza K.: *J. Chemom.* **24**, 111 (2010). **Robust and classical PLS** regression compared.

R-package *chemometrics*; see also **www.lcm.tuwien.ac.at/R**

repeated double Cross Validation (rdCV) - scheme

repetition loop: n_{REP} (20 - 100) times with different random splits into calibration and test set

double CV with all n objects

outer CV loop

CV splits into calibration set + test set (s_{TEST} segments)

inner CV loop with the calibration set

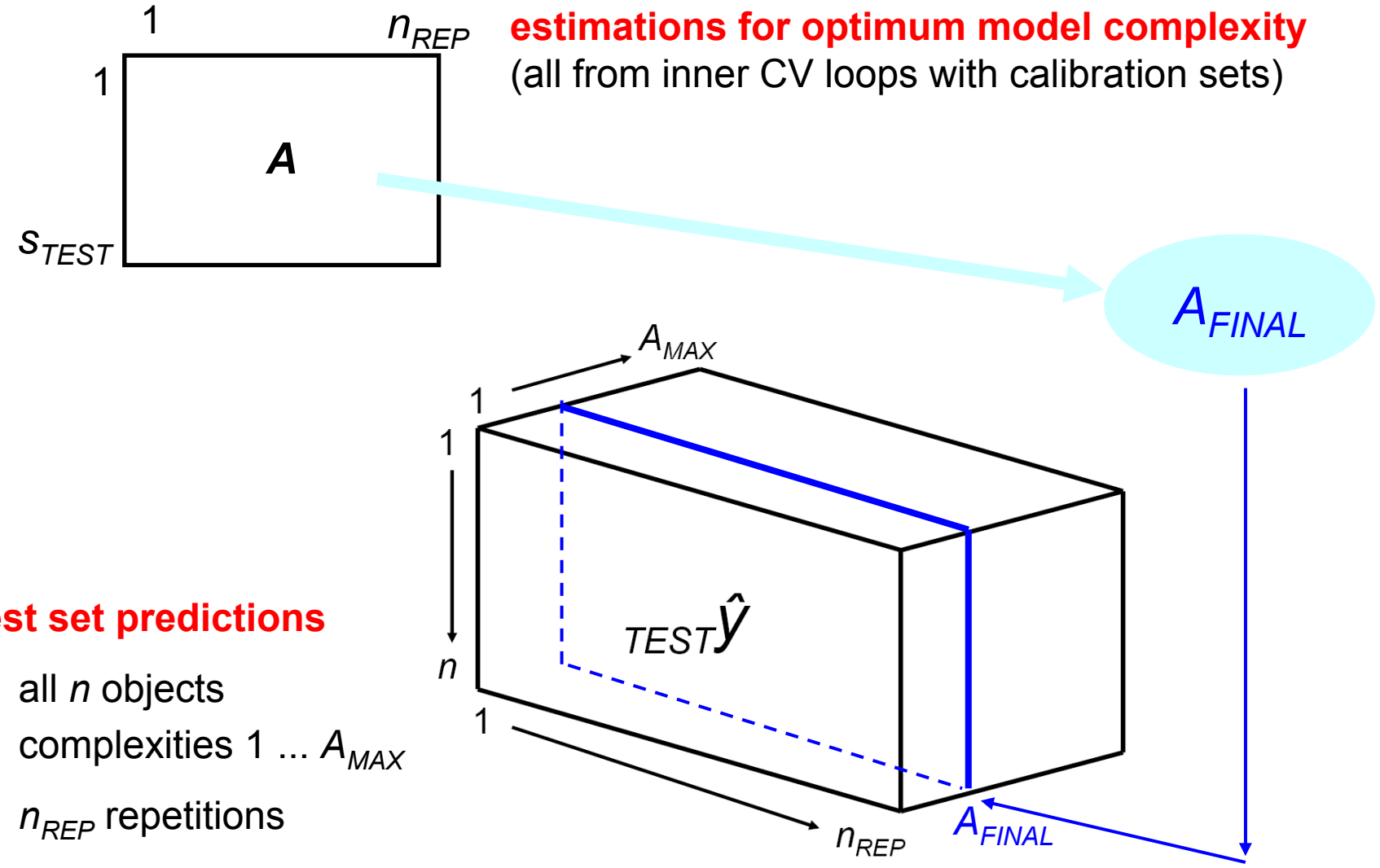
CV splits into training and validations sets (s_{CALIB} segments)

