Introduktion til **Data Mining**

Lektion 1

Torben Tvedebrink
tvede@math.aau.dk

Institut for Matematiske Fag

AALBORG UNIVERSITY
DENMARK

Oversigt

Torben Tvedebrink
tvede@math.aau.dk

Formalia

Kursusholder: Torben Tvedebrink [tvede@math.aau.dk]

Antal kursusgange: 10 lektioner, 10 arbejd-selv moduler

Eksamen: 'Take home'-eksamen (varrighed 56 timer)

Litteratur:
"Top Ten Algorithms in Data Mining" edt. Wu og Kumar

"Elements of Statistical Learning" Hastie, Tibshirani og Friedman

"An Introduction to Statistical Learning"
James, Witten, Hastie og Tibshirani

Software: R

Moodle: Data Mining (MATØK8, MAT8)

Torben Tvedebrink
tvede@math.aau.dk

Arbejdsform

Lektioner: (4 timer)
Forelæsning med opgaveregning af med opgaver af mindre opfang. Strukturen forsøges at være 3×30 min forelæsning.

Én gruppe studerende vil få til opgave at fremlægge opgaverne fra forrige selvstudie session (ca. 15 min).

[Jeg vil være til stede hele tiden]

Selvstudie sessions: (4 timer)
Arbejd selvstændigt med analyser af diverse datasæt ved hjælp af relevant algoritme/metode

[Jeg vil ofte være på mit kontor]

Torben Tvedebrink
tvede@math.aau.dk

Skema

	Man	Tirs	Ons	Tors	Fre	Emne
Uge 6			L1		S1	C4.5
Uge 7						
Uge 8			L2		S2	K-means
Uge 9			L3		S3	SVM
Uge 10						
Uge 11			L4		S4	Apriori
Uge 12			L5		S5	EM
Uge 13			L6		S6	PageRank
Uge 14			L7		S7	AdaBoost
Uge 15						
Uge 16			L8		S8	kNN
Uge 17			L9		S9	Naïve Bayes
Uge 18			L10		S10	CART

Torben Tvedebrink
tvede@math.aau.dk

Studieordning
Uddrag

- **Knowledge**
  - understand computer intensive techniques, CV and bootstrap. Can account for the variance-bias trade-off
  - know of various methods for visualising high-dim. data
  - know the difference between classification and regression. Classification methods relying on trees, prototype methods and Bayes classifiers
  - know of various supervised and unsupervised methods within statistical learning
  - know of association rule methods for the analysis of transaction data
  - can perform link mining for network e.g. internet pages
  - have knowledge of methods to do hierarchical and partitioning cluster analysis
  - know of model averaging, bagging and boosting
- Skills
- Competencies

Torben Tvedebrink
tvede@math.aau.dk

Studieordning
Uddrag

- Knowledge
- **Skills**
  - are able to identify and apply a relevant data mining algorithm in a specific context
  - can identify and discuss weaknesses and strengths of different data mining algorithm in relation to a specific analysis task
  - can interpret and communicate the results of a given data mining analysis to non- specialists
- Competencies

Torben Tvedebrink
tvede@math.aau.dk

Studieordning
Uddrag

- Knowledge
- Skills
- **Competencies**
  - have the ability to survey potentials and limitations of different data mining software packages
  - have the understanding to choose and apply specific software meeting user demands

Torben Tvedebrink
tvede@math.aau.dk

Hvad er Data Mining?

- Overskue og visualisere store mængder af data (store datasæt)
- Ekstraktion/identifikation af strukturer
- Klassificering af objekter
- Finde elementer med relevant information
- ...

Torben Tvedebrink
tvede@math.aau.dk

Larry Wasserman, Prof, Carnegie Mellon

Om forskellen mellem SL, ML og DM:

"The short answer is: None. They are [. . .] concerned with the same question: how do we learn from data?"

David R. Cox

The sequence question–data–analysis is emphasized here *[in statistical inference]*. In data mining the sequence may be closer to data–analysis–question.

Torben Tvedebrink
tvede@math.aau.dk

Trillinger
Statistical Learning, Machine Learning og Data Mining

Statistical Learning Tættere forbundet med statistisk inferens, e.g. hypothese tests, CI og estimatorer (ofte i lavere dimension).

Machine Learning Udspringer fra datalogi i ønsket om at konstruere algoritmer og systemer som kan *lære* fra data.

Data Mining "Mining the data"; At lede efter værdifuld indsigt i større data mængder/databaser.

Overordnet set bruges de samme teknikker i de tre discipliner, men fokus og ikke mindst baggrunden for de tre områder er skyld i at der findes tre tæt relaterede domæner.

Torben Tvedebrink
tvede@math.aau.dk

Mere om forskelle
Terminologi

Statistics	**Machine Learning**
Estimation	Learning
Classifier	Hypothesis
Data point	Example/Instance
Regression	Supervised Learning
Classification	Supervised Learning
Covariate	Feature
Response	Label
... use R	... use Matlab

her i kurset bruger vi **R**!

Torben Tvedebrink
tvede@math.aau.dk

Algoritmer og metoder

Ét kapitel i vores primære kilde per metode

1. C4.5 *[Classification]*
2. *K*-Means *[Classification]*
3. Support Vector Machines *[Statistical Learning]*
4. Apriori *[Association mining]*
5. EM-algorithm *[Statistical Learning]*
6. PageRank (Google) *[Link mining]*
7. AdaBoost *[Boosting]*
8. *k*-Nearest Neighbors *[Classification]*
9. Naïve Bayes *[Classification]*
10. CART (and random forest) *[Classification and Bagging]*

Torben Tvedebrink
tvede@math.aau.dk

Data

Vi tænker ofte på vores data i formen af en matrix, X, med p variable målt for hvert af vores n observationer,

$$\underset{n\times p}{X} = \{X_{ij}\}_{\substack{i=1,\ldots,n,\\ j=1,\ldots,p}}$$

hvor x_{ij} er den j'te variabel for den i'te observation. I.e. den i'te række er den p-dim rækkevektor $\boldsymbol{x}$.

Vi benævner typisk X som features, attributter, forklarende eller uafhængige variable.

Responsvariablen skrives typisk som $\boldsymbol{Y}$, hvor $\boldsymbol{Y}$ kan være kategorisk eller numerisk alt efter problemstillingen.

Vores primære kilde (Top Ten Algorithms in DM) har bidrag fra forskellige forfattere. Derfor benytter visse kapitler anden notation, men jeg vil bruge $(\boldsymbol{Y}, X)$-notationen i slides.

Torben Tvedebrink
tvede@math.aau.dk

p ofte meget stor - endda større end n

Indenfor data mining er p ofte meget stor (fx. 1000, 10^5, ...). Dvs. rigtig mange kovariater/features/attributer/...

Afhængigt af typen af data/problemstilling kan n være meget større end p (fx. databaser) eller meget mindre end p (fx. genomics, n typisk 100-200 og antal genetiske markører, p, ofte er $10^5 - 10^6$).

Torben Tvedebrink
tvede@math.aau.dk

Visualisering af data

Et af de absolut første skidt i analyser af data er at opnå (hvis muligt) en forståelse af data vha. simple plots.

"Man skal tegne før man kan regne"

(Thorvald Nicolai Thiele, 1838–1910)

De simpleste valg af plots er histogrammer eller boxplots for hver variabel (evt. grupperet efter andre variable) og parvise scatterplots.

Torben Tvedebrink
tvede@math.aau.dk

Eksempel data: Crabs data

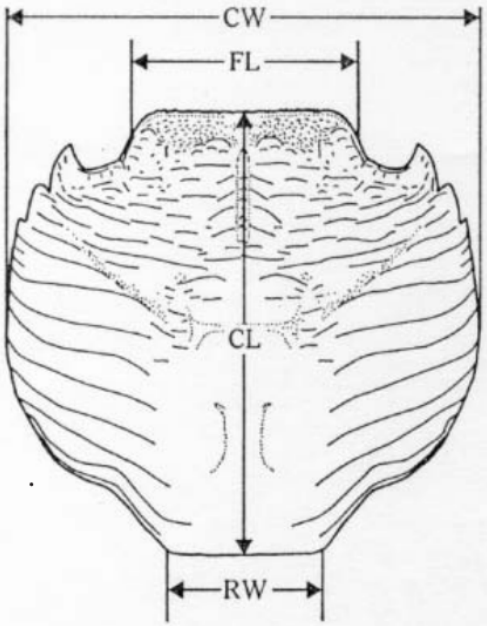


Fig. 1. Dorsal view of carapace of *Leptograpsus*, showing measurements taken. *FL*, width of frontal region just anterior to frontal tubercles. *RW*, width of posterior region. *CL*, length along midline. *CW*, maximum width. The body depth was also measured; in females but not in males the abdomen was first displaced.

Følgende er eksempler plottet for `crabs`-data fra `MASS`:

```
> head(crabs)
  sp sex index   FL  RW   CL   CW  BD
1  B   M     1  8.1 6.7 16.1 19.0 7.0
2  B   M     2  8.8 7.7 18.1 20.8 7.4
3  B   M     3  9.2 7.8 19.0 22.4 7.7
4  B   M     4  9.6 7.9 20.1 23.1 8.2
5  B   M     5  9.8 8.0 20.3 23.0 8.2
6  B   M     6 10.8 9.0 23.0 26.5 9.8
```


Torben Tvedebrink
tvede@math.aau.dk

Basale plots i R

Plots used in lecture 1

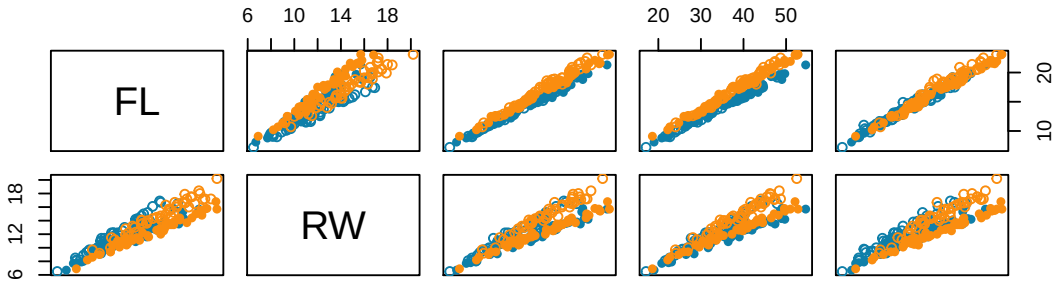
Base graphics

```
library(MASS)
data(crabs)

head(crabs)
```

```
##   sp sex index   FL  RW   CL   CW  BD
## 1  B   M     1  8.1 6.7 16.1 19.0 7.0
## 2  B   M     2  8.8 7.7 18.1 20.8 7.4
## 3  B   M     3  9.2 7.8 19.0 22.4 7.7
## 4  B   M     4  9.6 7.9 20.1 23.1 8.2
## 5  B   M     5  9.8 8.0 20.3 23.0 8.2
## 6  B   M     6 10.8 9.0 23.0 26.5 9.8
```

```
num_cols <- c("FL","RW","CL","CW","BD")
```

```
ccol <- function(sp) ifelse(sp=="B","#0f7fa9","#fa8d0f")
cpch <- function(sx) 1 + 15*(crabs$sex=="M")

pairs(crabs[num_cols], col=ccol(crabs$sp), pch=cpch(crabs$sex))
```



Torben Tvedebrink
tvede@math.aau.dk

Principal Components Analysis (PCA)

PCA kan benyttes til at reducere dimensionen af ens data ved at identificere lineære kombinationer af variablene som har størst varians.

Torben Tvedebrink
tvede@math.aau.dk

Principal Components Analysis (PCA)

PCA kan benyttes til at reducere dimensionen af ens data ved at identificere lineære kombinationer af variablene som har størst varians.

