Åbo Akademi University

Basics of Multivariate Modelling and Data Analysis

Kurt-Erik Häggblom

6. Principal component analysis (PCA)

6.1 Overview

6.2 Essentials of PCA

6.3 Numerical calculation of PCs

6.4 Effects of data preprocessing

6.5 Evaluation and diagnostics

[mostly from Varmuza and Filzmoser (2009) and PLS-toolbox manual by Wise et al. (2006)]

6.1 Overview

Today's laboratory instruments produce large amounts of data.

- It is not uncommon for processes to have hundreds or thousands of measured variables.
- Some analytical instruments measure tens of thousands of variables; in a typical FTIR spectrometer, the absorbance is measured at over 10,000 frequencies.
- Chemical processes are becoming more heavily instrumented and the data are recorded more frequently.

Generally, there is a great deal of correlated or redundant information in laboratory and process measurements.

- This information must be compressed in a manner that retains the essential information and is more easily displayed than each of the variables individually.
- Essential information often lies not in individual variables but in how they change with respect to one another, i.e. how the variables interact.
- In the presence of large amounts of noise, it would be desirable to take advantage of some sort of signal averaging.

Principal component analysis (PCA) can be considered as "the mother of all methods in multivariate data analysis."

- The aim of PCA is dimension reduction, which may be used for
  - visualization of multivariate data by scatter plots
  - transformation of highly correlating x-variables into a smaller set of uncorrelated latent variables that can be used by other methods
  - separation of relevant information (by a few latent variables) from noise
  - combination of several variables that characterize a chemical-technological process into a single or a few "characteristic" variables
- PCA can be seen as a method to compute a new orthogonal coordinate system formed by latent variables (components), where only the most informative dimensions are used.
- Latent variables from PCA optimally represent the distances between objects in the high-dimensional variable space — the distance of objects is a measure of the similarity of the objects.
- PCA is successful for data sets with correlating variables as is often the case with data from chemistry.
- PCA is a method for "exploratory data analysis" (unsupervised learning).

6.2 Essentials of PCA

In PCA, we are dealing only with the data matrix $\mathbf{X}$, there is no vector or matrix of "dependent variables". We use the following type of notation:

- x_j (with one subscript) denotes a variable j, but not the value of the variable
- x_{ij} (with two subscripts) denotes the value of variable j for observation i (i.e. object or measurement at time instant i)
- $\mathbf{x}_j$ (i.e. a vector in boldface with one subscript) denotes all values of variable j (i.e. for all observations i)

Consider a linear combination of all variables x_j, $j = 1, \ldots, p$. If every object (or measurement) i, $i = 1, \ldots, n$, is considered, we get the vector equation

$$\mathbf{t}_1 = \mathbf{x}_1 p_{11} + \mathbf{x}_2 p_{21} + \cdots + \mathbf{x}_p p_{p1} = \mathbf{X} \cdot \mathbf{p}_1$$

where

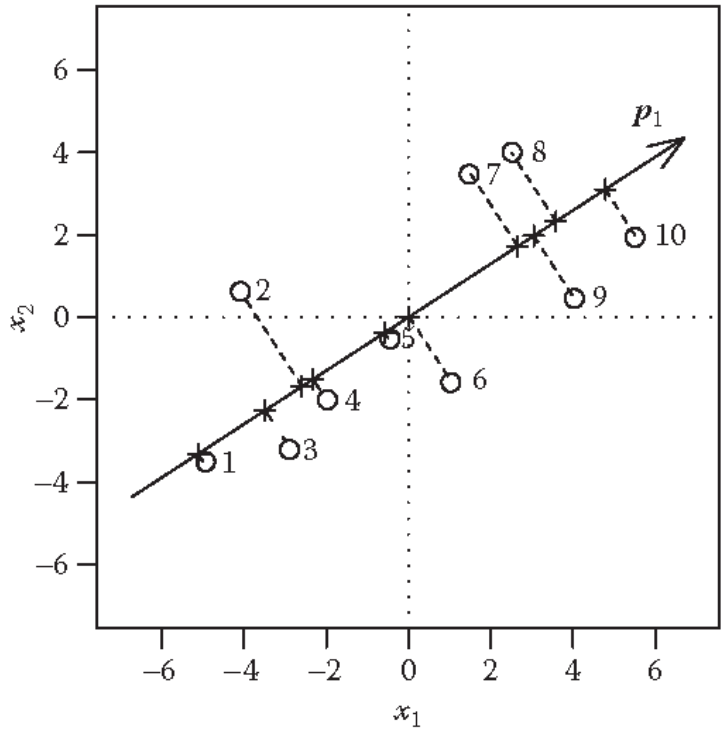
- $\mathbf{t}_1$ is a vector of scores, each score corresponding to an observation
- $\mathbf{p}_1$ is a vector of loadings, each loading corresponding to a variable

6.2.1 The first principal component

The loadings are determined by the following procedure:

- The variance of the scores $\mathbf{t}_1$ is maximized; this transfers the maximum amount of information from the data to the scores.
- If the $\mathbf{X}$-data is mean centred, this can by done by maximizing $\mathbf{t}_1^{\mathrm{T}}\mathbf{t}_1$.
- The loadings are constrained by $\mathbf{p}_1^{\mathrm{T}}\mathbf{p}_1 = 1$ in the optimization.
- Note that this is different from linear regression since the scores are not fitted to any dependent variable.
- These loadings are called the *first principal component* (PC1).

Example. The figure shows 10 objects (o) with two mean-centred variables (x_1, x_2).



The resulting loading vector is

$$\mathbf{p}_1 = [0.839 \quad 0.544]^{\mathrm{T}}$$

The scores (x) are given by

$$t_1 = 0.839x_1 + 0.544x_2$$

They lie on the line $x_2 = (0.544 / 0.839)x_1$.

6.2.2 The second principal component

In the example, we have

- sample variances of variables: $s_{x_1}^2 = 12.22,\ s_{x_2}^2 = 6.72,\ s_{x_1}^2 + s_{x_2}^2 = 18.94$
- sample variance of scores: $s_{t_1}^2 = 16.22$

Thus, the scores of the first principal components explain 86% of the total variance.

We can improve on this by including a *second principal component* (PC2).

We then define a new linear combination of all variables

$$\mathbf{t}_2 = \mathbf{x}_1 p_{12} + \mathbf{x}_2 p_{22} + \cdots + \mathbf{x}_p p_{p2} = \mathbf{X} \cdot \mathbf{p}_2$$

The loadings are determined by the following procedure:

- The variance of the scores $\mathbf{t}_2$ is maximized by maximizing $\mathbf{t}_2^{\mathrm{T}}\mathbf{t}_2$ (if data are mean-centred)
- The loadings are constrained by $\mathbf{p}_2^{\mathrm{T}}\mathbf{p}_2 = 1$ *and* $\mathbf{p}_1^{\mathrm{T}}\mathbf{p}_2 = 0$.
- The last condition makes the second principal component *orthogonal* to the first principal component; thus, they contain no redundant information.

Example. We continue the previous example. Since there are only two variables, x_1 and x_2, the second principal component can be orthogonal to the first in only one way. Even without optimization, we then know that

$$\mathbf{p}_2 = [-0.544 \quad 0.839]^{\mathrm{T}}$$

which satisfies $\mathbf{p}_2^{\mathrm{T}}\mathbf{p}_2 = 1$ and $\mathbf{p}_1^{\mathrm{T}}\mathbf{p}_2 = 0$. The scores are given by

$$t_2 = -0.544x_1 + 0.839x_2$$

The variance of these scores is 2.72; they explain the remaining 14% of the total variance of x_1 and x_2.

The data for this example are shown in the table.

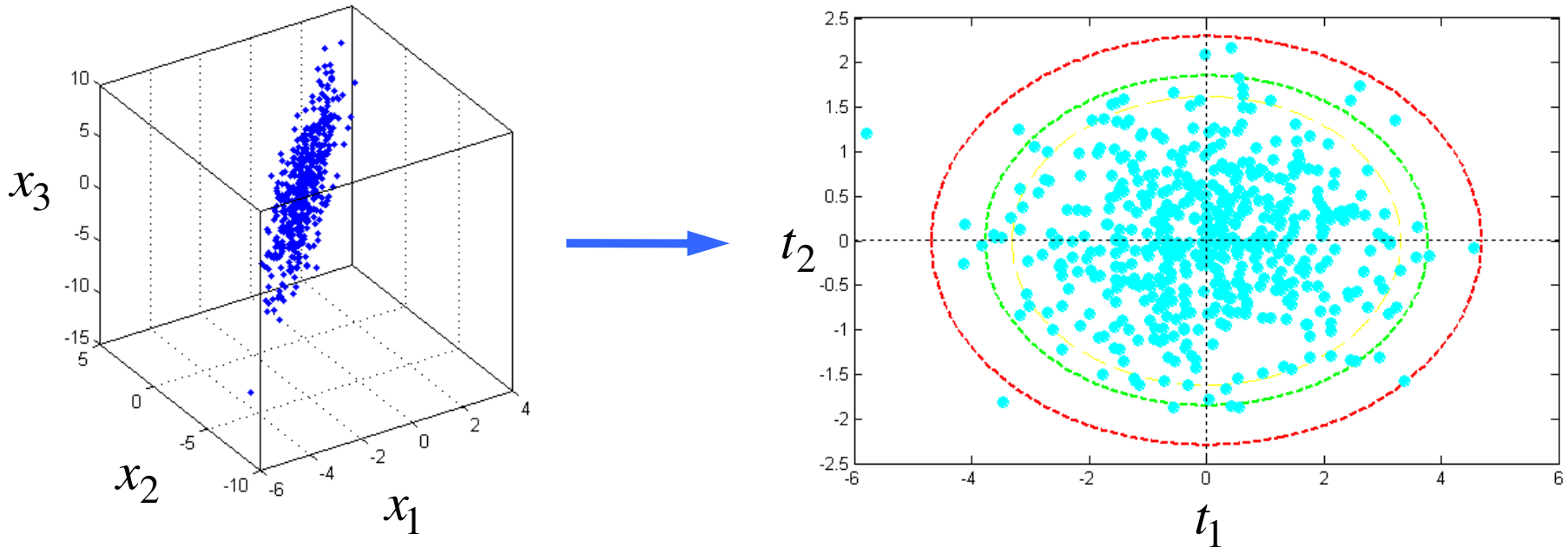
i	x_1	x_2	t_1	t_2
1	−5.0	−3.5	−6.10	−0.21
2	−4.0	0.5	−3.08	2.60
3	−3.0	−3.0	−4.15	−0.88
4	−2.0	−2.0	−2.77	−0.59
5	−0.5	−0.5	−0.69	−0.15
6	1.0	−1.5	0.02	−1.80
7	1.5	3.5	3.16	2.12
8	2.5	4.0	4.27	1.99
9	4.0	0.5	3.63	−1.76
10	5.5	2.0	5.70	−1.32
$\bar{x}$	0.00	0.00	0.00	0.00
v	12.22	6.72	16.22	2.72
v%	64.52	35.48	85.64	14.36

Note: i, Object number; t_1 and t_2 are the PCA scores of PC1 and PC2, respectively; $\bar{x}$, mean; v, variance; v%, variance in percent of total variance.