- one estimation of optimum complexity
- $_{TEST}\hat{y}$ for the current test set objects

  (for one of s_{TEST} segments, for all complexities)

- $_{TEST}\hat{y}$ for all n objects (for all complexities)
- s_{TEST} estimations of the optimization criterion

repeated double Cross Validation (rdCV) - **results**



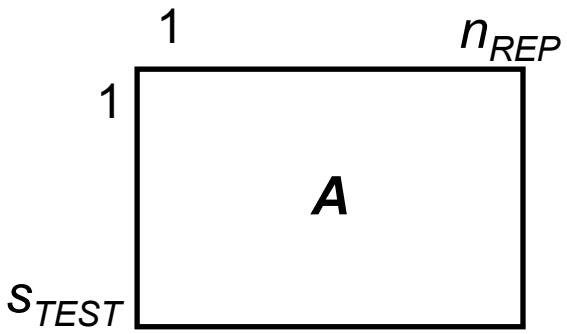
estimations for optimum model complexity
(all from inner CV loops with calibration sets)

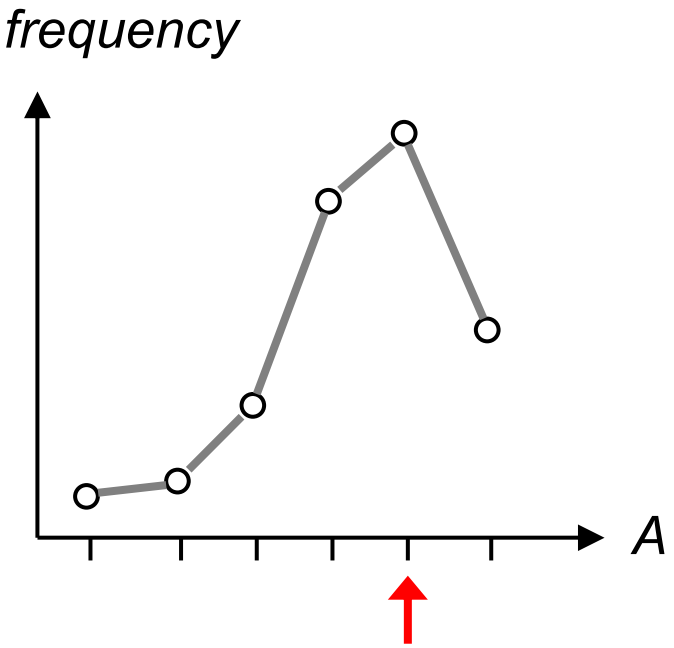
test set predictions

- all *n* objects
- complexities 1 ... A_{MAX}
- n_{REP} repetitions

repeated double Cross Validation (rdCV) - **results**

s_{TEST} * n_{REP} values for optimization parameter, *A*



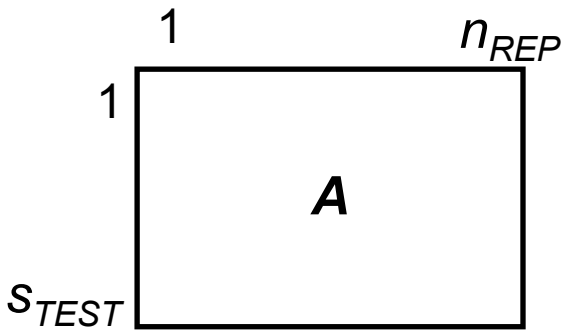


Typical, e. g.,
s_{TEST} = 4
n_{REP} = 50
give 200 estimations for the optimum complexity