Lad X være vores data matrix. Vi antager at X er centreret, dvs. vi har fratrukket gennemsnittet, $n^{-1}\mathbf{1}_n^\top X$, således at $X := X - n^{-1}\mathbf{1}_n\mathbf{1}_n^\top X$.

Sample covarians matricen, C, skrives som $p \times p$-matricen

$$C = n^{-1}X^\top X$$

Vi ønsker at finde projektionen $\boldsymbol{v}_1$ således variansen af $X\boldsymbol{v}_1$ er størst mulig... Hvorfor?

De efterfølgende projektioner, $X\boldsymbol{v}_i,\ i = 2, \ldots, p$, ønskes ukorrelerede med $X\boldsymbol{v}_j,\ j = 1, \ldots, i-1$ samt med størst mulig variationen 'langs' $\boldsymbol{v}_i$.

Torben Tvedebrink
tvede@math.aau.dk

Find PCA

Bemærk at

$$\mathbb{V}(X\boldsymbol{v}_1) = \boldsymbol{v}_1^\top \mathbb{V}(X)\boldsymbol{v}_1, \qquad \textit{Hvorfor?}$$

dvs. at variansen kan gøres vilkårligt stor. En mulig betingelse er derfor at $\|\boldsymbol{v}_1\|^2 = \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1$.

Torben Tvedebrink
tvede@math.aau.dk

Find PCA

Bemærk at

$$\mathbb{V}(X\boldsymbol{v}_1) = \boldsymbol{v}_1^\top \mathbb{V}(X)\boldsymbol{v}_1, \qquad \textit{Hvorfor?}$$

dvs. at variansen kan gøres vilkårligt stor. En mulig betingelse er derfor at $\|\boldsymbol{v}_1\|^2 = \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1$.

Det svarer således til at maksimere følgende mht. $\boldsymbol{v}_1$:

$$\max_{\boldsymbol{v}_1} \boldsymbol{v}_1^\top C\boldsymbol{v}_1 - \lambda(\boldsymbol{v}_1^\top \boldsymbol{v}_1 - 1)$$

Hvilket medfører at vi skal løse $(C - \lambda I)\boldsymbol{v}_1 = \mathbf{0}$ mht. $\boldsymbol{v}_1$ og λ.

– Hvad minder det om?

Torben Tvedebrink
tvede@math.aau.dk

Geometrisk/intuitivt argument

Vi ønsker $\mathbb{V}(X\boldsymbol{v}_1)$ størst mulig hvor $\|\boldsymbol{v}_1\| = 1$:

$$\max_{\boldsymbol{v}_1 : \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1} \mathbb{V}(X\boldsymbol{v}_1) = \max_{\boldsymbol{v}_1 : \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1} \boldsymbol{v}_1^\top \mathbb{V}(X)\boldsymbol{v}_1 = \max_{\boldsymbol{v}_1 : \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1} \boldsymbol{v}_1^\top C \boldsymbol{v}_1,$$

hvor C er kovariansmatricen for X. *Hvad ved vi om C?*

Torben Tvedebrink
tvede@math.aau.dk

Geometrisk/intuitivt argument

Vi ønsker $\mathbb{V}(X\boldsymbol{v}_1)$ størst mulig hvor $\|\boldsymbol{v}_1\| = 1$:

$$\max_{\boldsymbol{v}_1 : \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1} \mathbb{V}(X\boldsymbol{v}_1) = \max_{\boldsymbol{v}_1 : \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1} \boldsymbol{v}_1^\top \mathbb{V}(X)\boldsymbol{v}_1 = \max_{\boldsymbol{v}_1 : \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1} \boldsymbol{v}_1^\top C \boldsymbol{v}_1,$$

hvor C er kovariansmatricen for X. *Hvad ved vi om C?*

Vi har således at C er positiv semi-definit, således at vi kan dekomponere C i en ortonormal matrix U og en diagonal matrix Λ med egenværdier $\lambda_1 > \lambda_2 > \cdots > \lambda_p$:

$$C = U^\top \Lambda U$$

Torben Tvedebrink
tvede@math.aau.dk

Rotation af data

Dvs. vi har

$$\max_{\boldsymbol{v}_1 : \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1} \mathbb{V}(X\boldsymbol{v}_1) = \max_{\boldsymbol{v}_1 : \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1} \boldsymbol{v}_1^\top U^\top \Lambda U \boldsymbol{v}_1,$$

hvor vi definerer $\tilde{\boldsymbol{v}}_1 = U\boldsymbol{v}_1$. *Hvad er normen af* $\tilde{\boldsymbol{v}}_1$*?*

Torben Tvedebrink
tvede@math.aau.dk

Rotation af data

Dvs. vi har

$$\max_{\boldsymbol{v}_1 : \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1} \mathbb{V}(X\boldsymbol{v}_1) = \max_{\boldsymbol{v}_1 : \boldsymbol{v}_1^\top \boldsymbol{v}_1 = 1} \boldsymbol{v}_1^\top U^\top \Lambda U \boldsymbol{v}_1,$$

hvor vi definerer $\tilde{\boldsymbol{v}}_1 = U\boldsymbol{v}_1$. *Hvad er normen af* $\tilde{\boldsymbol{v}}_1$*?*

$$\|\tilde{\boldsymbol{v}}_1\|^2 = \tilde{\boldsymbol{v}}_1^\top \tilde{\boldsymbol{v}}_1 = (U\boldsymbol{v}_1)^\top U\boldsymbol{v}_1 = \boldsymbol{v}_1^\top U^\top U \boldsymbol{v}_1 = \boldsymbol{v}_1^\top \boldsymbol{v}_1 = \|\boldsymbol{v}_1\|^2$$

Dvs. vi har reduceret vores problem til

$$\max_{\tilde{\boldsymbol{v}}_1 : \tilde{\boldsymbol{v}}_1^\top \tilde{\boldsymbol{v}}_1 = 1} \tilde{\boldsymbol{v}}_1^\top \Lambda \tilde{\boldsymbol{v}}_1 = \max_{\tilde{\boldsymbol{v}}_1 : \tilde{\boldsymbol{v}}_1^\top \tilde{\boldsymbol{v}}_1 = 1} \sum_{i=1}^{p} \tilde{v}_{1,i}^2 \lambda_i,$$

hvilket naturligvis er maksimeret for $\tilde{v}_{1,1} = 1$, dvs. $\tilde{\boldsymbol{v}}_1 = \boldsymbol{e}_1$.

Torben Tvedebrink
tvede@math.aau.dk

Rekursiv måde at konstruere efterfølgende PCAer

For at finde næste PCA, $\boldsymbol{v}_2$, er fremgangsmåden tilsvarende, med den tilføjelse at den skal være ukorreleret til de foregående PCAer.

Bemærk at kovariansen mellem $X\boldsymbol{v}_1$ og $X\boldsymbol{v}_2$ er nul. *[Vis]*

Torben Tvedebrink
tvede@math.aau.dk

Rekursiv måde at konstruere efterfølgende PCAer

For at finde næste PCA, $\boldsymbol{v}_2$, er fremgangsmåden tilsvarende, med den tilføjelse at den skal være ukorreleret til de foregående PCAer.

Bemærk at kovariansen mellem $X\boldsymbol{v}_1$ og $X\boldsymbol{v}_2$ er nul. *[Vis]*

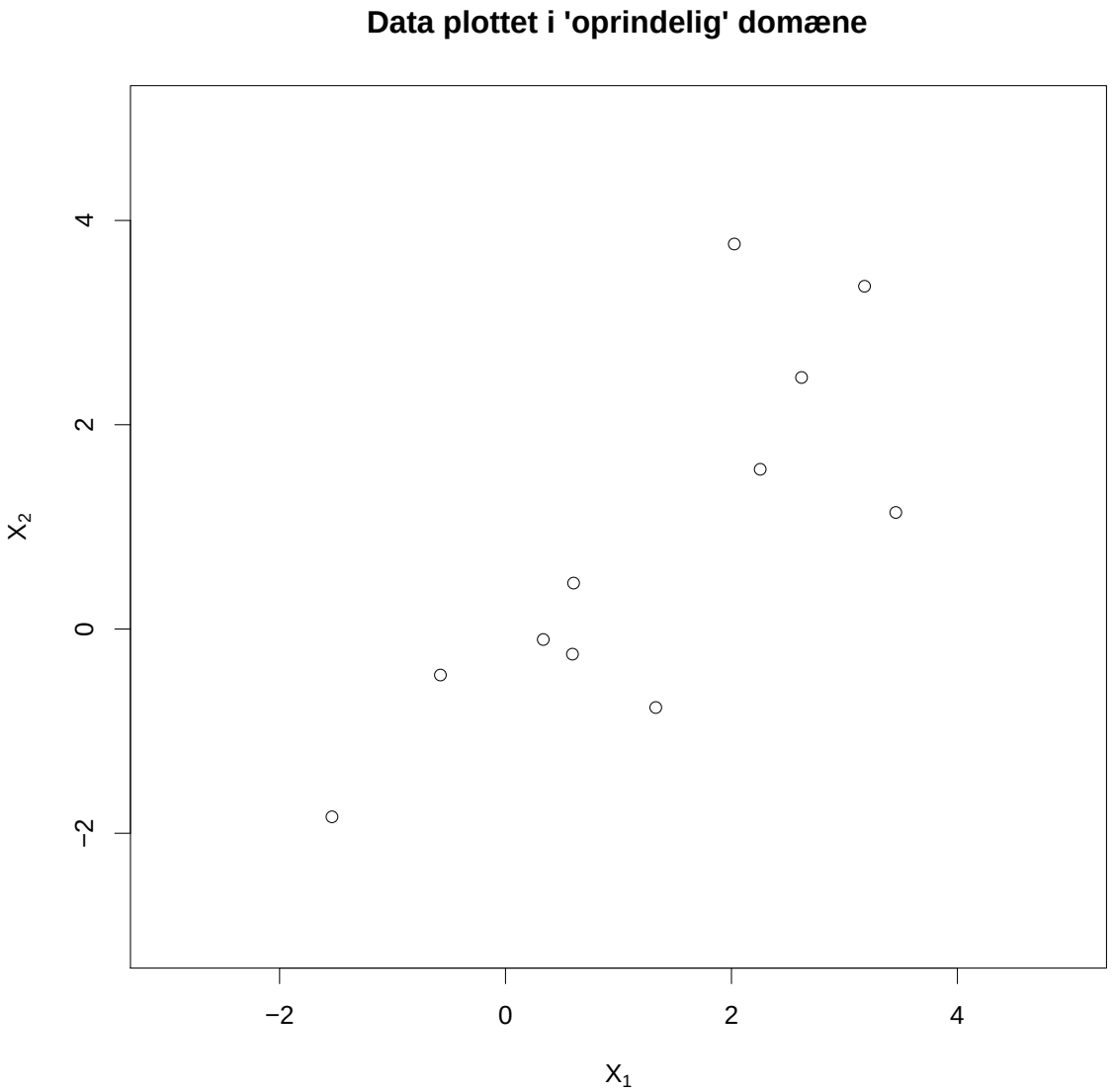
Vi skal således maksimere

$$\max_{\substack{\tilde{\boldsymbol{v}}_2 : \tilde{\boldsymbol{v}}_2^\top \tilde{\boldsymbol{v}}_2 = 1 \\ \tilde{v}_{2,1} = 0}} \tilde{\boldsymbol{v}}_2^\top \Lambda \tilde{\boldsymbol{v}}_2 = \max_{\substack{\tilde{\boldsymbol{v}}_2 : \tilde{\boldsymbol{v}}_2^\top \tilde{\boldsymbol{v}}_2 = 1 \\ \tilde{v}_{2,1} = 0}} \sum_{i=1}^{p} \lambda_i \tilde{v}_{2,i}^2,$$

hvorfor vi vælger $\tilde{v}_{2,2} = 1$, dvs. $\tilde{\boldsymbol{v}}_2 = \boldsymbol{e}_2$.