6.2.3 Higher dimensions

The principal aim of PCA is dimension reduction; that means to explain as much variability (usually variance) as possible with as few PCs as possible. In the previous example, the dimension was not reduced if more than one principal component was used. However, if the data has some underlying structure, the dimension can almost always be reduced.

The following figures illustrate a three-dimensional case, were data can be explained very well by two principal components. This means that the dimension is reduced to two. [From MacGregor, *NPCW* (2009)]



6.2.4 Restoring variables from reduced data

If the number of variables is p, and the number of objects (or time instants of measurements) is n, the number of principal components is limited by

$$k \leq \max(n, p)$$

If the number of variables is large, we usually have $k \ll \max(n, p)$.

Assume that the number of PCs is k. The score and loading vectors for all PCs can be collected into a score matrix $\mathbf{T}$ and a loading matrix $\mathbf{P}$ according to

$$\mathbf{T} = \begin{bmatrix} \mathbf{t}_1 & \mathbf{t}_2 & \cdots & \mathbf{t}_k \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \cdots & \mathbf{p}_k \end{bmatrix}$$

All linear combinations of the data in $\mathbf{X}$ can then be expressed compactly as

$$\mathbf{T} = \mathbf{X} \cdot \mathbf{P}$$

All PCs $\ell, \ell = 1, \ldots, k$, are determined by maximizing the variance with the constraint that every PC is orthogonal to all other PCs. This means that $\mathbf{P}^{\mathrm{T}}\mathbf{P} = \mathbf{I}$. If $\mathbf{T}$ and $\mathbf{P}$ have been stored, which requires much less memory than storing $\mathbf{X}$ when $k \ll \max(n, p)$, the original $\mathbf{X}$ can be calculated approximately (since some info or *noise is lost* in the data reduction) by

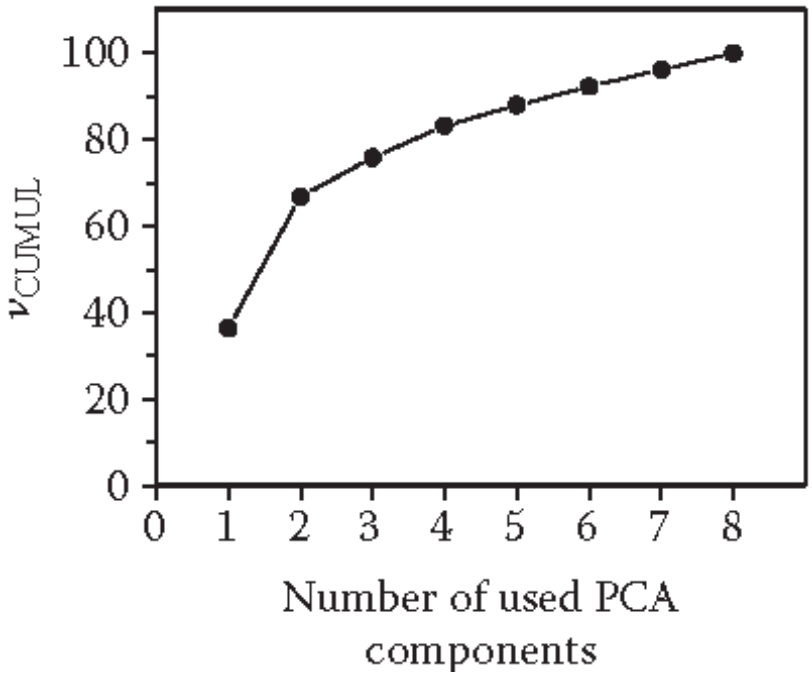
$$\mathbf{X} \approx \mathbf{T} \cdot \mathbf{P}^{\mathrm{T}}$$

6.2.5 Number of principal components

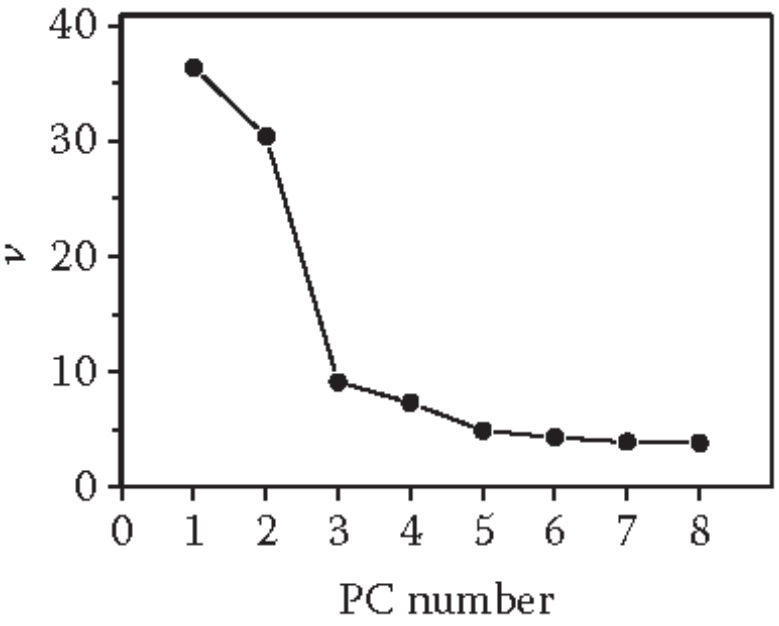
The variance of the PC scores, preferably given in percent of the total variance of the original variables, are important indicators of how many PCs to include. Because of the way the PCs are determined, the scores of each new PC has a lower variance than the scores of the previous PC.

Following are some methods for determining how man PCs to include.

- If the score variance of a certain PC is close to the noise variance of the data, the PC does not contain any useful information. Obviously, the number of PCs should be less than this number. However, in practice it may be difficult to apply this criterion.
- A simple method, illustrated by the fig., is to plot the cumulative variance of scores (the variance for each new PC added to the variance of previous PCs) against the PC number. As a rule of thumb, the PCs should explain at least 80 %, maybe 90%, of the total variance.



- However, it might be more useful to plot the variance of individual PCs against the PC number. Such a plot is called a *scree plot*. A small variance reduction compared to the previous reduction when a PC is added, indicates that the new PC should not be included. In this example, two or three PCs is the indicated choice.



- If the data is autoscaled so that each variable has a variance = 1, a rule of thumb is to select only PCs with a variance > 1.
- A more reliable, but computationally much more demanding, way is to use cross-validation based on "bootstrap techniques" for the selection of the number of PCs. (This is dealt with later.)
- Selection of the number of PCs can also be based on statistical tests.

6.3 Numerical calculation of PCs

6.3.1 Calculation via eigenvalues

It is useful (but not necessary) to consider the *stochastic nature* of the problem. Because the variables x_j, $j = 1, \ldots, p$, are stochastic, the latent variables

$$t_\ell = x_1 p_{1\ell} + x_2 p_{2\ell} + \cdots + x_p p_{p\ell} = \mathbf{x}^{\mathrm{T}} \mathbf{p}_\ell, \quad \ell = 1, \ldots, k$$

are also stochastic variables (note that x_j and t_ℓ denote variables, not data). We want to maximize the variance

$$\sigma_{t_\ell}^2 \equiv \mathrm{Var}(t_\ell) = \mathrm{E}\{[t_\ell - \mathrm{E}(t_\ell)]^2\} = \mathrm{E}\{[(\mathbf{x}^{\mathrm{T}} - \mathrm{E}(\mathbf{x}^{\mathrm{T}}))\mathbf{p}_\ell]^2\}$$

$$= \mathbf{p}_\ell^{\mathrm{T}} \mathrm{E}\{[\mathbf{x} - \mathrm{E}(\mathbf{x})][\mathbf{x}^{\mathrm{T}} - \mathrm{E}(\mathbf{x}^{\mathrm{T}})]\}\mathbf{p}_\ell \equiv \mathbf{p}_\ell^{\mathrm{T}} \mathrm{Cov}(\mathbf{x})\mathbf{p}_\ell \equiv \mathbf{p}_\ell^{\mathrm{T}} \mathbf{C} \mathbf{p}_\ell$$

where $\mathbf{C}$ is the covariance matrix of the vector of stochastic variables in $\mathbf{x}$. We can include the constraint $\mathbf{p}_\ell^{\mathrm{T}} \mathbf{p}_\ell = 1$ by using the Lagrangian expression

$$\sigma_{t_\ell}^2 = \mathbf{p}_\ell^{\mathrm{T}} \mathbf{C} \mathbf{p}_\ell - \lambda_\ell (\mathbf{p}_\ell^{\mathrm{T}} \mathbf{p}_\ell - 1), \quad \ell = 1, \ldots, k$$

Maximization of $\sigma_{t_\ell}^2$ by setting the derivative of $\sigma_{t_\ell}^2$ with respect to $\mathbf{p}_\ell$ equal to zero gives the solution

$$\mathbf{C}\mathbf{p}_\ell = \lambda_\ell \mathbf{p}_\ell, \quad \ell = 1, \ldots, k$$

We need to find $\mathbf{p}_\ell$, $\ell = 1, \ldots, k$, that are nonzero and that satisfy the previous expression, where $\mathbf{C}$ is a matrix and λ_ℓ is a scalar. This is known as an *eigenvalue problem*. Here

- λ_ℓ is an eigenvalue of $\mathbf{C}$
- $\mathbf{p}_\ell$ is the corresponding eigenvector of $\mathbf{C}$

The eigenvalues can be found by solving the equation $\det(\lambda\mathbf{I} - \mathbf{C}) = 0$ and the corresponding eigenvectors by solving sets of linear equations. However, this is done by standard software.