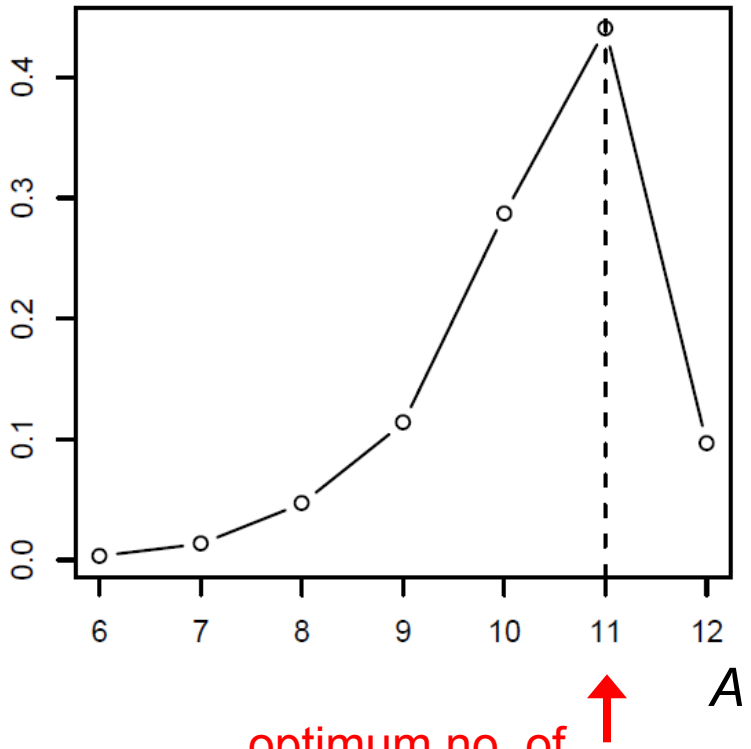
- Most frequent value as A_{FINAL}
- Or other heuristics,
  or a set of values for A_{FINAL}
  (⟶ consensus model)

repeated double Cross Validation (rdCV) - **results**

s_{TEST} * n_{REP} values for optimization parameter, *A*



frequency (300 values)

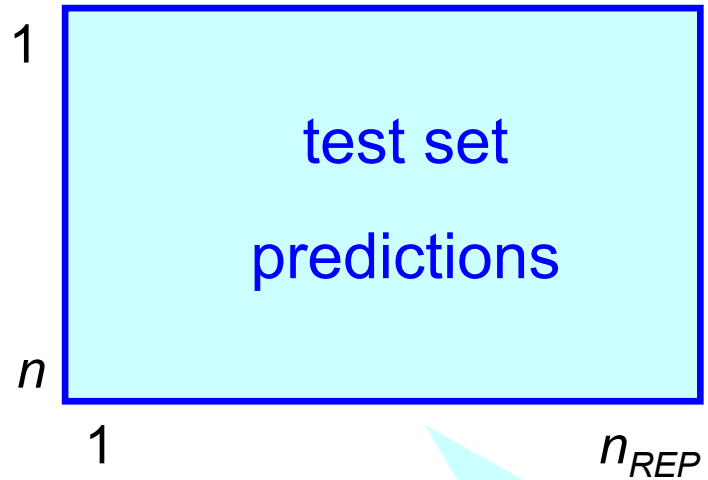


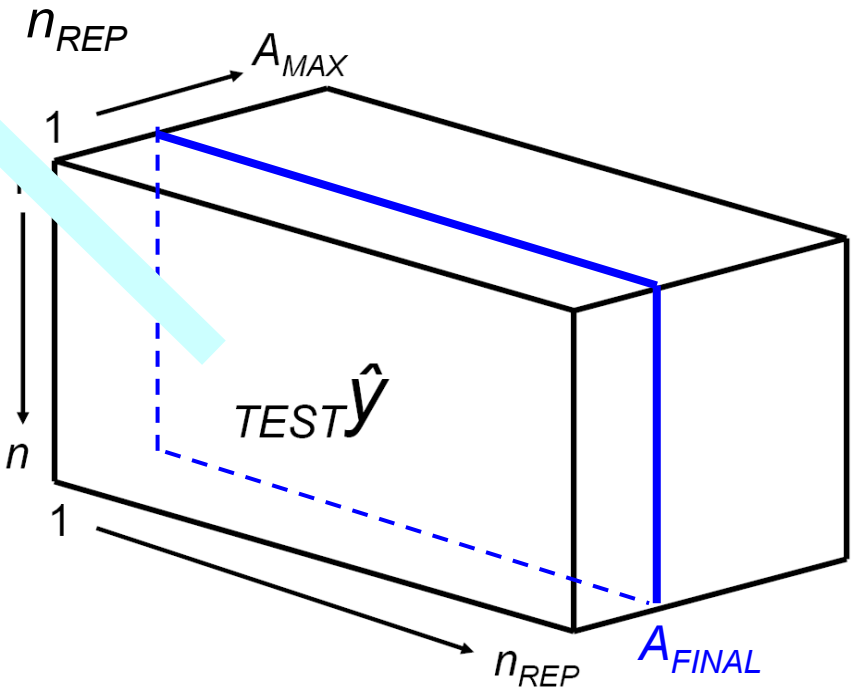
optimum no. of PLS components; A_{FINAL} = 11