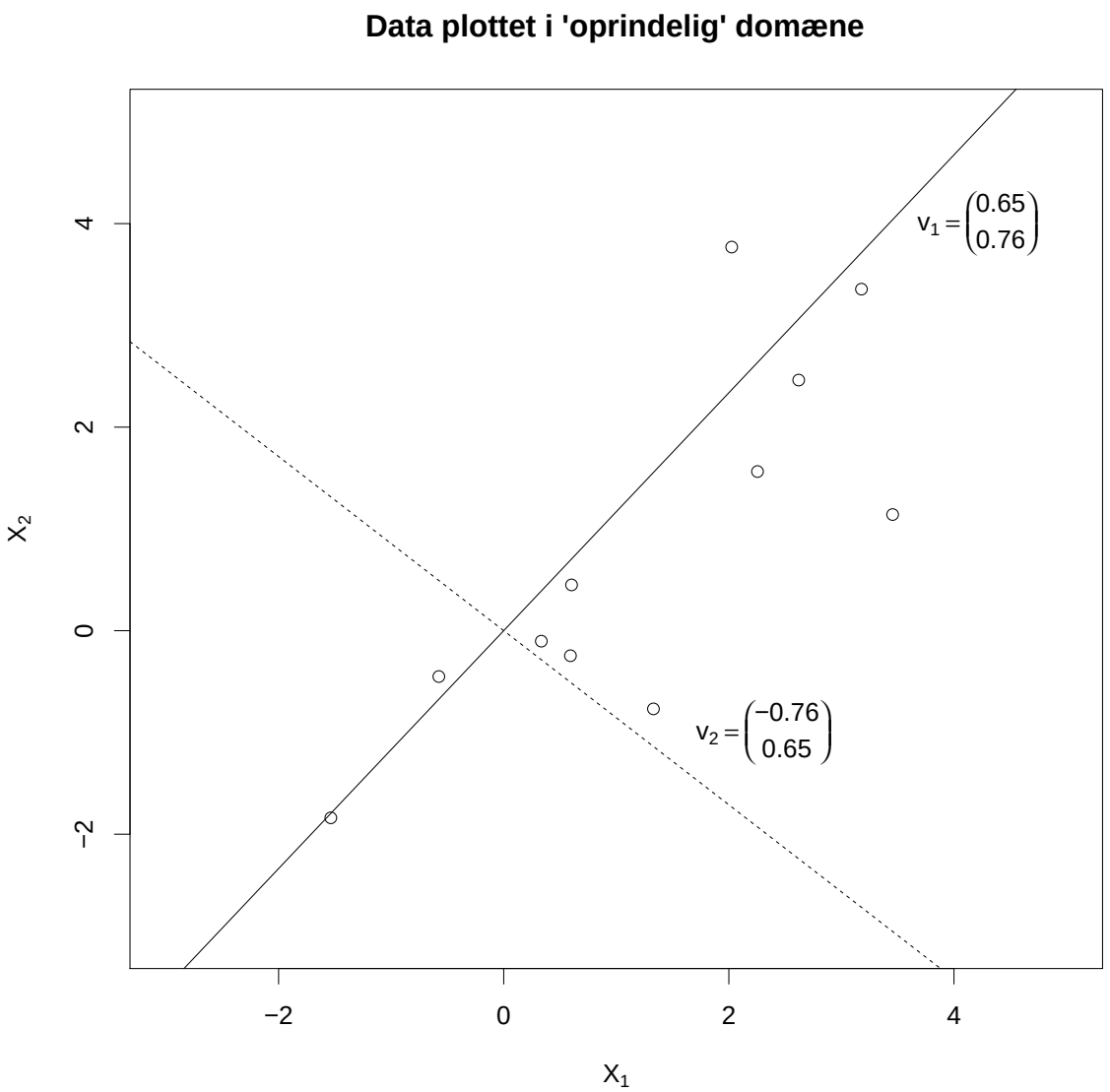
Torben Tvedebrink
tvede@math.aau.dk

PCA
Simpelt eksempel



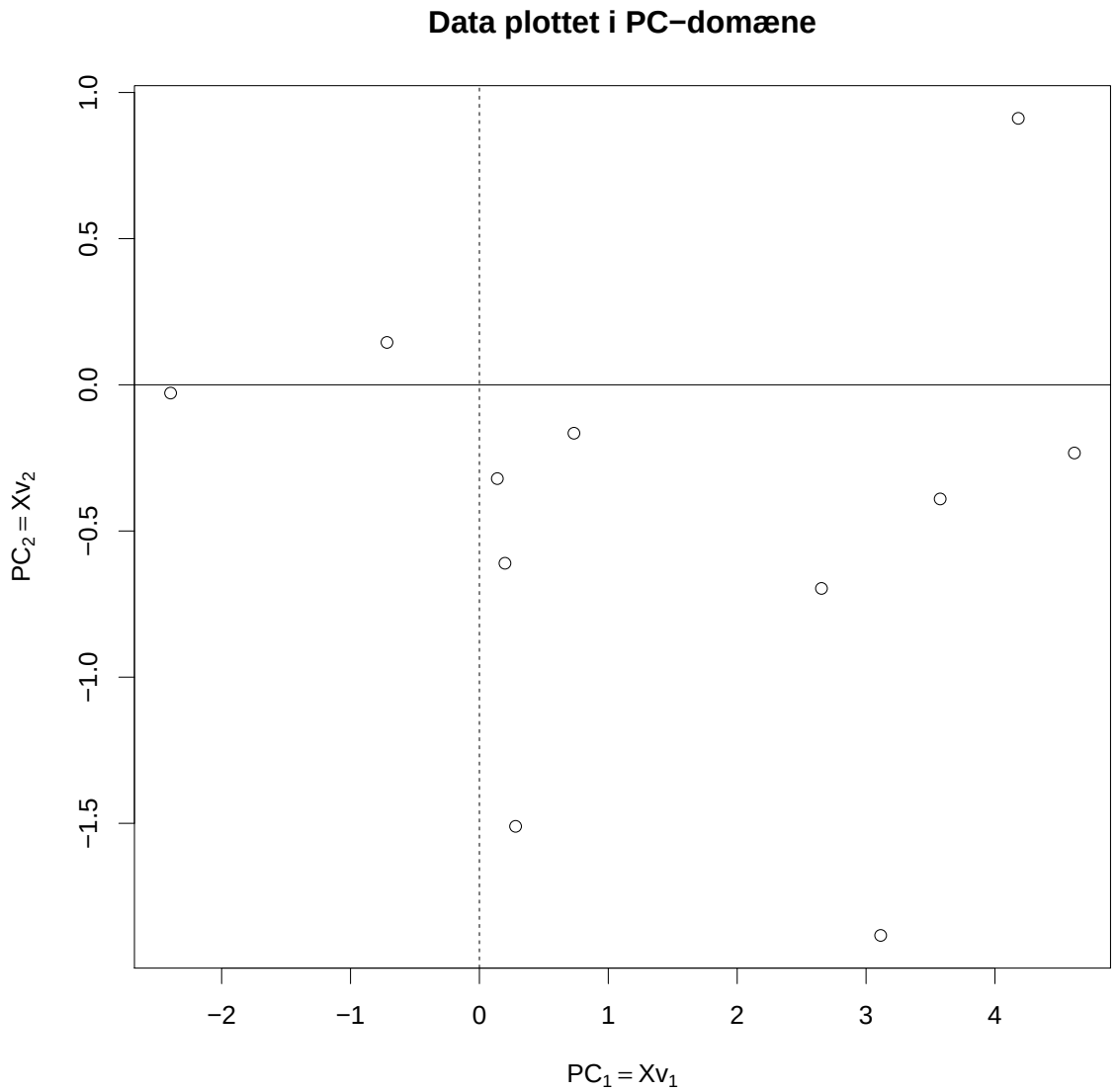
Torben Tvedebrink
tvede@math.aau.dk

PCA
Simpelt eksempel



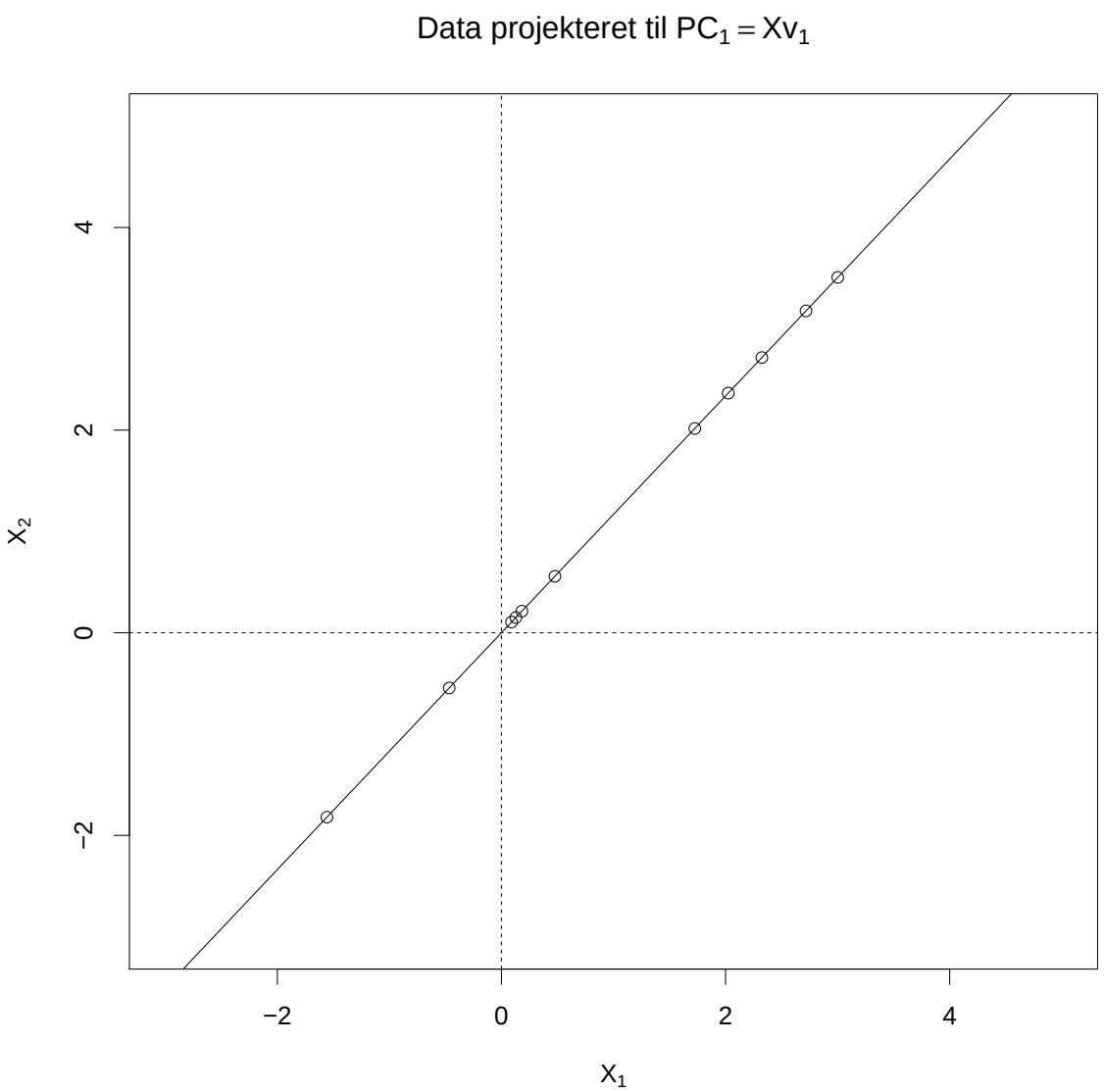
Torben Tvedebrink
tvede@math.aau.dk

PCA
Simpelt eksempel



Torben Tvedebrink
tvede@math.aau.dk

PCA
Simpelt eksempel



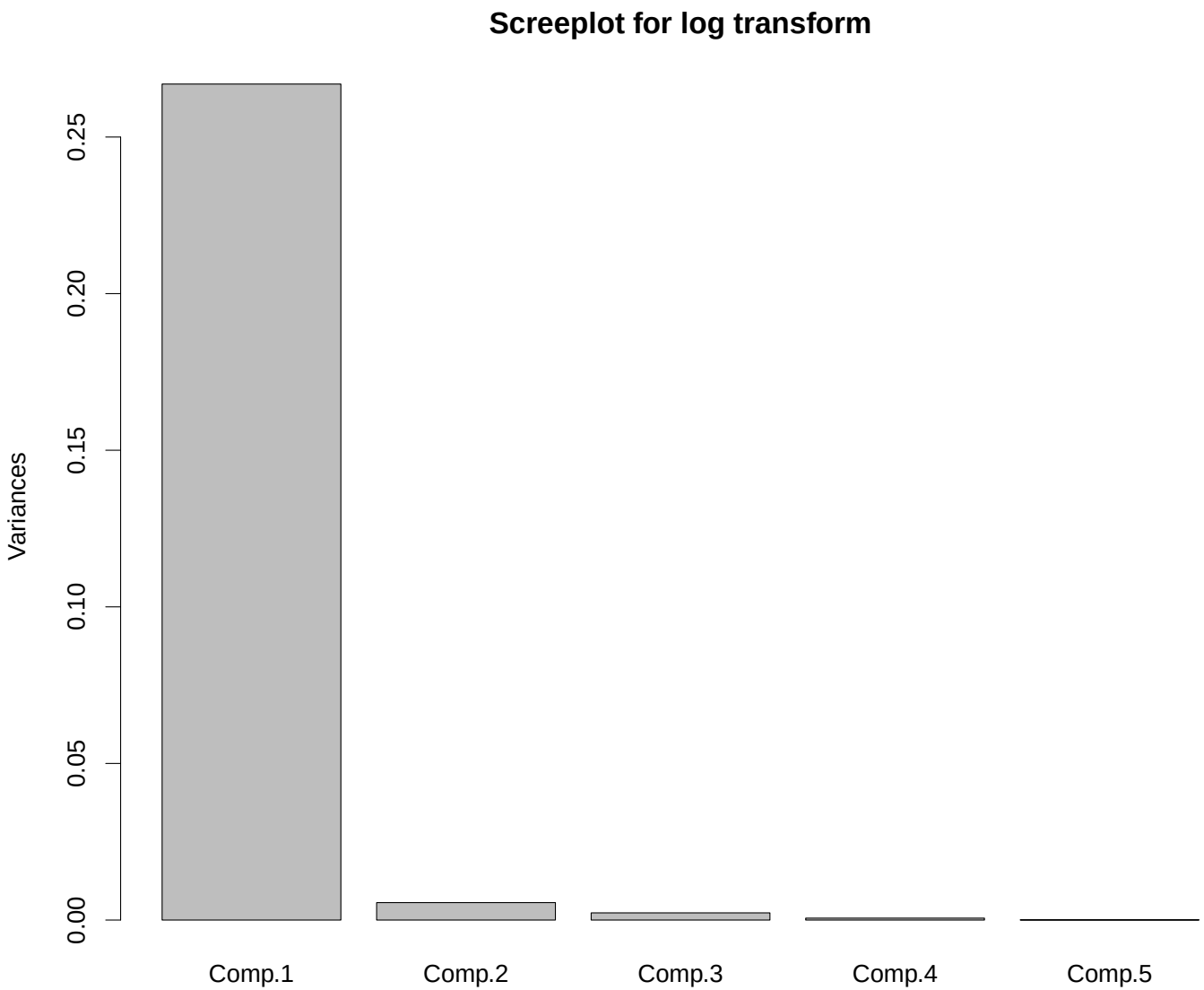
Torben Tvedebrink
tvede@math.aau.dk

PCA eksempel - i R

```
> (crab.pca <- princomp(log(crabs[,-(1:3)])))
Call:
princomp(x = log(crabs[, -(1:3)]))

Standard deviations:
     Comp.1      Comp.2      Comp.3      Comp.4      Comp.5
0.516640451 0.074653581 0.047914392 0.024804021 0.009052189

> loadings(crab.pca)

Loadings:
   Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
FL -0.452 -0.157  0.438  0.752  0.114
RW -0.387  0.911
CL -0.453 -0.204 -0.371        -0.784
CW -0.440        -0.672         0.591
BD -0.497 -0.315  0.458 -0.652  0.136

               Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
SS loadings       1.0    1.0    1.0    1.0    1.0
Proportion Var    0.2    0.2    0.2    0.2    0.2
Cumulative Var    0.2    0.4    0.6    0.8    1.0
```