The covariance matrix $\mathbf{C}$ is a symmetric matrix. The following then applies:

- all $\lambda_\ell \geq 0$, $\ell = 1, \ldots$
- $\mathbf{p}_m^{\mathrm{T}}\mathbf{p}_\ell = 0$, $m \neq \ell$, $\ell = 1, \ldots$

From $\mathbf{C}\mathbf{p}_\ell = \lambda_\ell\mathbf{p}_\ell$ it is clear that $\mathbf{p}_\ell$ is not unique; it can be multiplied by an arbitrary nonzero factor. Thus, we can always choose to scale $\mathbf{p}_\ell$ so that

- $\mathbf{p}_\ell^{\mathrm{T}}\mathbf{p}_\ell = 1$, $\ell = 1, \ldots$

Thus, all principal component requirements can be satisfied by an eigenvalue calculation.

It is interesting to note that

$$\sigma_{t_\ell}^2 = \mathbf{p}_\ell^{\mathrm{T}} \mathbf{C} \mathbf{p}_\ell = \mathbf{p}_\ell^{\mathrm{T}} \lambda_\ell \mathbf{p}_\ell = \lambda_\ell \mathbf{p}_\ell^{\mathrm{T}} \mathbf{p}_\ell = \lambda_\ell$$

This means that the variance of the latent variables is equal to the eigenvalues.

The first principal component is thus given by the eigenvector for the largest eigenvalue, the second principal component by the eigenvector for the second largest eigenvalue, etc. The scores can be calculated from

$$\mathbf{t}_\ell = \mathbf{X}\mathbf{p}_\ell \;,\; \ell = 1, \ldots, k \quad \Leftrightarrow \quad \mathbf{T} = \mathbf{X} \cdot \mathbf{P}$$

The calculation of eigenvalues requires the covariance matrix $\mathbf{C}$. As formulated here, this is the true covariance matrix of the entire population, which is unknown. We can use any approximation that seems appropriate, e.g.

- the sample covariance matrix $\mathbf{C} \approx \frac{1}{n-1} \mathbf{X}^{\mathrm{T}} \mathbf{X}$ (if data is mean-centred)
- some robust approximation of the covariance matrix, e.g.
  - the minimum covariance determinant (MCD)
  - a covariance matrix based on the median absolute deviation (MAD)

6.3.2 Calculation via singular value decomposition (SVD)

The covariance matrix has size $p \times p$, where p is the number of variables. In chemometrics, we often have $p > n$, where n is the number of observations (instruments are cheap, time is not…). Then, the sample covariance matrix

- might be very large
- is singular

This can be overcome by doing a *singular value decomposition* of $\mathbf{X}$. Every matrix can be decomposed as

$$\mathbf{X} = \mathbf{U} \cdot \mathbf{\Sigma} \cdot \mathbf{V}^{\mathrm{T}} \quad \text{with} \quad \mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I}, \quad \mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{I}$$

and $\mathbf{\Sigma}$ a diagonal matrix with diagonal elements $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$.

We note that $\mathbf{X}^{\mathrm{T}}\mathbf{X} \cdot \mathbf{V} = \mathbf{V}\mathbf{\Sigma}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}} \cdot \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}} \cdot \mathbf{V} = \mathbf{V} \cdot \mathbf{\Sigma}^{\mathrm{T}}\mathbf{\Sigma}$ from which it follows

$$\mathbf{X}^{\mathrm{T}}\mathbf{X} \cdot \mathbf{v}_i = \sigma_i^2 \cdot \mathbf{v}_i, \quad i = 1, \ldots, n$$

where $\mathbf{v}_i$ is the *i*th column vector of $\mathbf{V}$. Thus, σ_i^2 is an eigenvalue and $\mathbf{v}_i$ is the corresponding eigenvector of $\mathbf{X}^{\mathrm{T}}\mathbf{X}$. This means that

$$\mathbf{P} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_k \end{bmatrix}$$

A problem is that $\mathbf{X}$, from which we make the SVD, could still be a large matrix if p is large. If $n \leq p$, $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ is a smaller matrix than $\mathbf{X}$ and $\mathbf{X}^{\mathrm{T}}\mathbf{X}$. As shown below, we can find the solution by means of an SVD of $\mathbf{X}\mathbf{X}^{\mathrm{T}}$.

From $\mathbf{X} = \mathbf{U} \cdot \mathbf{\Sigma} \cdot \mathbf{V}^{\mathrm{T}}$ we have

$$\mathbf{X}\mathbf{X}^{\mathrm{T}} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}} \cdot \mathbf{V}\mathbf{\Sigma}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}} = \mathbf{U}\mathbf{\Sigma} \cdot \mathbf{\Sigma}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}} = \mathbf{U}\mathbf{\Sigma}^{2}\mathbf{U}^{\mathrm{T}}$$

which shows that $\mathbf{U}$ and $\mathbf{\Sigma}^2$ are obtained by an SVD of $\mathbf{X}\mathbf{X}^{\mathrm{T}}$.

Since $\mathbf{\Sigma}^{-1}\mathbf{U}^{\mathrm{T}} \cdot \mathbf{X} = \mathbf{\Sigma}^{-1}\mathbf{U}^{\mathrm{T}} \cdot \mathbf{U} \cdot \mathbf{\Sigma} \cdot \mathbf{V}^{\mathrm{T}} = \mathbf{V}^{\mathrm{T}}$ we can then calculate

$$\mathbf{V} = \mathbf{X}^{\mathrm{T}} \cdot \mathbf{U} \cdot \mathbf{\Sigma}^{-1}$$

from which we get

$$\mathbf{P} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_k \end{bmatrix}$$

Since the SVD is based on $\mathbf{X}\mathbf{X}^{\mathrm{T}}$, which is equivalent to (but easier than) an SVD of $\mathbf{X}^{\mathrm{T}}\mathbf{X}$, it corresponds to the use of the sample covariance matrix as an approximation of the true covariance matrix. Thus, robust approximations of the covariance matrix cannot be used with the SVD solution.

6.4 Effects of data preprocessing

Unfortunately, PCA is quite sensitive to the

- preprocessing of data
- outliers in the data

However, this also means that the preprocessing and handling of outliers is important. In order to do this

- insight into the problem is needed;
- even then, it is quite subjective.

6.4.1 Centring

Because the principal components (or more generally, the latent variables) are linear combinations of the original variables of the form $\mathbf{T} = \mathbf{XP}$,

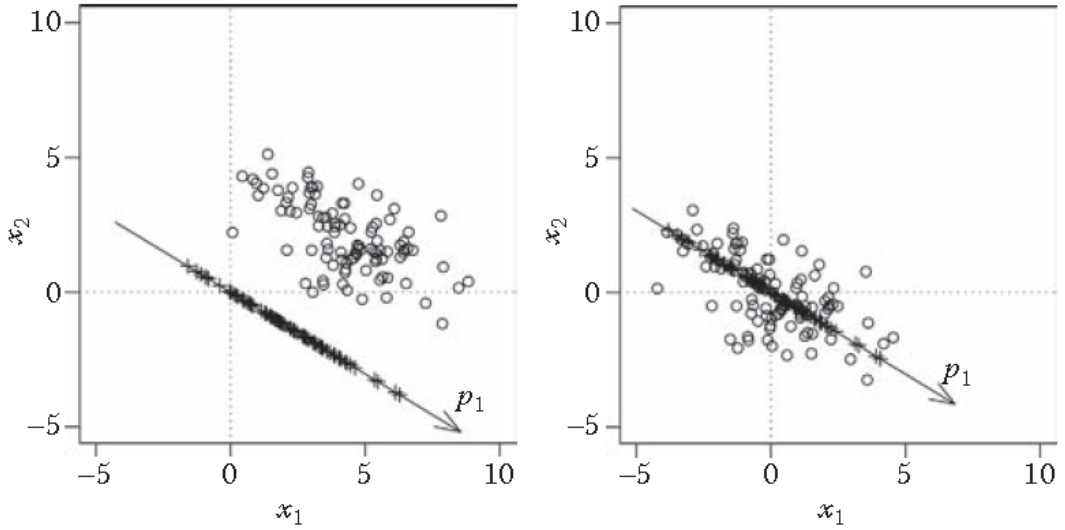
- there is no constant term;
- the scores will lie on a line through the origin of the *x*-coordinate system.

For this reason, it is natural to mean-centre the *x*-variables, i.e.

- subtract the mean value of each variable from its true values.

The figures illustrate the effect of mean-centring a two-dimensional data set:

- left: original *x*-data
- right: mean-centred



The direction of PC1 is not affected (since it is an orthogonal projection of the data), but the scores are not mean-centred if the data are not.

Using uncentred data will have unfavourable effects on, e.g.,

- statistical tests (of confidence)
- data recovery by $\mathbf{X} = \mathbf{TP}^{\mathrm{T}}$ (after previous data reduction)

There are also situations when **data centring might *not* be used**, e.g. if we are interested in studying the

- deviation from desired values (process monitoring and control)
- effect of some (medical) treatment (before-after comparison)

For robustness reasons (due to outliers), we might use some other centring technique than mean-centring, e.g. median-centring.