Modeling the GC retention index (*y*) for *n* = 208 PAC by *m* = 467 molecular descriptors (*Dragon* software).

rdCV with s_{TEST} = 3 segments in outer loop, and n_{REP} = 100 repetitions

repeated double Cross Validation (rdCV) - **results**

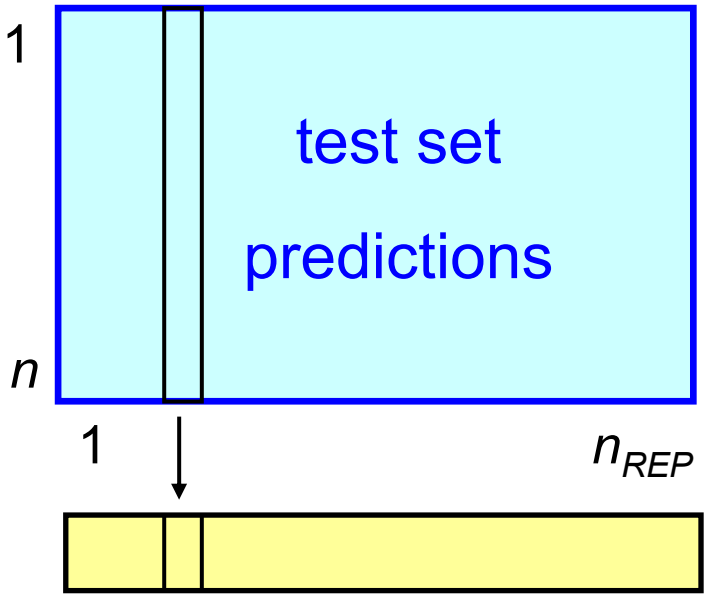




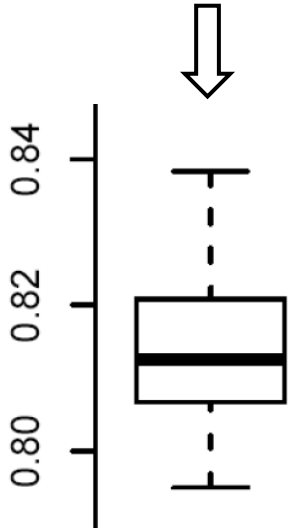
test set predictions

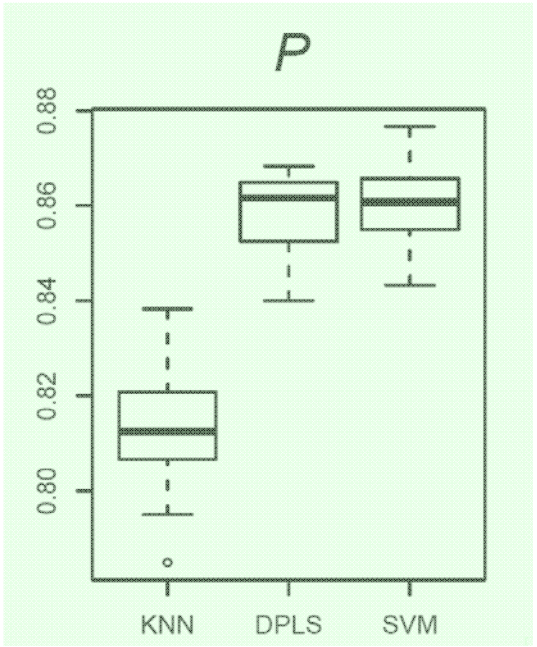
- all *n* objects
- complexities 1 ... A_{MAX}
- n_{REP} repetitions