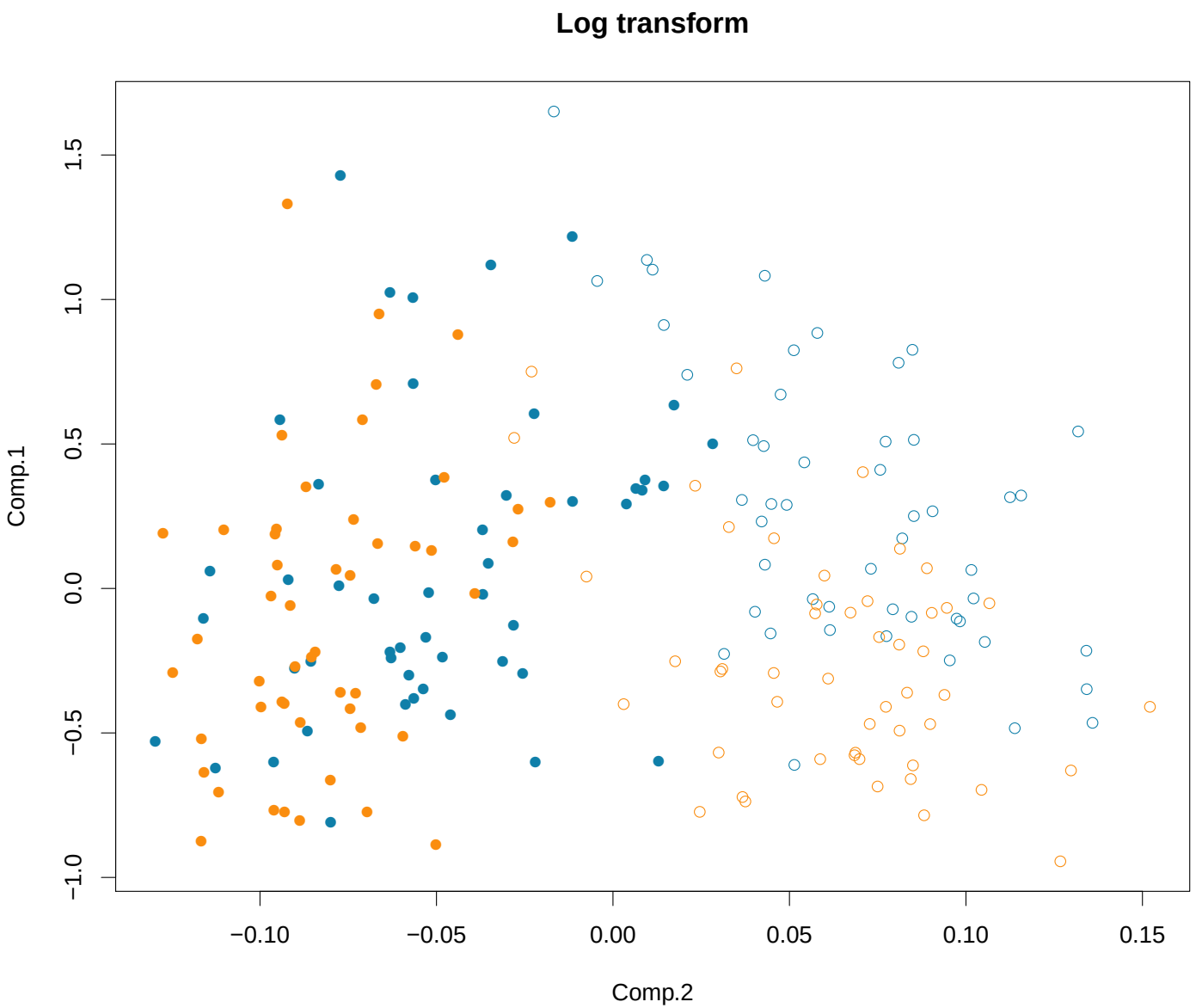

Torben Tvedebrink
tvede@math.aau.dk

PCA eksempel fortsat - plots



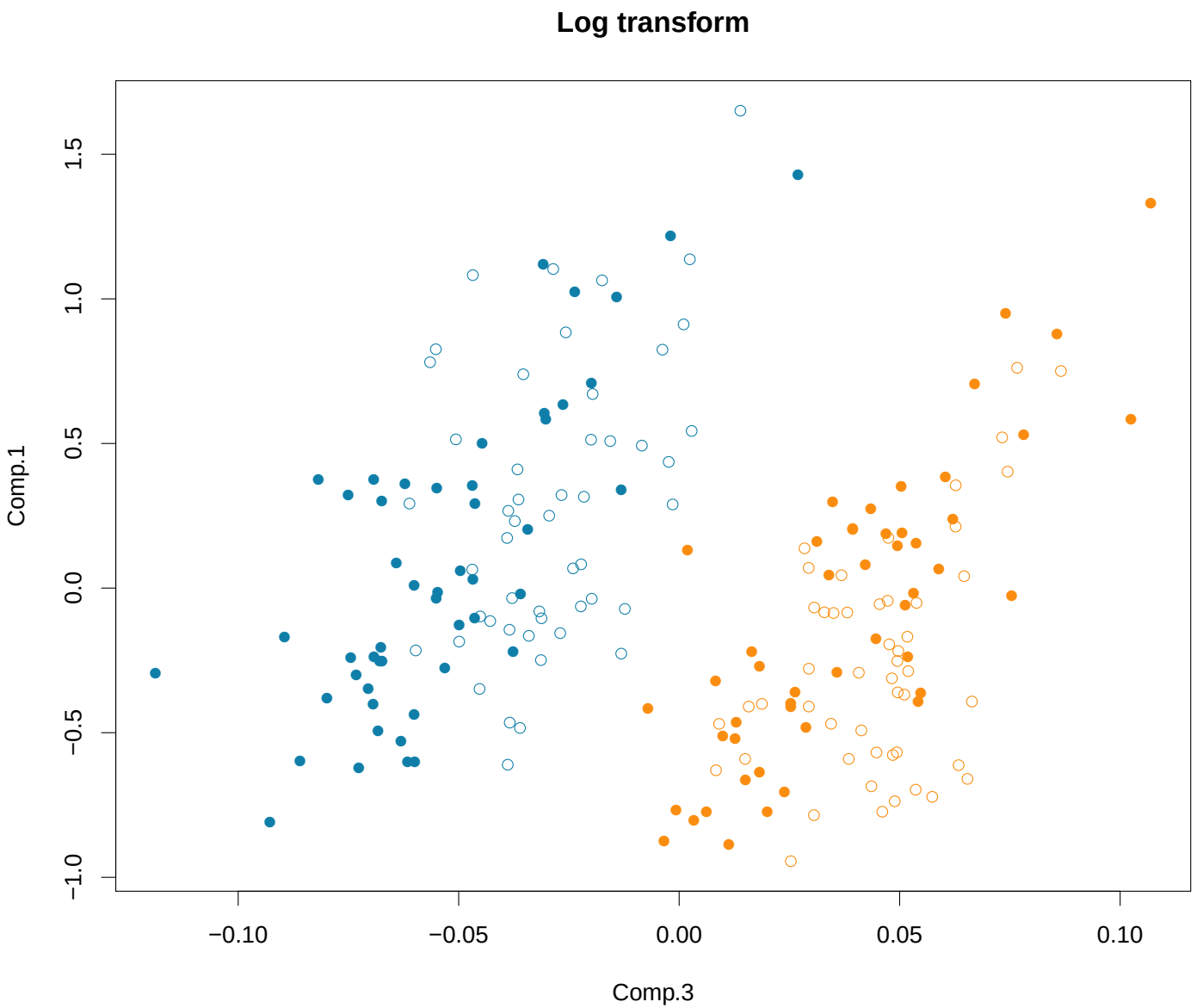
Torben Tvedebrink
tvede@math.aau.dk

PCA eksempel fortsat - plots



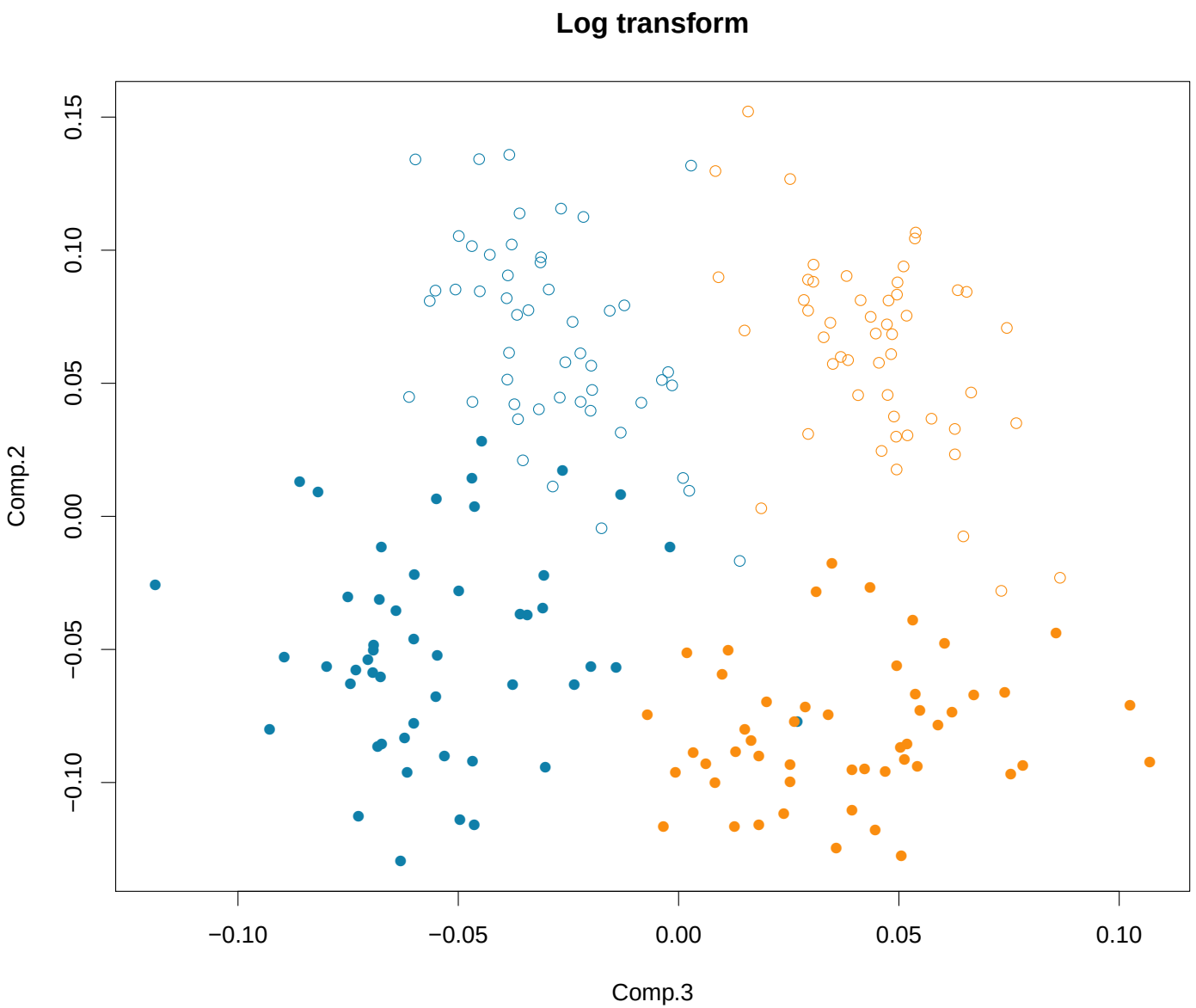
Torben Tvedebrink
tvede@math.aau.dk

PCA eksempel fortsat - plots



Torben Tvedebrink
tvede@math.aau.dk

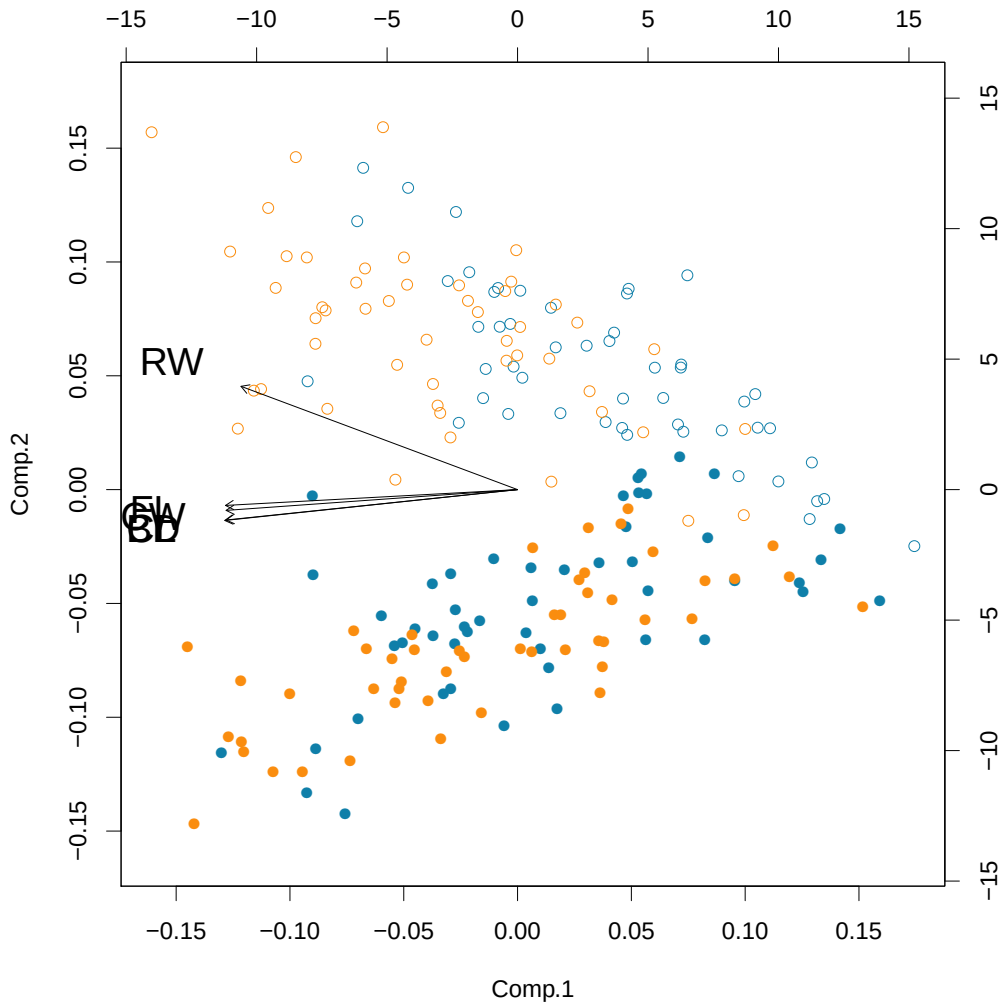
PCA eksempel fortsat - plots



Torben Tvedebrink
tvede@math.aau.dk

Biplots

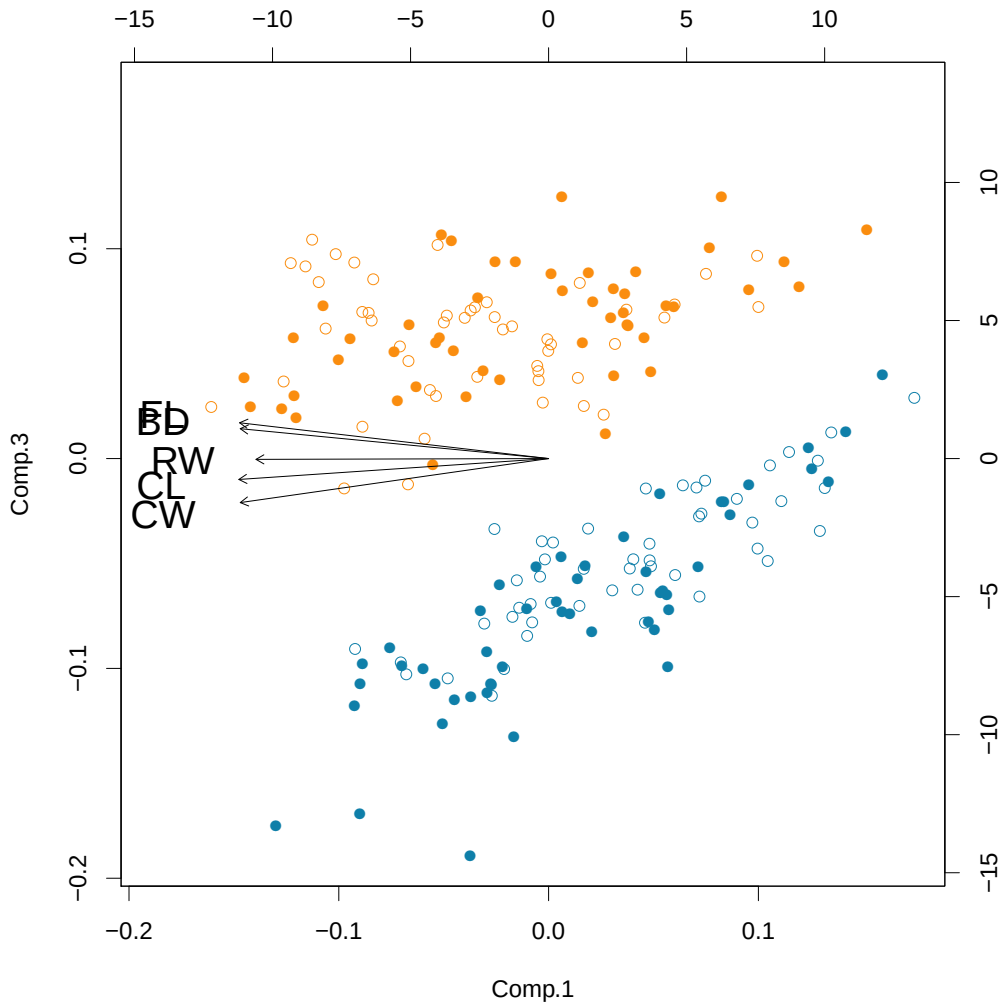
Plot variable og data på samme plot



Torben Tvedebrink
tvede@math.aau.dk

Biplots

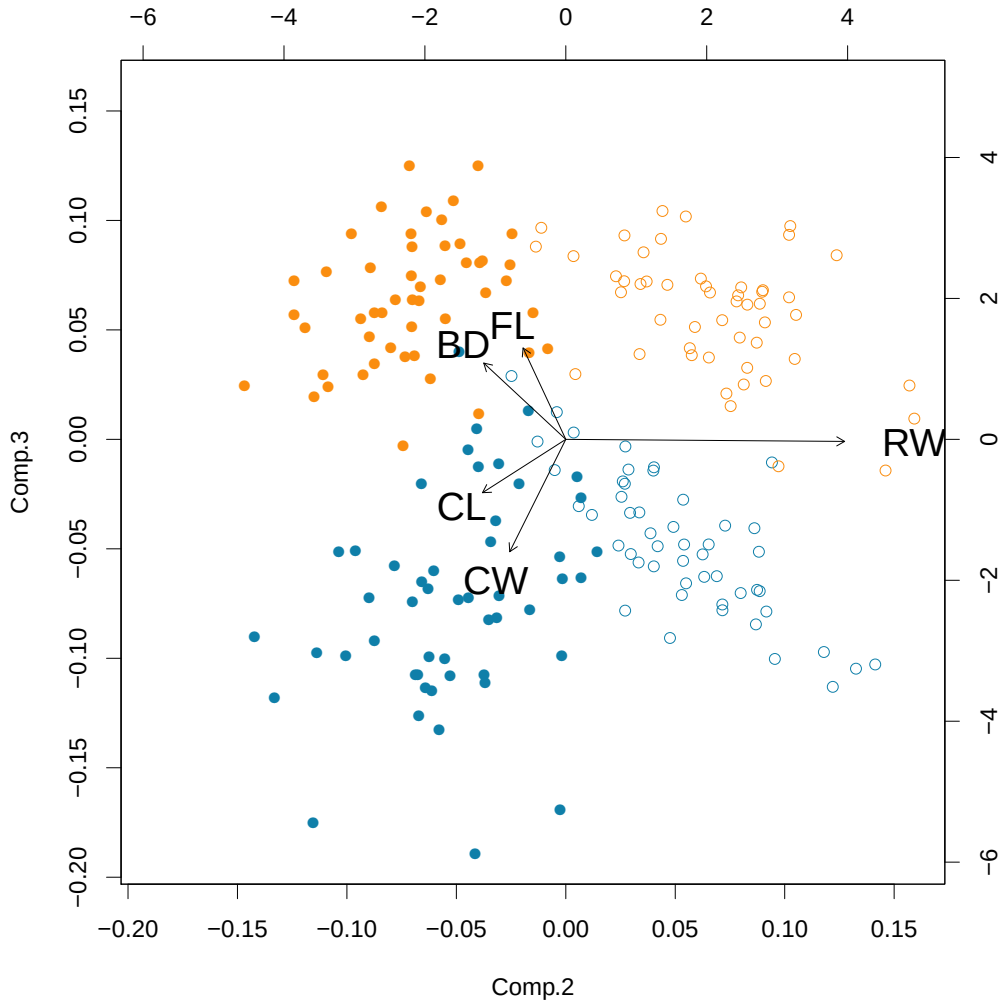
Plot variable og data på samme plot



Torben Tvedebrink
tvede@math.aau.dk

Biplots

Plot variable og data på samme plot



Torben Tvedebrink
tvede@math.aau.dk

Multidimensional Scaling (MDS)

Formål: Ønsker at bevare (så godt som muligt) p-dimentionale afstande mellem data punkter men repræsentere data i lavere dimension k (fx. $k = 2$ eller $k = 3$). Bemærk, afstandene er beregnet mellem højere dimensionale observationer ($p > 3$).

Torben Tvedebrink
tvede@math.aau.dk

Multidimensional Scaling (MDS)

Formål: Ønsker at bevare (så godt som muligt) p-dimentionale afstande mellem data punkter men repræsentere data i lavere dimension k (fx. $k = 2$ eller $k = 3$). Bemærk, afstandene er beregnet mellem højere dimensionale observationer ($p > 3$).