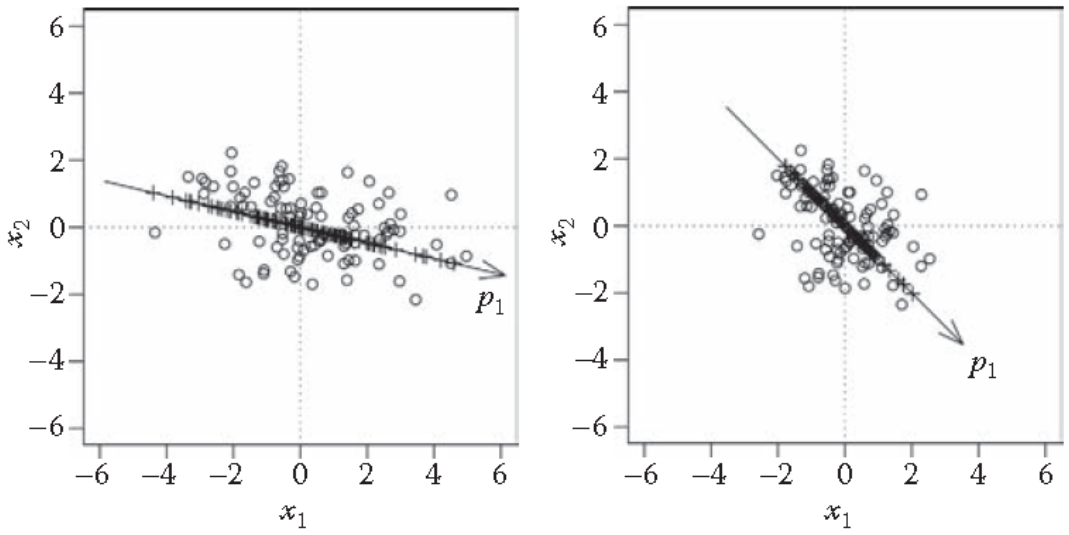
6.4.2 Scaling

Scaling is used to affect the magnitude of data by applying a scaling factor to the (centred) data of each variable.

- The purpose is to make the numerical values of each variable equally significant (on average).
- This is important when the variables express different types of quantities with different units.

Since PCA (and other related techniques) is based on the concept of maximizing the variance of the principal components, which are linear combinations of the original data, it is natural to scale the *x*-variables to make their variances equal. This is obtained by dividing the data of each variable by the (sample) standard deviation of the variable data.

The *scaling affects the direction of the principal components*, as shown by the figures (left – unscaled; right – scaled).



It is ***not always right*** to make all variable variances equal. For example:

- If (some of) the variables denote the same physical quantity, using the same units (e.g. temperature measurements in different places in a process), the difference in variances may contain important information — this information is destroyed if the variables are scaled to equal variance.
- If a variable contains mostly noise, it does not contain much useful information; scaling to the same variance as important variables will then put too much weight on the noisy data.

The use of

- unscaled data gives variables with high variance high importance
- data scaled to equal variance tends to make variables with low original variance more importance

As a compromise, so-called *Pareto scaling* is sometimes used. Then

- variables are scaled by dividing data with the square-root of the (sample) standard deviation.

If the relative importance of variables is known from process knowledge, the variable weightings can be up-graded or down-graded to reflect this.

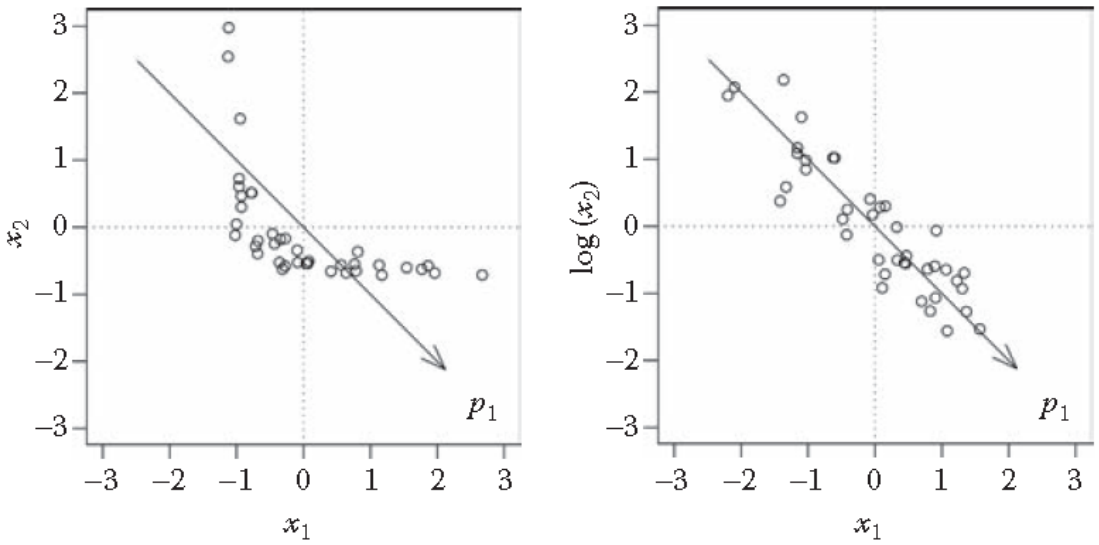
6.4.3 Variable transformations

The main goal of PCA is dimension reduction and an analysis in the reduced dimension, where the main multivariate characteristics of the process are retained.

- Both the dimension reduction and the analysis is best performed if the data distribution is elliptically symmetric around the centre.
- They will not work well for highly skewed data.

A remedy to this is to make the data more normally distributed by a variable transformation.

The figure to the left shows skewed autoscaled data. Even if PC1 here explains 79% of the variation, it fails to explain the main data variability. After a logarithmic transformation of x_2, the result is much better (right figure).

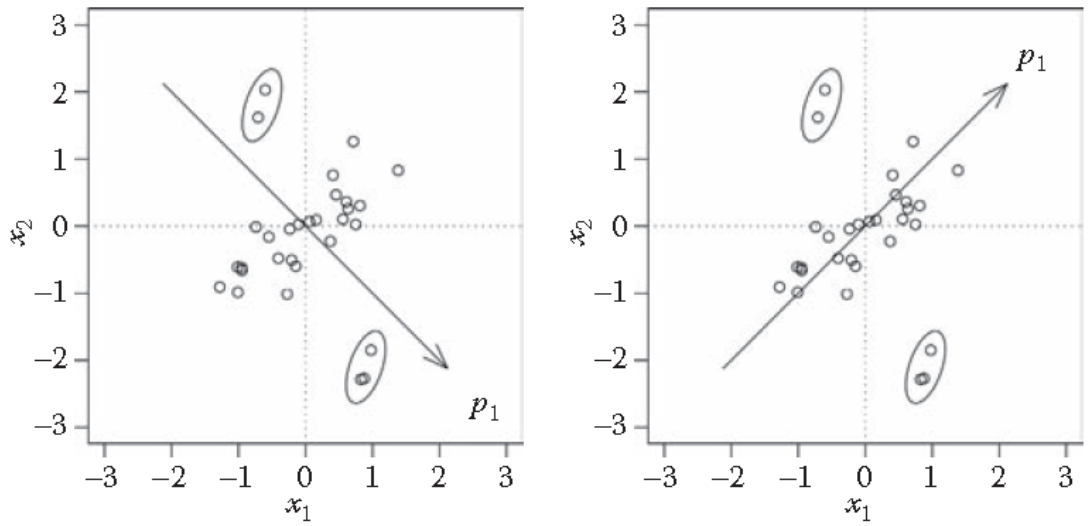


6.4.4 Outliers

PCA is sensitive to outliers. Outliers increase classical nonrobust measures of variance, and since the PCs are following directions of maximum variance, they will be attracted by outliers.

The figures illustrate this effect. Here groups of (probable) outliers are marked by ellipses and the arrows show the direction of the first principal component.

- The left-hand figure shows the result when classical nonrobust PCA is used — PC1 does not follow the direction of the main data.
- The right-hand figure shows the result when a robust version of PCA is used — now the main data is well described.



6.5 Evaluation and diagnostics

The PCA result and the data can be analysed and evaluated in many ways.

- Observation diagnostics
  - outliers
  - groups and trends
- Variable diagnostics
  - loadings and correlations
  - explained variance
- Model diagnostics
  - overall fit
  - cross-validation (prediction)

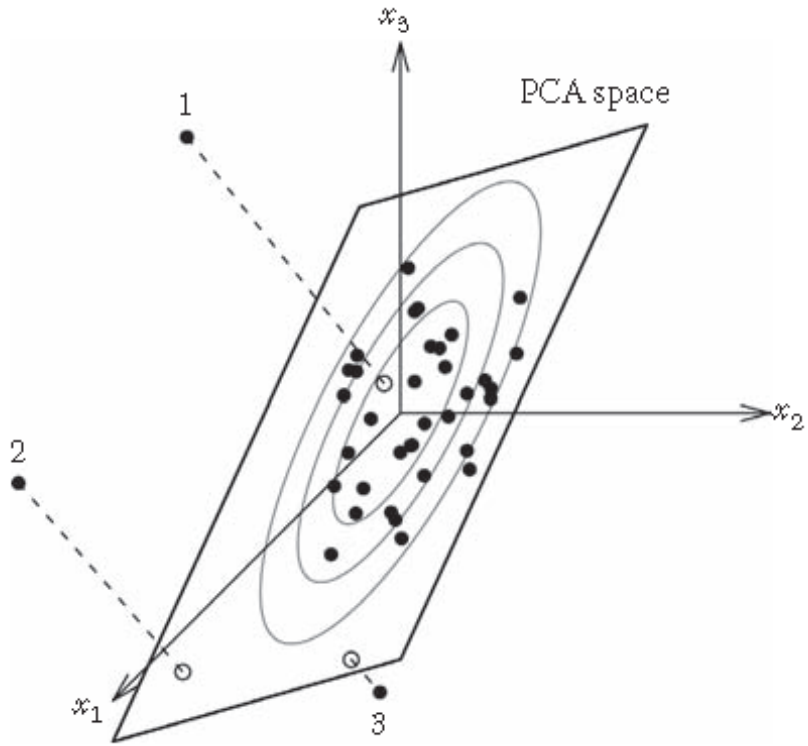
6.5.1 Observation diagnostics

One of the main diagnostic issues is the detection of various kinds of **outliers** because they may