repeated double Cross Validation (rdCV) - **results**



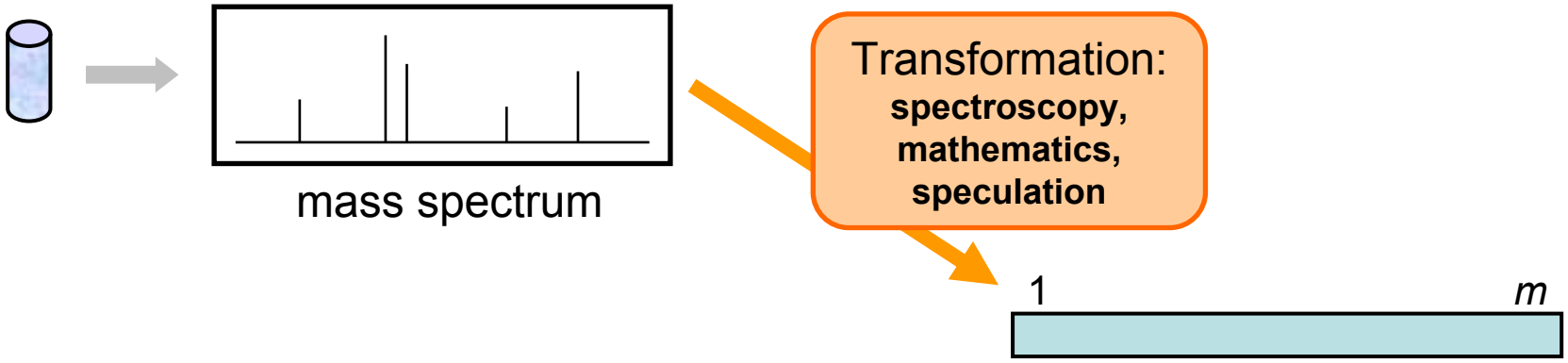
SEP, R^2, P, ...
for the repetitions





repeated double Cross Validation (rdCV) - **example**

Spectra-structure relationship (KNN, DPLS, SVM)



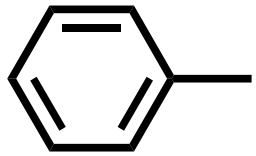
Binary classification

Chemical substructure present / not present (class 1 / class 2)

n = 600 (class 1: 300; class 2: 300), *m* = 658

Dataset '*phenyl*' in R-package '*chemometrics*'

C_6H_5 -



Werther W., Demuth W., Krueger F.R., Kissel J., Schmid E.R., Varmuza K.: *J. Chemom.*, **16**, 99 (2002)

Varmuza K., Filzmoser P.: Introduction to multivariate statistical analysis in chemometrics. CRC Press, Boca Raton, FL, USA (2009)

repeated double Cross Validation (rdCV) - **example**

Spectra-structure relationship (KNN, DPLS, SVM)