Vi antager at alle variable i X har middelværdi 0.

Torben Tvedebrink
tvede@math.aau.dk

Multidimensional Scaling (MDS)

Formål: Ønsker at bevare (så godt som muligt) p-dimentionale afstande mellem data punkter men repræsentere data i lavere dimension k (fx. $k = 2$ eller $k = 3$). Bemærk, afstandene er beregnet mellem højere dimensionale observationer ($p > 3$).

Vi antager at alle variable i X har middelværdi 0.

Lad $B = XX^\top$, hvor X er $n \times p$-data matrix. Den euklidiske afstand i mellem to observationer (rækker i X)

$$d_{ij}^2 = d^2(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{k=1}^{p} (x_{ik} - x_{jk})^2$$

kan bestemmes vha B, idet $B = [b]_{ij} = \sum_{k=1}^{p} x_{ik} x_{jk}$:

$$d_{ij}^2 = b_{ii} + b_{jj} - 2b_{ij}$$

Torben Tvedebrink
tvede@math.aau.dk

MDS - fortsat

Idet søjler i X har sum 0, ved har vi ligeledes $\sum_{i=1}^{n} b_{ij} = \sum_{j=1}^{n} b_{ij} = 0$. Derfor

$$\sum_{i=1}^{n} d_{ij}^2 = \sum_{i=1}^{n} b_{ii} + b_{jj} - 2b_{ij} = \mathrm{tr}(B) + nb_{jj}$$

$$\sum_{j=1}^{n} d_{ij}^2 = \sum_{j=1}^{n} b_{ii} + b_{jj} - 2b_{ij} = \mathrm{tr}(B) + nb_{ii}$$

$$\sum_{j=1}^{n}\sum_{i=1}^{n} d_{ij}^2 = \sum_{j=1}^{n}\sum_{i=1}^{n} b_{ii} + b_{jj} - 2b_{ij} = 2n\,\mathrm{tr}(B)$$

Torben Tvedebrink
tvede@math.aau.dk