- distort classical (nonrobust) PCA
- reveal something interesting about the process (outliers need not be measurement errors)

As interpreted by the PCA, there are essentially two kinds of outliers — outliers that result in

- an exceptionally large score (PC value) $\mathbf{T}$ for some observation
- a large residual $\mathbf{E} = \mathbf{X} - \mathbf{T}\mathbf{P}^{\mathrm{T}}$ for some observation; such outliers are orthogonal to the space covered by the PCs



The types of outliers are illustrated in the figure (black point, projection is white):

- pt 1 = small score, large residual
- pt 2 = large score, large residual
- pt 3 = large score, small residual

Score outliers

Large score outliers are seen in a **scores plot** where one principal component is plotted against another. If there are more than two PCs,

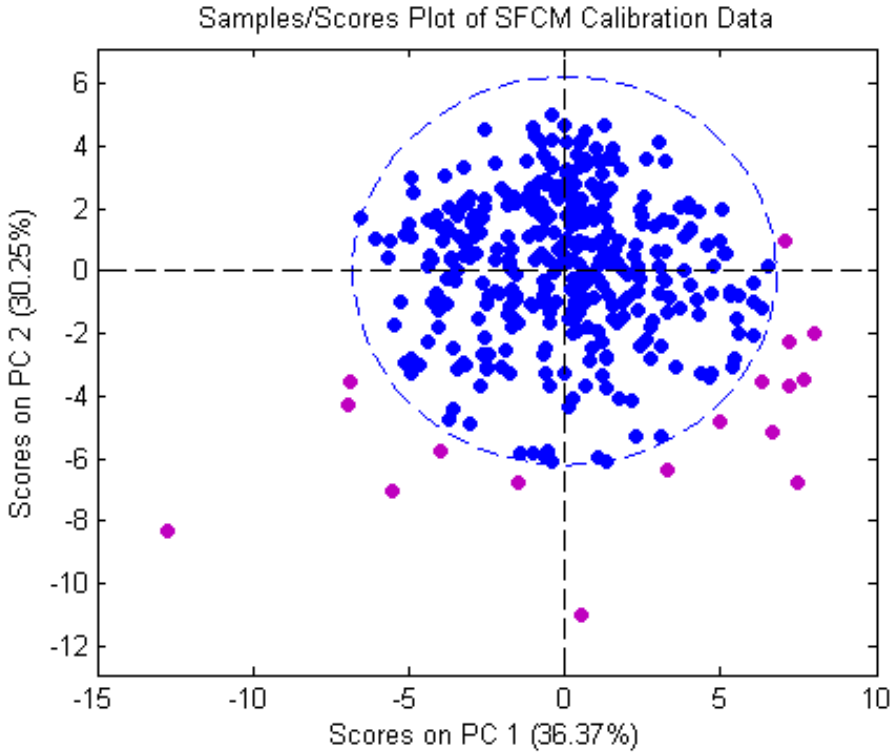
- there are several possible 2-D plots
- 3-D plots could be used (by choosing the z axis to have a PC). Note that 3-D plots can be rotated in the PLS toolbox to obtain a suitable view.

A statistical confidence region can be added to a 2-D score plot as an aid in the detection of outliers; see the figure.

The confidence region is based on

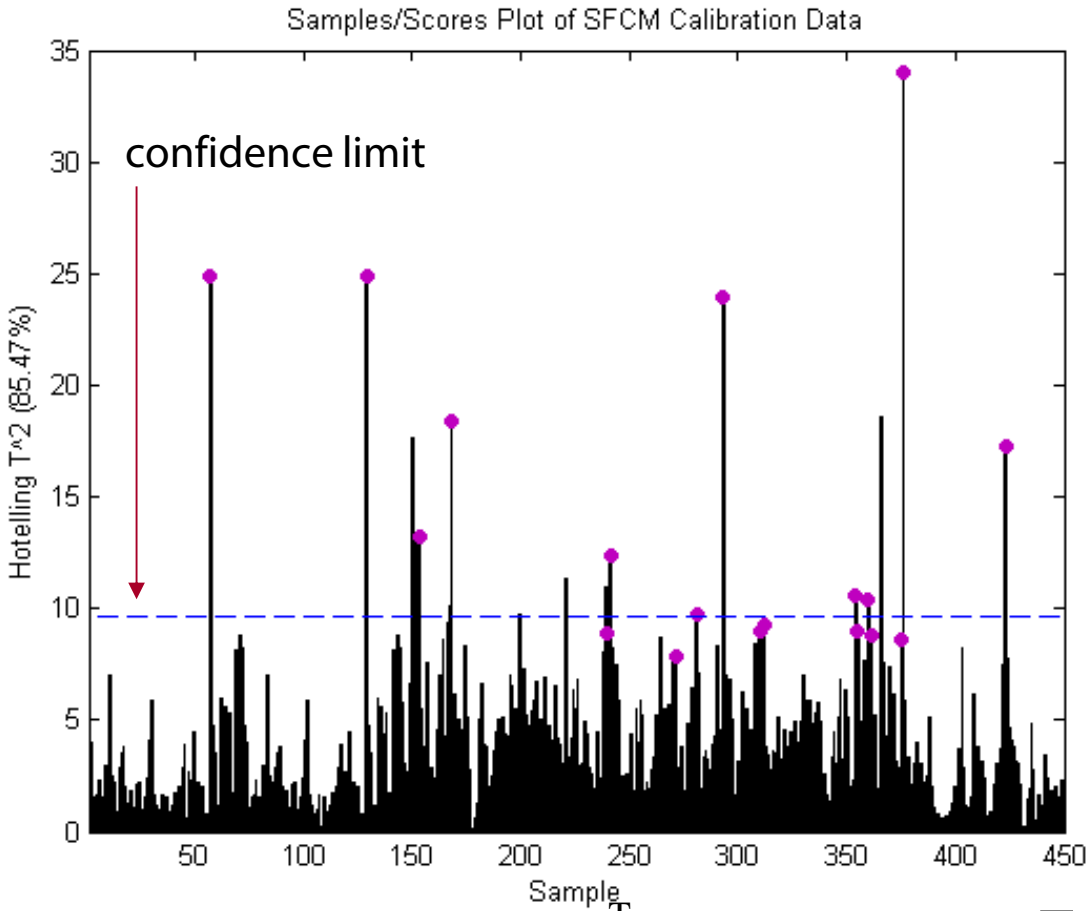
- Hotelling's T^2-test, which is a multivariate version of Student's t-test
- a desired confidence limit can be selected (usually 95% or 99%); in the PLS-toolbox it can be selected in the Plot Controls window

Note that this statistic is *only a guide*, not a hard rule for outlier selection.



- When there are only two PCs, a 2D score plot is sufficient for detection of score outliers according to some confidence level (usually 95 % or 99%).
- When there are more than 2 PCs
  - some outliers may remain undetected in a 2D plot
  - some observations believed to be outliers may not be outliers when all PCs are considered jointly
- For more than two PCs, it is safest to consider Hotelling's T^2 statistics jointly for all PCs.
- This figure shows that
  - some outliers are undetected
  - some indicated outliers are not outliers

in the previous 2D plot.



- Hotelling's T^2 value for the ith observation is

$$T_i^2 = \mathbf{t}_i^{\mathrm{T}} \mathbf{\Lambda} \mathbf{t}_i$$

where $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_k)$ are PC eigenvalues and $\mathbf{t}_i^{\mathrm{T}}$ is the ith row of T.

Residual outliers

Large residual outliers can be detected by plotting "**Q residuals**". For a given observation i, the Q residual is defined as

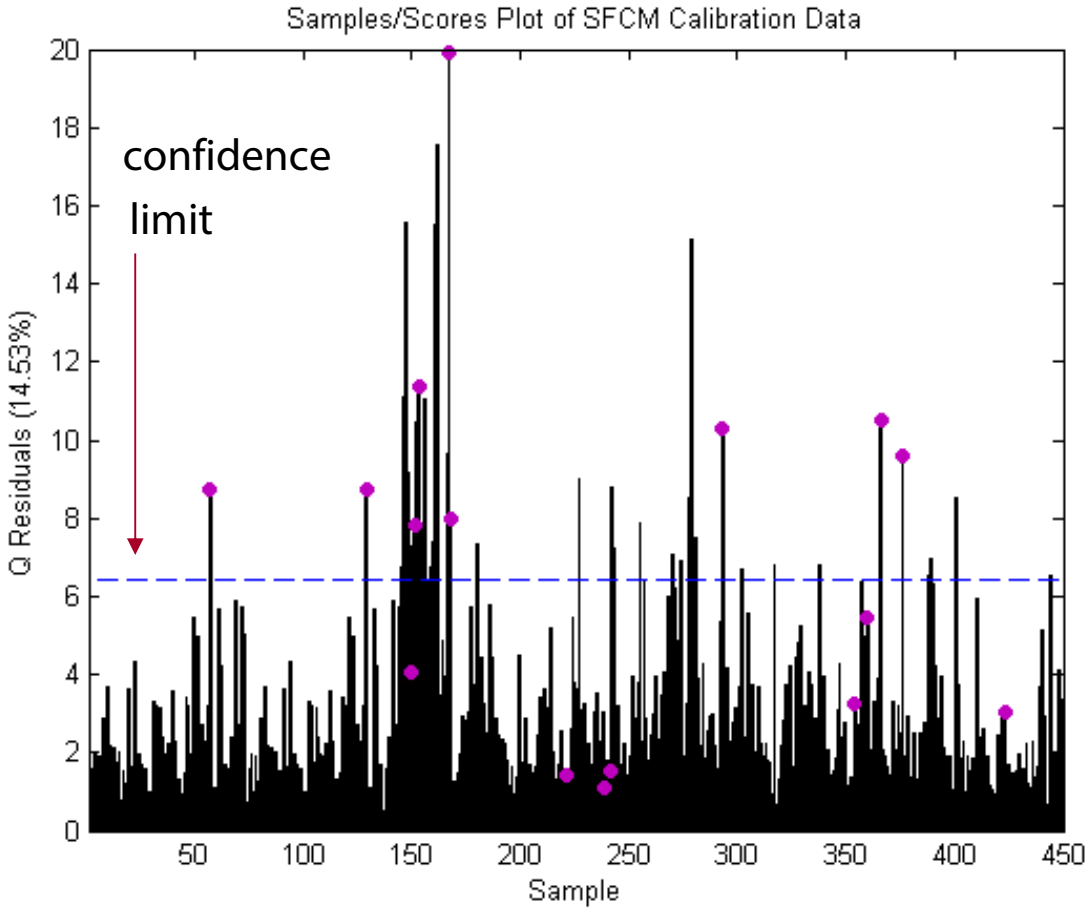
$$Q_i = \mathbf{e}_i^{\mathrm{T}}\mathbf{e}_i = \mathbf{x}_i^{\mathrm{T}}(\mathbf{I} - \mathbf{P}\mathbf{P}^{\mathrm{T}})\mathbf{x}_i$$