rdCV

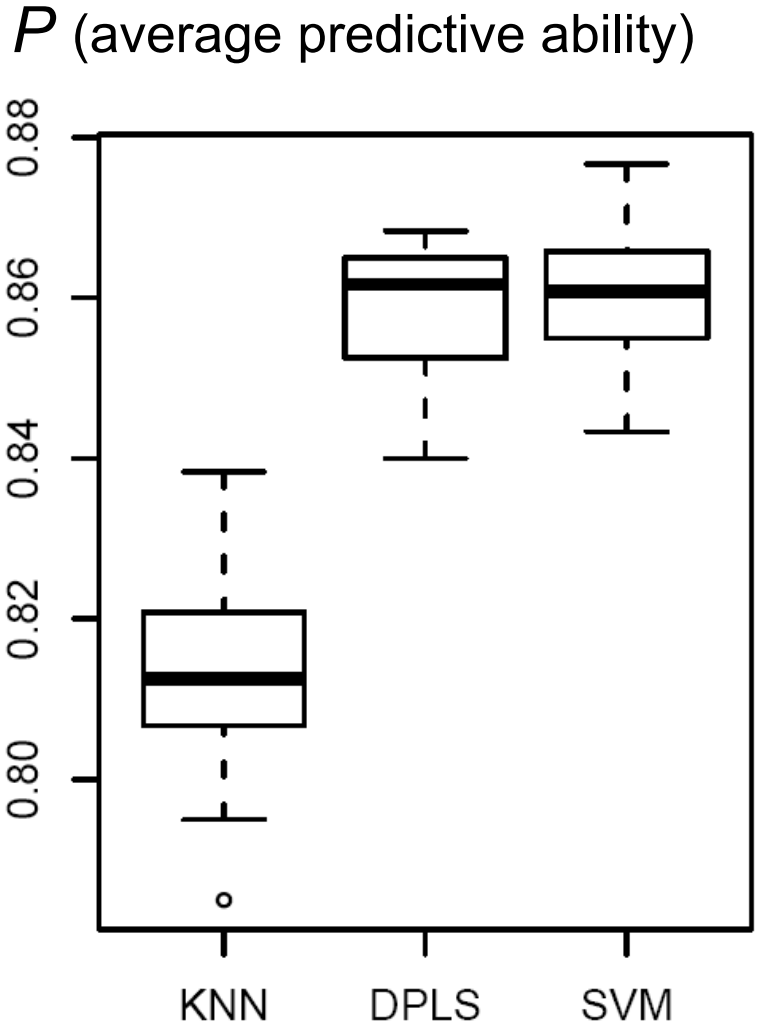
20 repetitions;
$s_{OUT} = 2$; $s_{IN} = 6$

Optimized parameter

KNN: $k_{FINAL} = 3$
DPLS: $a_{FINAL} = 2$
SVM: $\gamma_{FINAL} = 0.0002$

Computation time

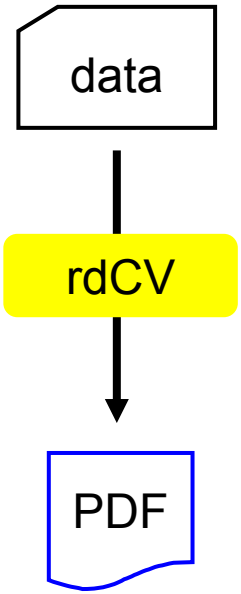
KNN	550 s
DPLS	42 s
SVM	940 s



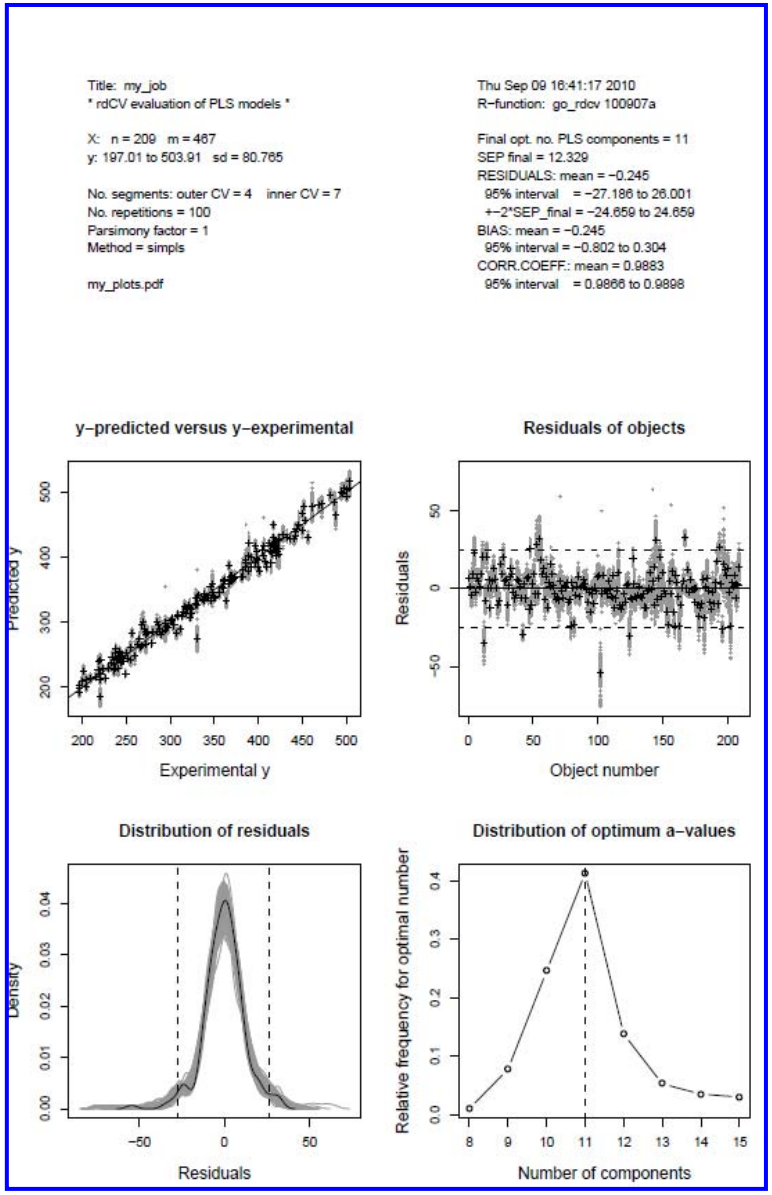
repeated double Cross Validation (rdCV) - summary