Vi samler ledende

Vi havde fra tidligere at $d_{ij}^2 = b_{ii} + b_{jj} - 2b_{ij}$. Forrige slide viste at

$$b_{jj} = n^{-1}\left(\sum_i d_{ij}^2 - \mathrm{tr}(B)\right), \; b_{ii} = n^{-1}\left(\sum_j d_{ij}^2 - \mathrm{tr}(B)\right)$$

$$\mathrm{tr}(B) = (2n)^{-1}\sum_{i,j} d_{ij}^2$$

B kan således genskabes ud fra viden om parvise afstande d_{ij}^2:

$$b_{ij} = \frac{1}{2n}\left\{\sum_{i=1}^n d_{ij}^2 + \sum_{j=1}^n d_{ij}^2 - \sum_{i,j} d_{ij}^2 - n d_{ij}^2\right\}$$

Torben Tvedebrink
tvede@math.aau.dk

MDS - Forskellige mål

Klassisk MDS går ud på at minimere afstanden mellem observerede afstande δ_{ij} (i høj dimension) og d_{ij} (afstande i planen):

$$E_{\text{Classic}}(\delta, d) = \frac{\sum_{i \neq j}(\delta_{ij}^2 - d_{ij}^2)}{\sum_{i \neq j} \delta_{ij}^2},$$

hvor nævneren sikre standardiserede residualer (δ_{ij} kunne skaleres hvorved tælleren vokser).

Torben Tvedebrink
tvede@math.aau.dk

MDS - Forskellige mål

Klassisk MDS går ud på at minimere afstanden mellem observerede afstande δ_{ij} (i høj dimension) og d_{ij} (afstande i planen):

$$E_{\text{Classic}}(\delta, d) = \frac{\sum_{i \neq j}(\delta_{ij}^2 - d_{ij}^2)}{\sum_{i \neq j} \delta_{ij}^2},$$

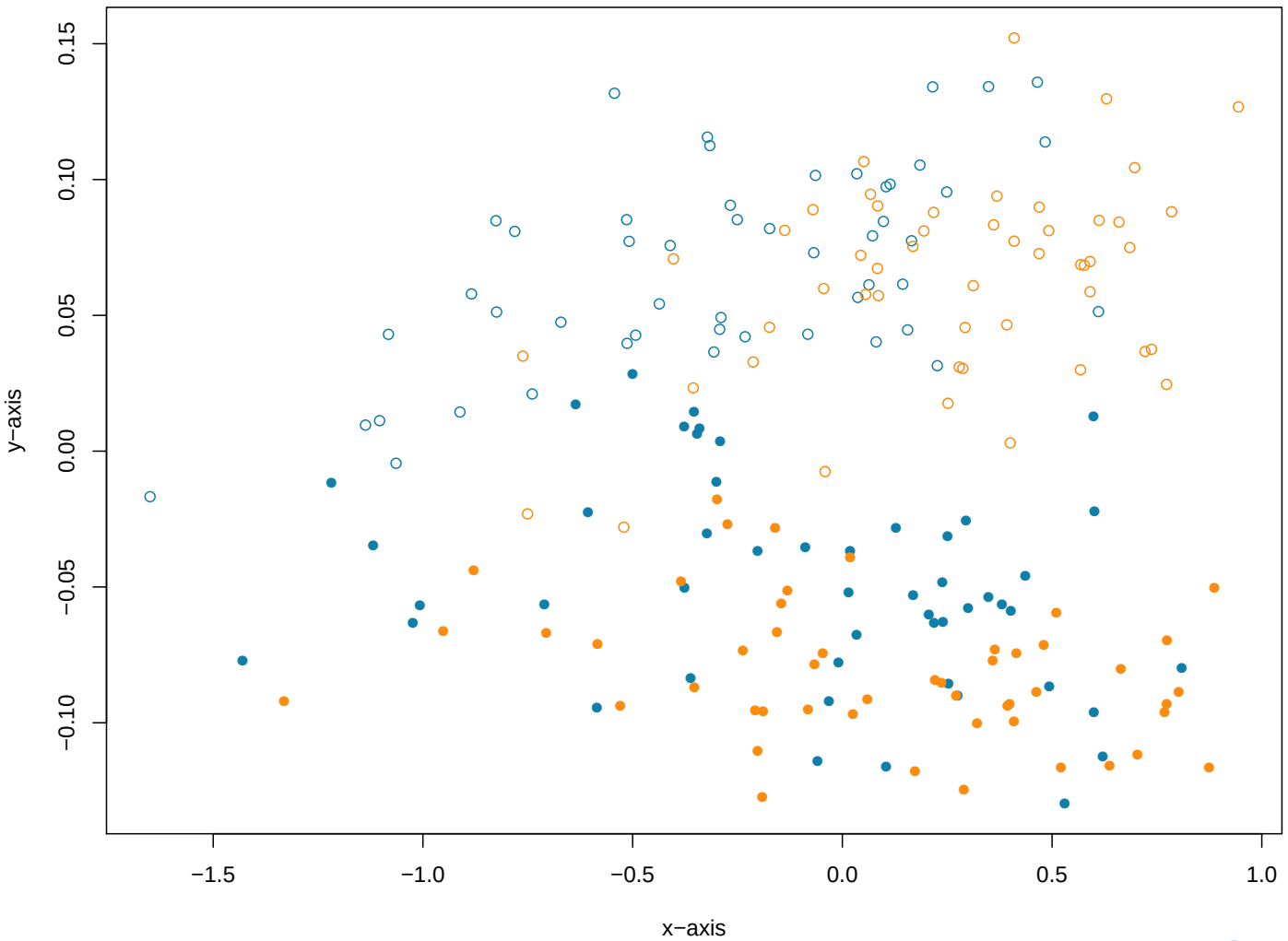
hvor nævneren sikre standardiserede residualer (δ_{ij} kunne skaleres hvorved tælleren vokser).