where $\mathbf{e}_i^{\mathrm{T}}$ is the *i*th row of the *residual matrix* $\mathbf{E} = \mathbf{X} - \mathbf{T}\mathbf{P}^{\mathrm{T}}$ and $\mathbf{x}_i^{\mathrm{T}}$ is the *i*th row of the data matrix $\mathbf{X}$.

- The Q statistic indicates how well each *observation* matches the PCA model.

The figure shows the Q residuals for the same example as the previous figure — the red points are score outliers

- some score outliers are also residual outliers
- some score outliers are not residual outliers



Residuals vs. scores

A plot of Q residuals vs. Hotelling's T^2 statistic gives a compact view of both residual and score outliers (as well as inliers).

The axis texts of the figure show that

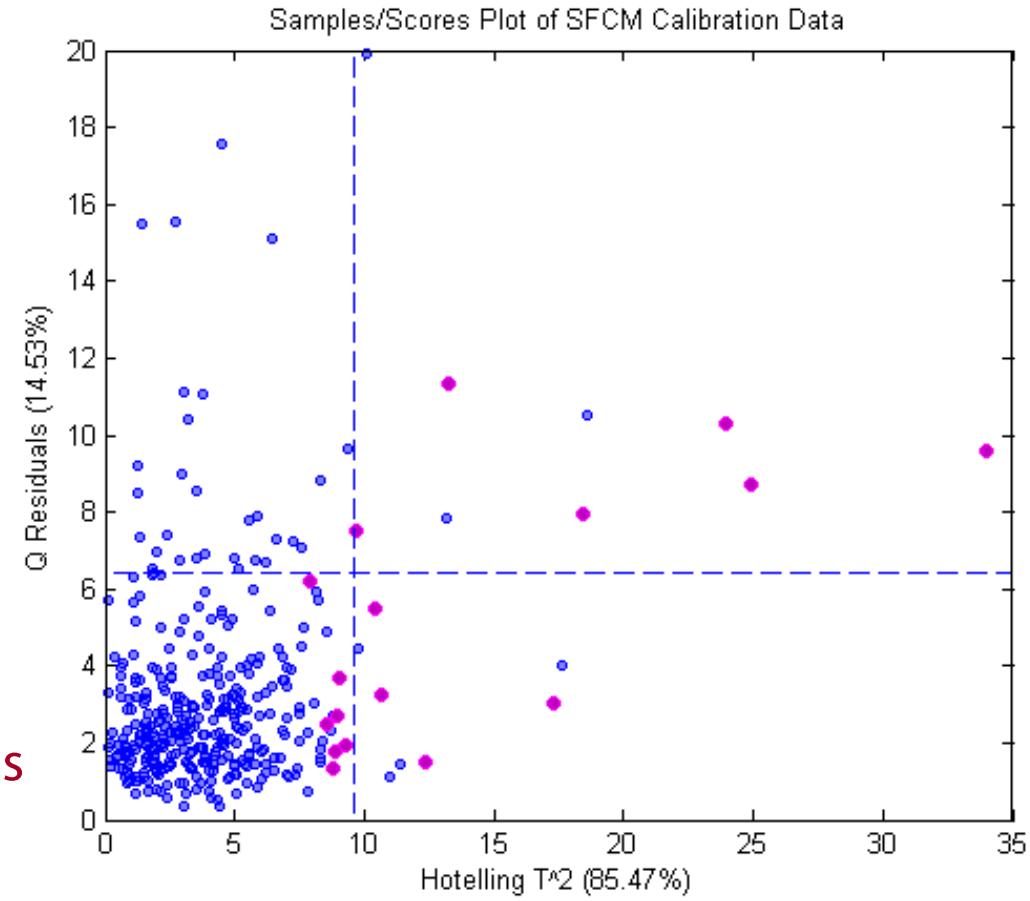
- scores capture 85.47 % of the total $\mathbf{X}$ variance
- 14.53 % of the variance remains in the residuals

The figure shows that there are

- many large score outliers, some are also (slight) residual outliers
- five very large residual outliers, which are not score outliers

Note that the PCA model describes

- score outliers well
- residual outliers badly



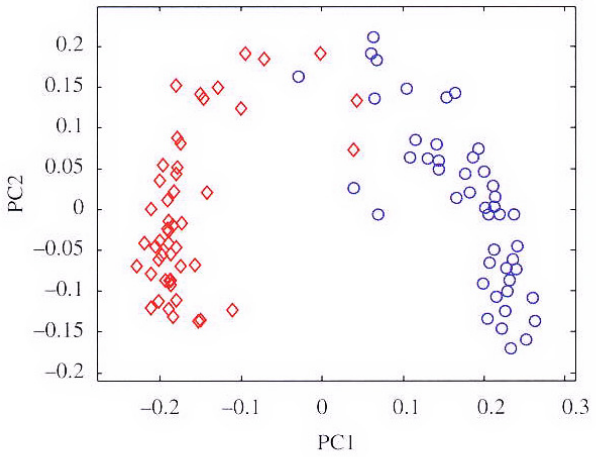
Groups and trends

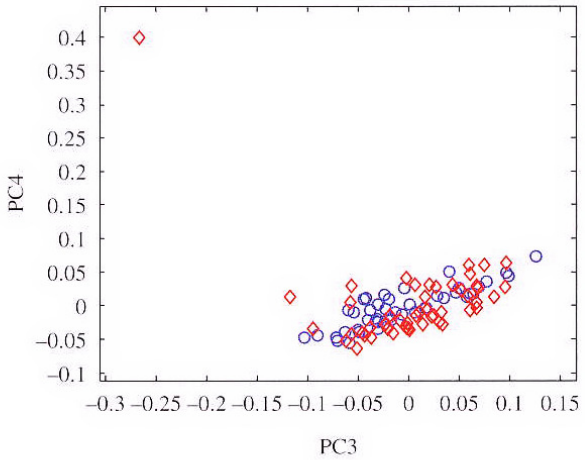
Groups and trends can be detected from a **scores plot**

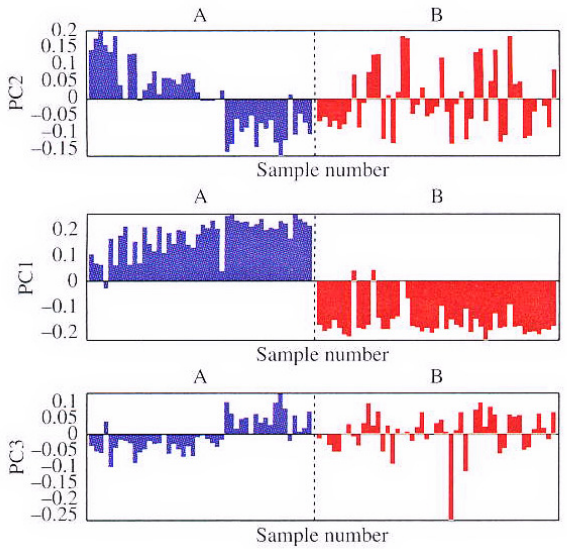
- often PC2 vs. PC1 (but could be any PC# vs. another PC#)
- sometimes a suitably rotated 3D plot is needed

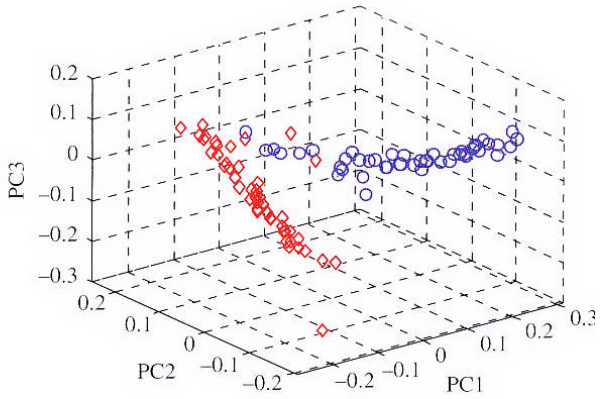
In the PLS-toolbox

- a group or trend can be selected to have another colour
- the colouring will appear in all plots
- for time-series data, it is then easy to see in a plot of PC# vs. observation # where in time the abnormality is (could be a severe process disturbance)