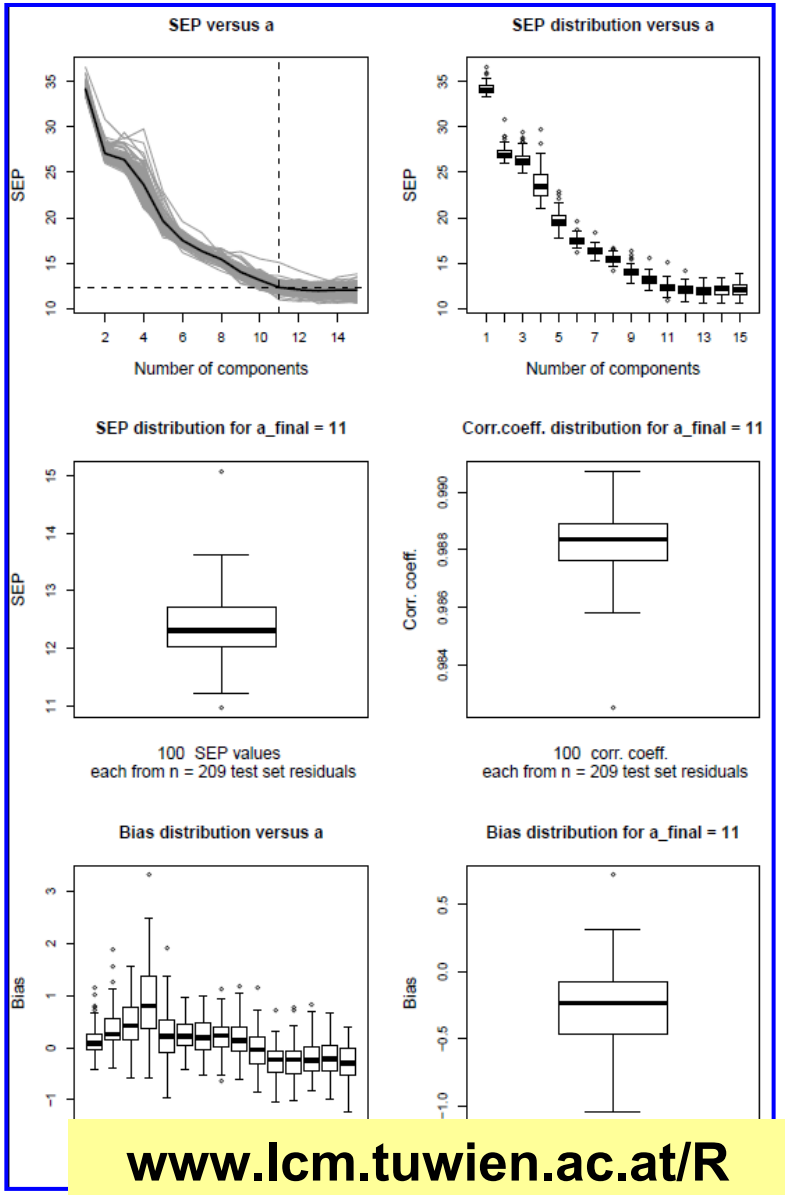
- **A resampling method combining some systematics and randomness.**
- **For calibration and classification.**
- **For data sets with ca ≥ 25 objects.**
- **Optimization of model complexity (model parameter) is separated from the estimation of model performance.**
- **Provides estimations of the variability of model complexity and of performance.**
- **Easily applicable and fast**
  - **R-package "*chemometrics*"**
  - **www.lcm.tuwien.ac.at/R**