Andre afstandsmål er Sammon's mapping (Sa) og Kruskal's STREES[2] for Non-metric MDS

$$E_{\text{Sammon}}(\delta, d) = \frac{1}{\sum_{i \neq j} \delta_{ij}} \sum_{i \neq j} \frac{(\delta_{ij} - d_{ij})^2}{\delta_{ij}}$$

Torben Tvedebrink
tvede@math.aau.dk

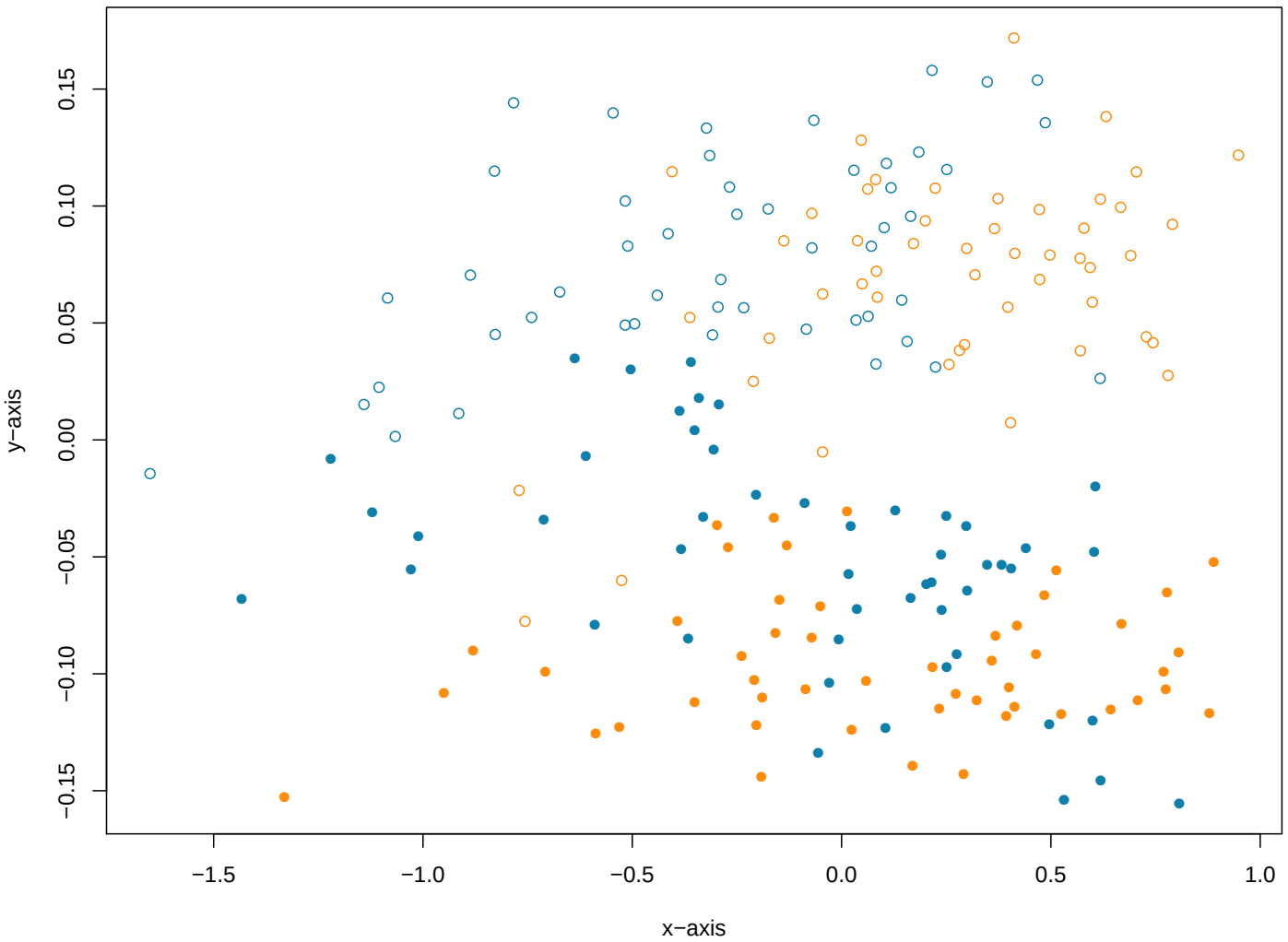
MDS - Eksempel



`cmdscale`

Torben Tvedebrink
tvede@math.aau.dk

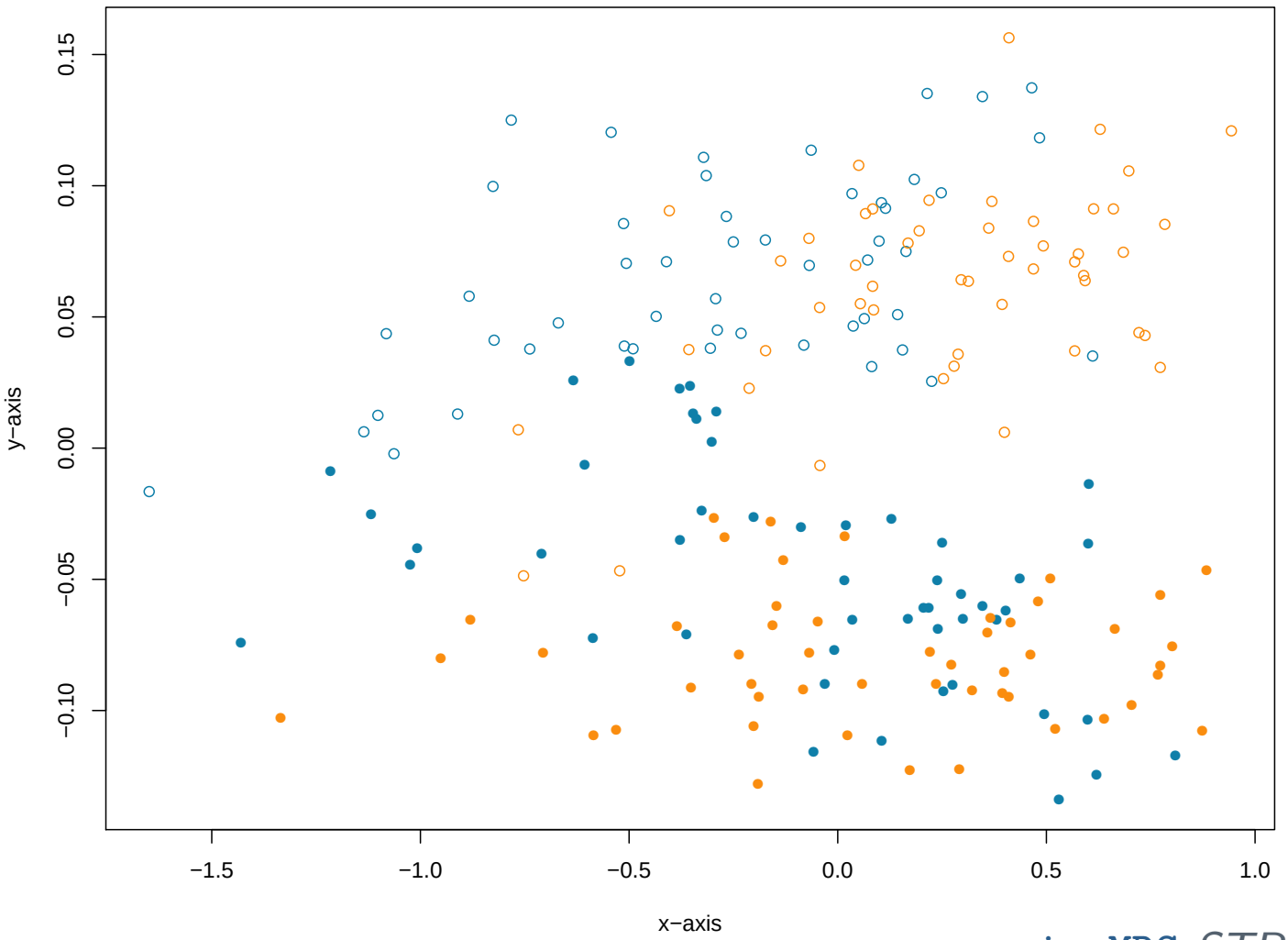
MDS - Eksempel



sammon

Torben Tvedebrink
tvede@math.aau.dk

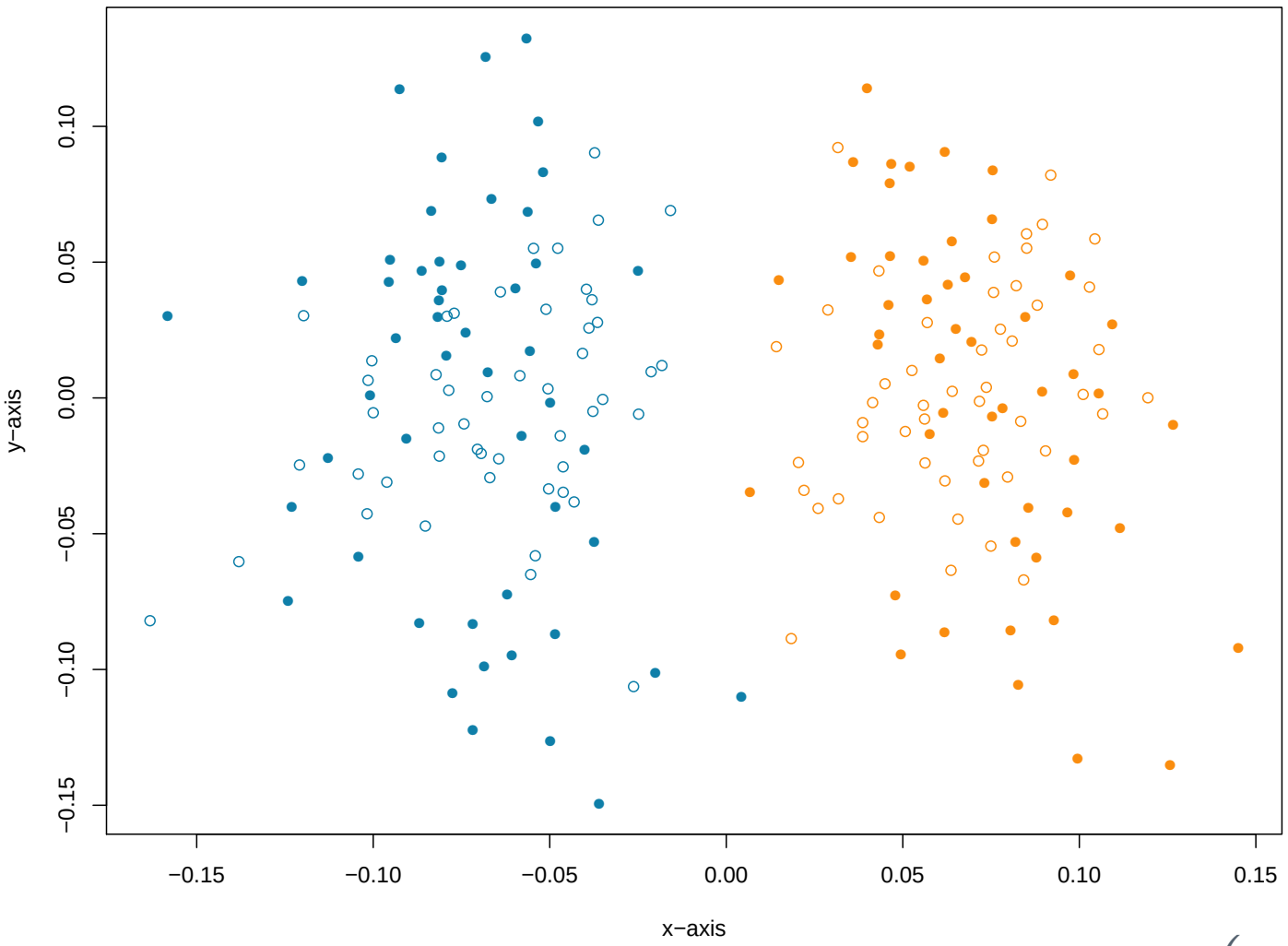
MDS - Eksempel



isoMDS $STRESS^2$

Torben Tvedebrink
tvede@math.aau.dk

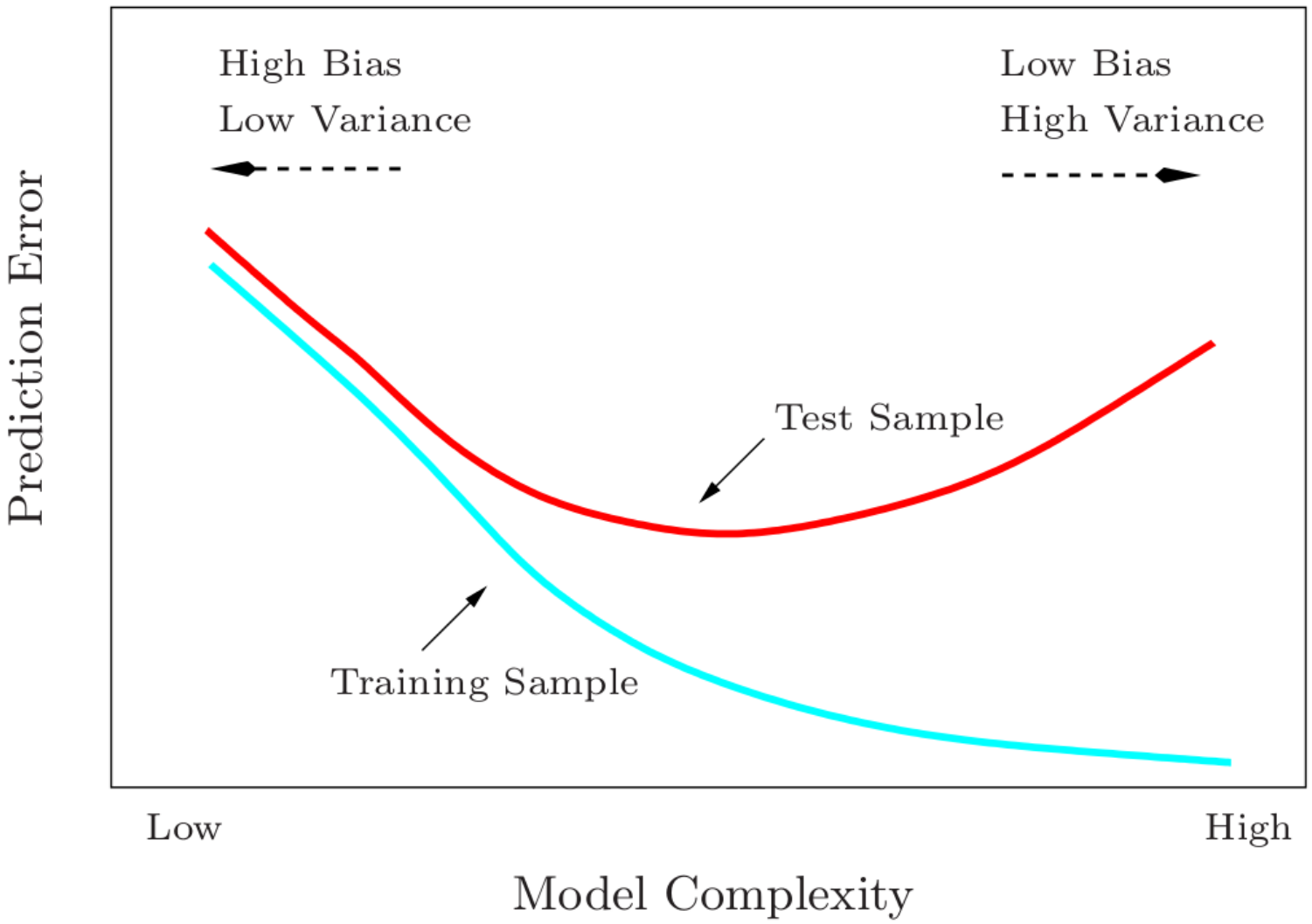
MDS - Eksempel



sammon *(rescaled)*

Torben Tvedebrink
tvede@math.aau.dk

Modelkompleksitet



Torben Tvedebrink
tvede@math.aau.dk

Mean Squared Error

Antag at den sande sammenhæng mellem responsen Y og forklarende variable X er givet ved

$$Y = f(\boldsymbol{X}) + \varepsilon,$$

hvor f er en funktion og ε er fejlledet, med $\mathbb{E}(\varepsilon) = 0$.

Baseret på data kan vi estimere den funktionen f og få estimatet $\hat{f}$. Ved at bruge MSE kan vi sige noget om modellens præcision:

$$\text{MSE} = n^{-1} \sum_{i=1}^{n} (Y_i - \hat{f}(\boldsymbol{X}_i))^2$$