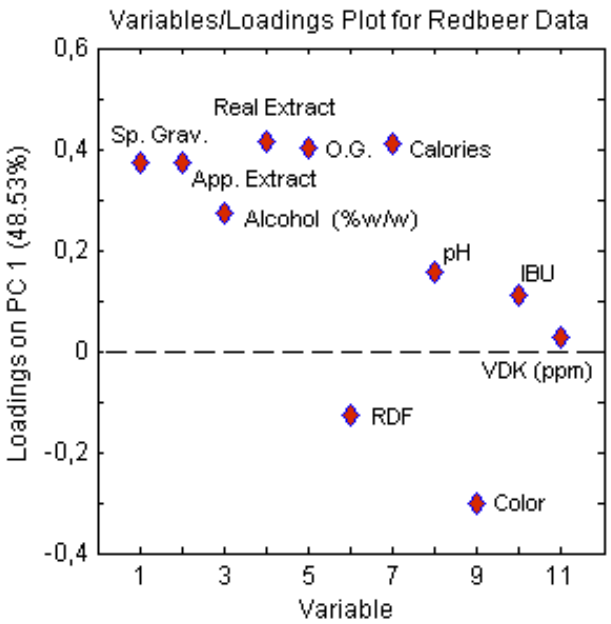
6.5.2 Variable diagnostics

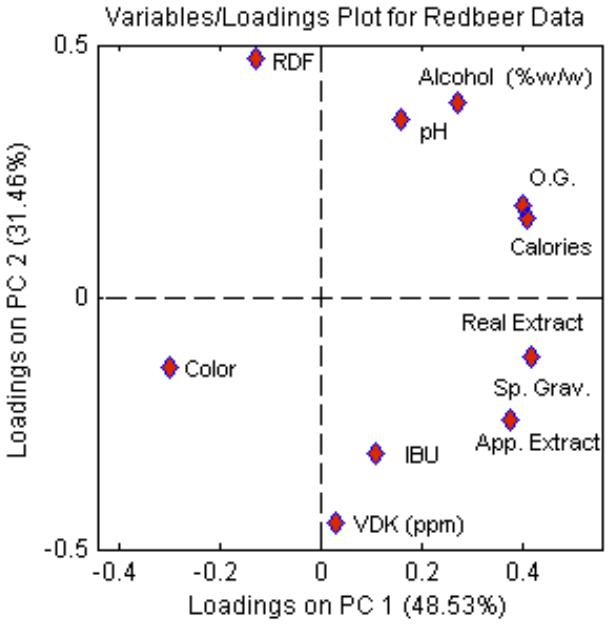
Loadings and correlations

In variable diagnostics, the contribution of each variable x_j to the PC model is studied.

- A **loadings plot** shows loadings ("weights") ($p_{j\ell}$-parameters) of the variables x_j. Loadings for a chosen PC can be plotted
  - against the variable index (upper fig.)
  - against another PC (lower fig.)
  - against two other PCs (3D plot)
- If the variables have been properly scaled so that equal variations are equally significant
  - a loading far from zero (or the origin) means that the variable has a large impact on the PC model (i.e. the variable is "important")
  - variables that are close to each other have similar effects; they may be highly correlated

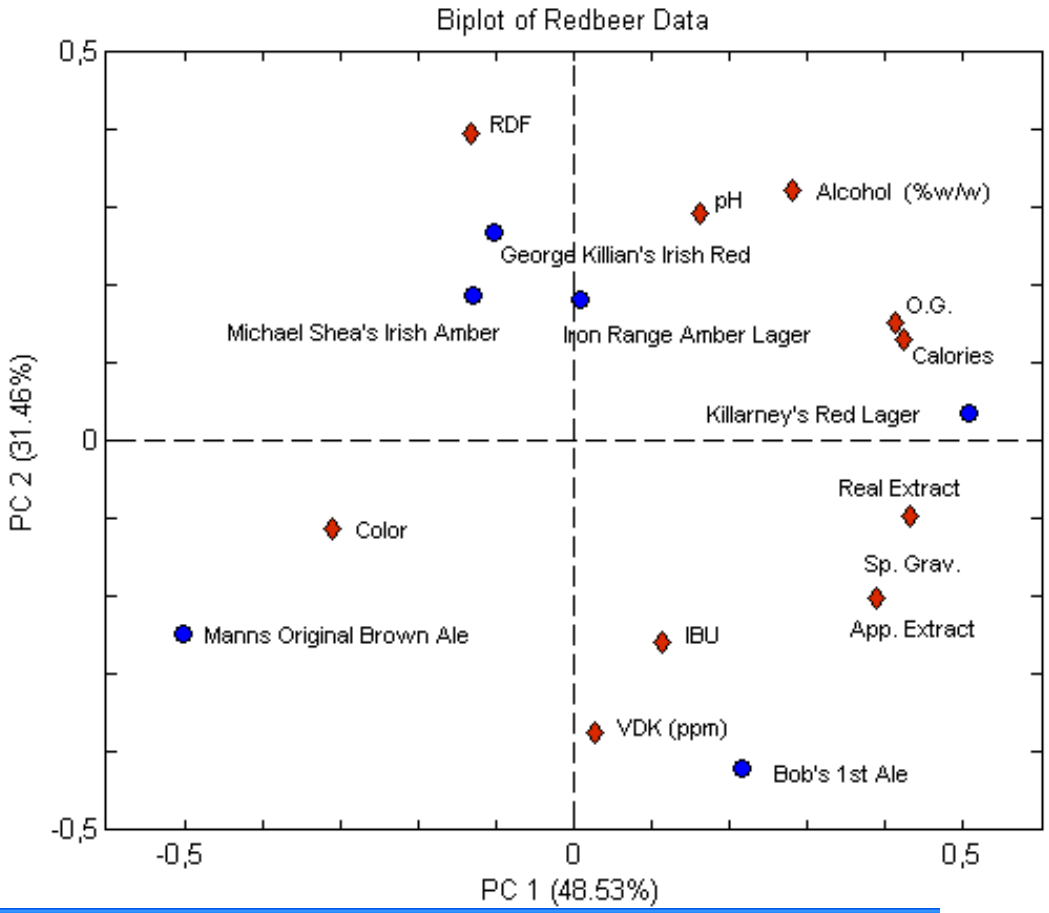
[The plots show various measured properties of a number of different "redbeers"; data from the PLS-toolbox.]





- A **biplot** combines a scores plot and loadings plot; it gives info about
  - clustering of objects and variables
  - relationships between objects and variables.
- Objects and variables tend to be
  - positively correlated if close to each other
  - negatively correlated if far away from each other

  Compare figure and table!
- Variables need to be properly scaled (e.g. equal variance)



Redbeerdata.xls

	A	B	C	D	E	F	G	H	I	J	K	L
1		Specific Gravity	Apparent Extract	Alcohol (%w/w)	Real Extract (%w/w)	O.G.	RDF	Calories	pH	Color	IBU	VDK (ppm)
2												
3	Michael Shea's Irish Amber	1,01016	2,60	3,64	4,29	11,37	63,70	150,10	4,01	,0	16,1	0,02
4	Iron Range Amber Lager	1,01041	2,66	3,81	4,42	11,82	64,00	156,30	4,33	11,6	21,1	0,04
5	Bob's 1st Ale	1,01768	4,50	3,17	5,89	12,04	52,70	162,70	3,93	30,7	21,1	0,11
6	Manns Original Brown Ale	1,00997	2,55	2,11	3,58	7,77	54,90	102,20	4,05	58,9	18,2	0,05
7	Killarney's Red Lager	1,01915	4,87	3,83	6,64	14,0	54,30	190,20	4,36	12,3	17,9	0,02
8	George Killian's Irish Red	1,01071	2,74	3,88	4,48	12,0	64,10	158,80	4,28	53,0	14,2	0,03

Explained variance

Each column $\mathbf{e}_j$ of the residual matrix $\mathbf{E}$ contains information related to a given variable x_j. For each variable, the sum of squares

- $\mathbf{x}_j^{\mathrm{T}}\mathbf{x}_j$ is proportional to the sample variance of the variable
- $\mathbf{e}_j^{\mathrm{T}}\mathbf{e}_j$ is proportional to the *residual variance not explained* by the PC model

The *explained variance* of a variable x_j with a given number of PCs can be defined as

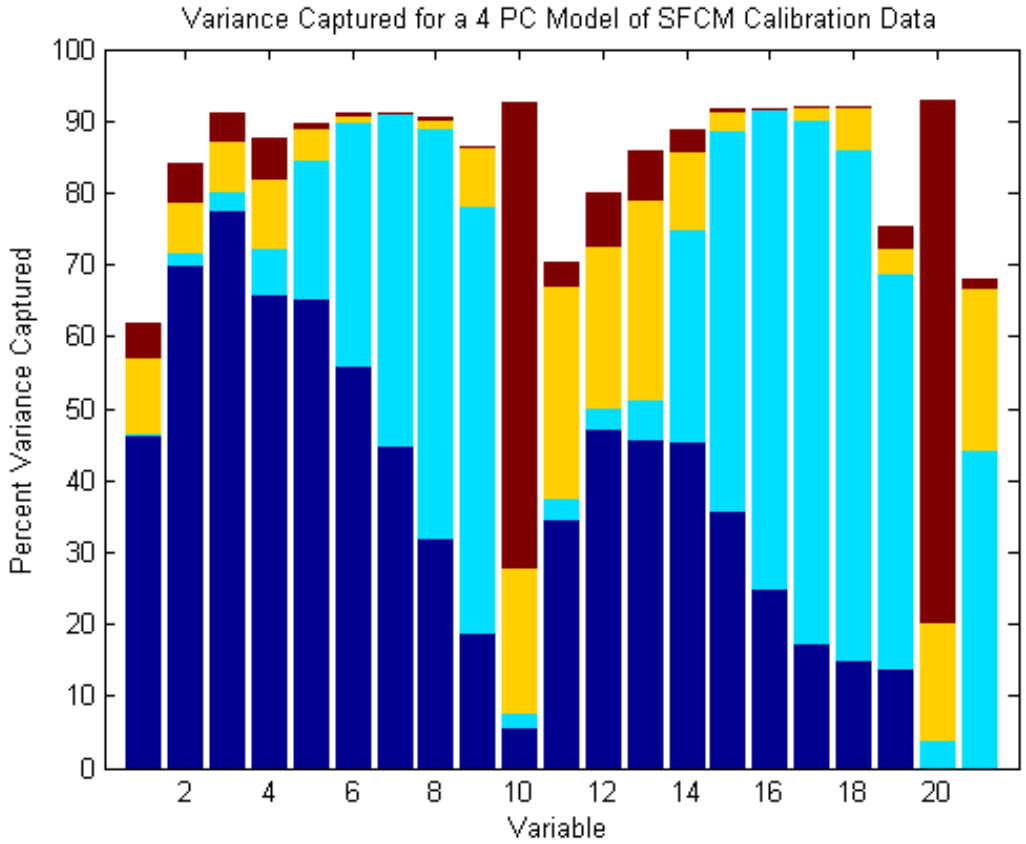
$$R_j^2 = 1 - (\mathbf{e}_j^{\mathrm{T}}\mathbf{e}_j) / (\mathbf{x}_j^{\mathrm{T}}\mathbf{x}_j)$$

The figure shows such a plot.