repeated double Cross Validation (rdCV) - software



GC-retention indices of 206 PACs



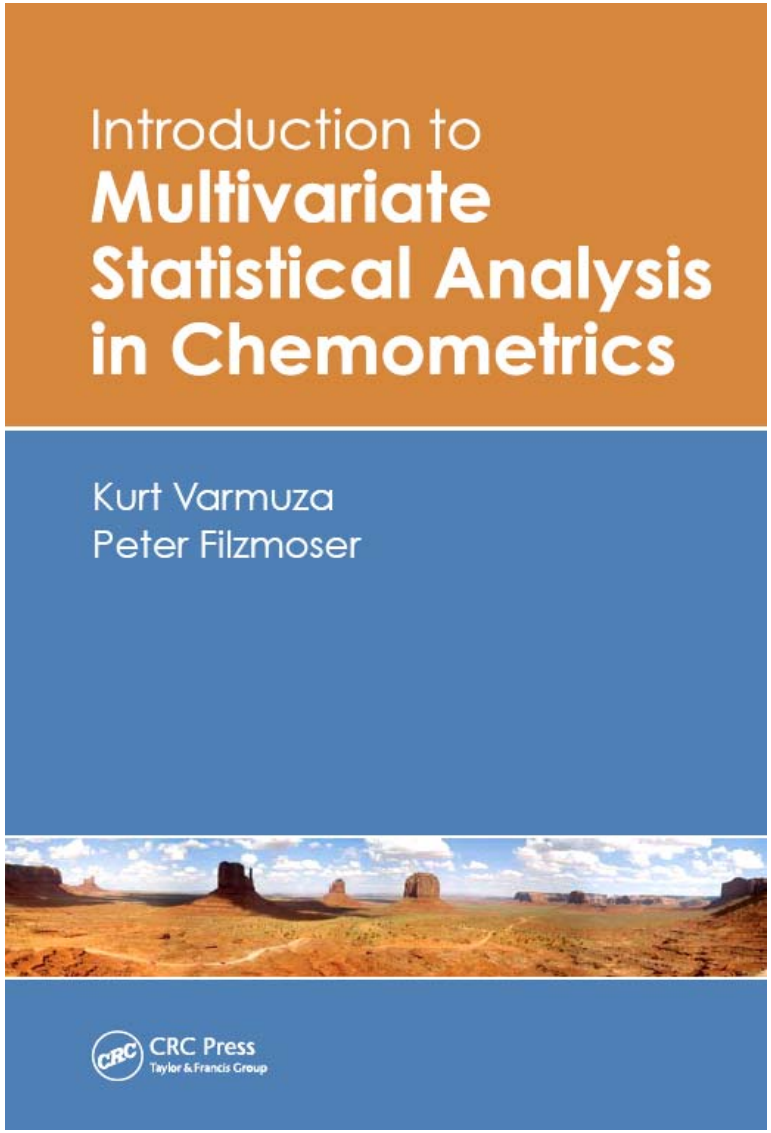


www.lcm.tuwien.ac.at/R

Contents

- ☞ **Take time and effort for validation**
- ☞ **Consider variability**
- ☞ **Accept variability and uncertainty**

Book including examples and data sets for R



CRC Press, Taylor & Francis Group,
Boca Raton, FL, USA, 2009
ISBN: 9781420059472

Ca 320 pages,
appr. € 100

Includes many R-codes (examples)

However, description of methods without R

R package *chemometrics*

Info: www.lcm.tuwien.ac.at