The different colours show how much of the variance each PC explains (lower part = lower PC).

In this example we note that

- < 75 % of variables 1, 11, 19 and 21 is explained
- PC4 is needed to model variables 10 and 20



6.5.3 Model diagnostics

By model diagnostics we try to evaluate how well the model will match new data *not used* for model building. Because of problems due to overfitting, it is important to find a good trade-off between model fit and predictability.
In the case of PCA

- this trade-off involves the number of principal components
- the main decision tool is *cross-validation* (CV)

Cross-validation

Cross-validation serves two critical functions:

- it enables an assessment of the optimal complexity of a model (# of PCs)
- it allows an evaluation of the performance of a model when it is applied to new data

The basic idea behind cross-validation is to split the data into a

- calibration set, which is used for parameter optimization
- validation set, which is used for structural decisions, i.e. the number of PCs
- evaluation set, which is used for evaluating the performance

Because of the purpose of the various data sets, it is convenient occasionally to refer to them as follows:

- modelling set (data): calibration *and* validation set (data) taken together
- test set (data): validation *or* evaluation set (data)

Usually the data splitting is repeated several times so that many different combinations of calibration, validation and evaluation sets are obtained from the same data.

There are many standardized ways of splitting the data. The choice mainly depends on the type of data we are working with. In the splitting of data, there are two “traps” that should be avoided:

- The *ill-conditioned trap*: if the calibration/modelling set and the test set cover different data "spaces“ (i.e. they cannot be described well by the same model) the CV result will be overly pessimistic.
- The *replicate sample trap*: if the same data is present in both the calibration/modelling set and the test set the CV result will be overly optimistic.

Assuming that

- n is the number of observations in the full data set
- $s \leq n/2$ is the number of data splits for CV tests; in the methods described below, it is also the number of CV tests

some of the main data splitting techniques can be described as follows (in the figures, data with same colour go to the same test set, different colours go to different test sets):

- Leave-one-out: Each single object in the data set is used as a different test set; this means that n tests are performed.
  - computationally demanding for large n, should have $n < 20$
  - significant risk for overly optimistic or pessimistic results

- Venetian blinds: Each test set is determined by selecting every s:th object in the data set, starting at objects numbered $1 \ldots s$.
  - generally safe to use with many objects in random order
  - for time-series data, the method can be useful for estimating errors from non-temporal sources (errors not varying in time); risk for overly optimistic results if s is "small"
  - significant risk for overly optimistic results if data are repetitive

- Contiguous blocks: Each test sets is determined by selecting contiguous blocks of n / s objects in the data set, starting at object number 1.
  - generally safe to use with many objects in random order
  - for time-series data and batch data, the method can be useful for assessing temporal stability and batch-to-batch predictability
  - good way to avoid overly optimistic results
  - significant danger of overly pessimistic results for repetitive data

Number of principal components

The number of PCs can be selected e.g. by means of a scree plot (section 6.2.5) but a better method generally is to use

- cross-validation by splitting the modelling data set $\mathbf{X}$ into a
  - calibration set $\mathbf{X}_{\text{cal}}$
  - validation set $\mathbf{X}_{\text{val}}$

For a given number of PCs, which is varied from 1 to some upper limit (e.g. 20),

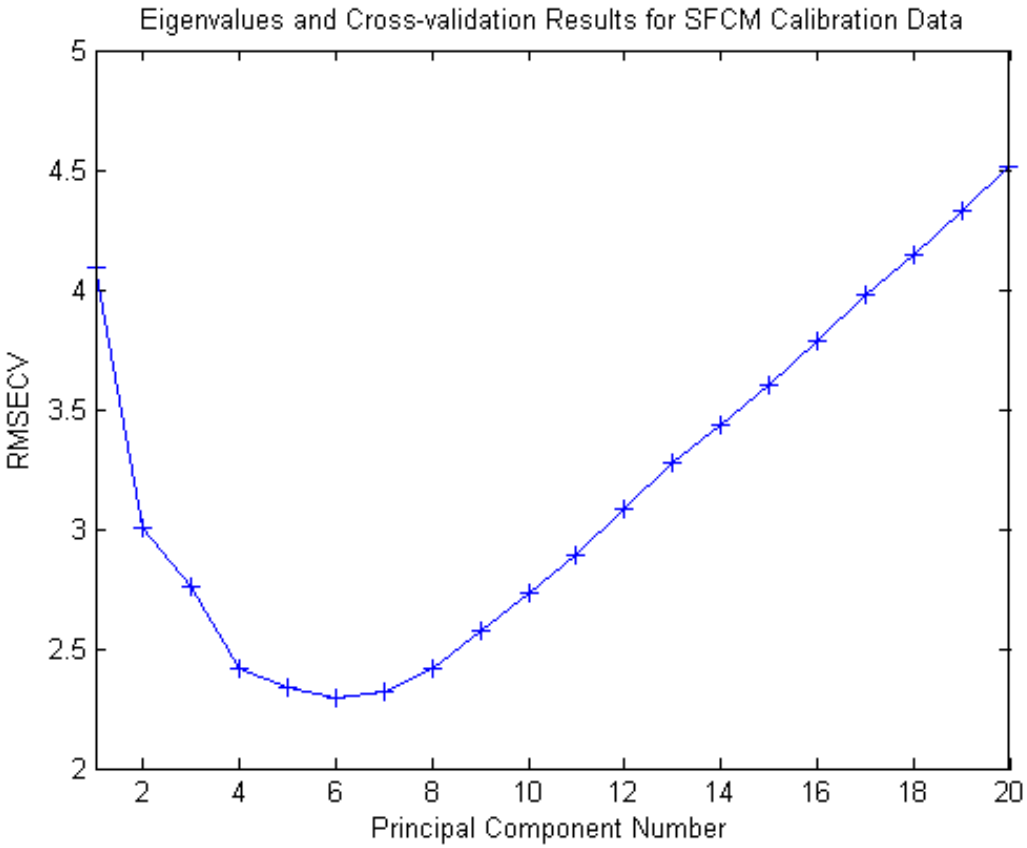
- the loading matrix $\mathbf{P}_{\text{cal}}$ is determined from the calibration set $\mathbf{X}_{\text{cal}}$
- the validation error is given by $\mathbf{E}_{\text{val}} = \mathbf{X}_{\text{val}}(\mathbf{I} - \mathbf{P}_{\text{cal}}\mathbf{P}_{\text{cal}}^{\text{T}})$

The validation error matrix $\mathbf{E}_{\text{val}}$ can be studied in the same way as the residual matrix $\mathbf{E}$ in sections 6.5.1. and 6.5.2.

A useful measure for selection of the number of PCs is the "root-mean-square error of cross-validation", RMSECV. When the cross-validation is repeated s times, the error can be defined

$$E_{\text{RMS}}^{\text{CV}} = \sqrt{\frac{1}{np}\sum_{m=1}^{s}\sum_{i=1}^{n/s}\sum_{j=1}^{p} e_{ij}^2}$$

where n/s is assumed to be an integer and e_{ij} is element (i, j) of each of the s $\mathbf{E}_{\text{val}}$ matrices.



The figure shows a plot of RMSECV vs. the number of PCs. Because a large # of PCs results in overfitting, the plot will have a minimum for some # of PCs. This number, or a slightly smaller number if the minimum is flat, is the indicated choice. In this case, 4 PCs is probably the best choice.

Performance evaluation

To evaluate the performance of the PC model, a separate data set than the modelling data set is needed. This data set, the

- evaluation data set

is used for cross-validation of the modelling data set in exactly the same way as the validation data set was used for cross-validation of the calibration data set.

For each evaluation, this gives a matrix of prediction errors, $\mathbf{E}_{\mathbf{pred}}$, which can be calculated by

$$\mathbf{E}_{\text{pred}} = \mathbf{X}_{\text{eval}}(\mathbf{I} - \mathbf{P}\mathbf{P}^{\text{T}})$$

A measure of the goodness of the model is given by

$$Q^2_{\text{pred}} = 1 - \sum_{j=1}^{p} \mathbf{e}_j^{\text{T}}\mathbf{e}_j \Big/ \sum_{j=1}^{p} \mathbf{x}_j^{\text{T}}\mathbf{x}_j$$

where $\mathbf{e}_j$ is the j:th column in $\mathbf{E}_{\mathbf{pred}}$ and $\mathbf{x}_j$ is the j:th column in $\mathbf{X}_{\text{eval}}$.

It depends on the application what a good Q^2_{pred} value is, but generally

- $Q^2_{\text{pred}} > 0.5$ is good, $Q^2_{\text{pred}} > 0.9$ is excellent

Basically all diagnostic tests treated above, such as

- scores plots and Hotelling's T^2 statistics
- residual plots and Q statistics
- biplots
- explained/unexplained variance plots

can be used to analyse the modelling set and the evaluation set together.