Torben Tvedebrink
tvede@math.aau.dk

Variance-bias tradeoff

Vi forestiller os at vi kan gentage ekserimentet et antal gange, og ønsker at bestemme den MSE udover disse realisationer.

Det betyder at vi ser på den forventede MSE:

$$\mathbb{E}[\text{MSE}] = n^{-1} \sum_{i=1}^{n} \mathbb{E}[(Y_i - \hat{f}(\boldsymbol{X}_i))^2].$$

Torben Tvedebrink
tvede@math.aau.dk

Variance-bias tradeoff

Vi forestiller os at vi kan gentage ekserimentet et antal gange, og ønsker at bestemme den MSE udover disse realisationer.

Det betyder at vi ser på den forventede MSE:

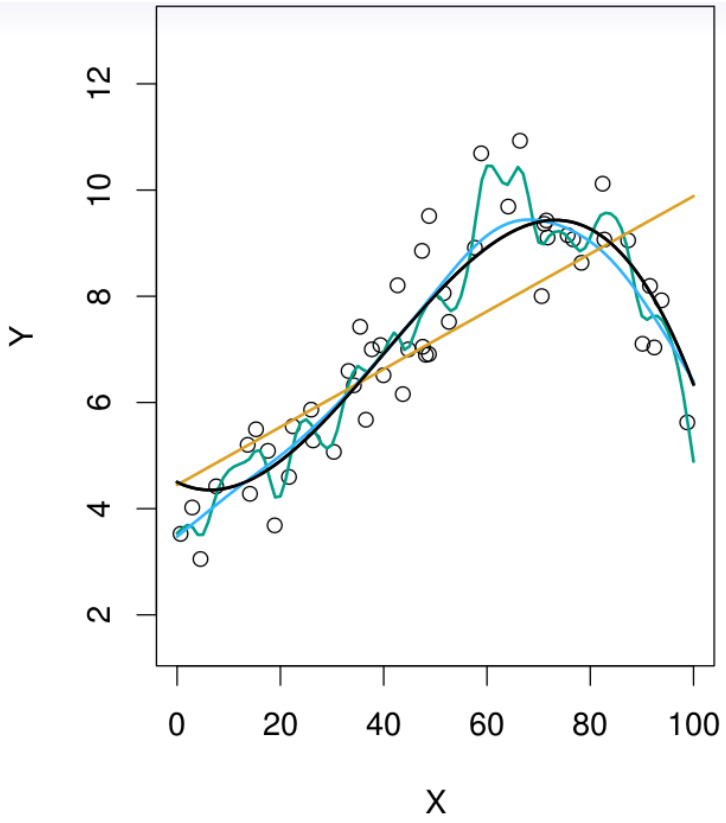
$$\mathbb{E}[\text{MSE}] = n^{-1} \sum_{i=1}^{n} \mathbb{E}[(Y_i - \hat{f}(\boldsymbol{X}_i))^2].$$

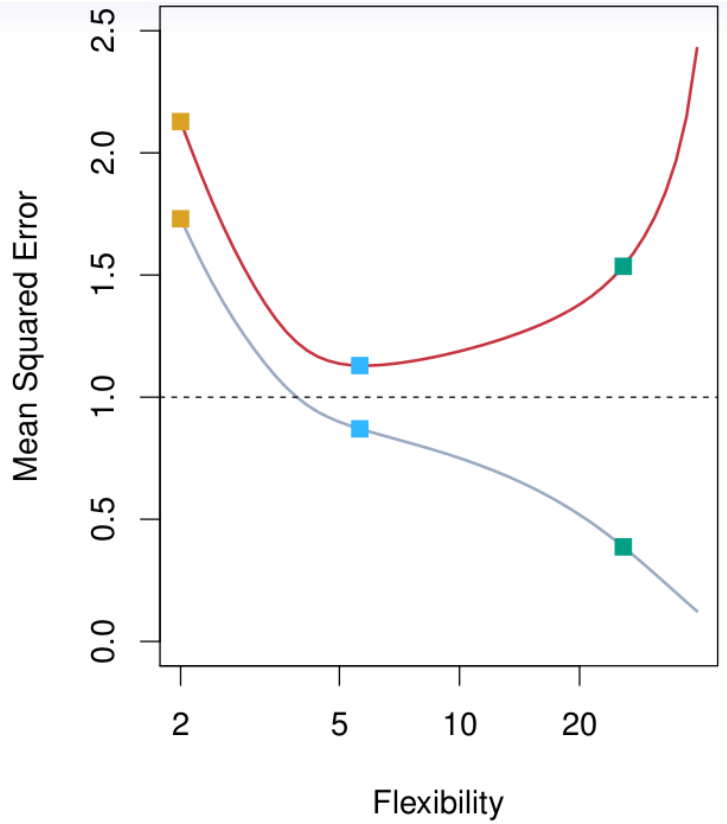
Vi får ved at ekspandere på passende vis at

$$\mathbb{E}[\text{MSE}] = \underbrace{\mathbb{E}(\varepsilon^2)}_{\mathbb{V}(\varepsilon)} + \underbrace{\mathbb{E}[(f - \mathbb{E}\{\hat{f}\})^2]}_{\text{bias}^2} + \underbrace{\mathbb{E}[(\mathbb{E}(\hat{f}) - \hat{f})^2]}_{\mathbb{V}(\hat{f})}$$

Torben Tvedebrink
tvede@math.aau.dk

Eksempler

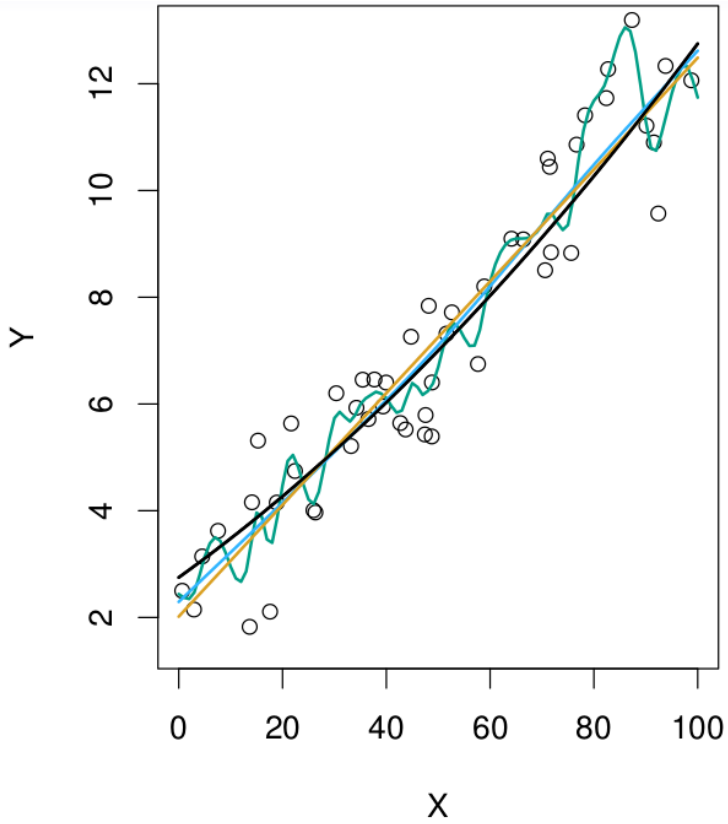


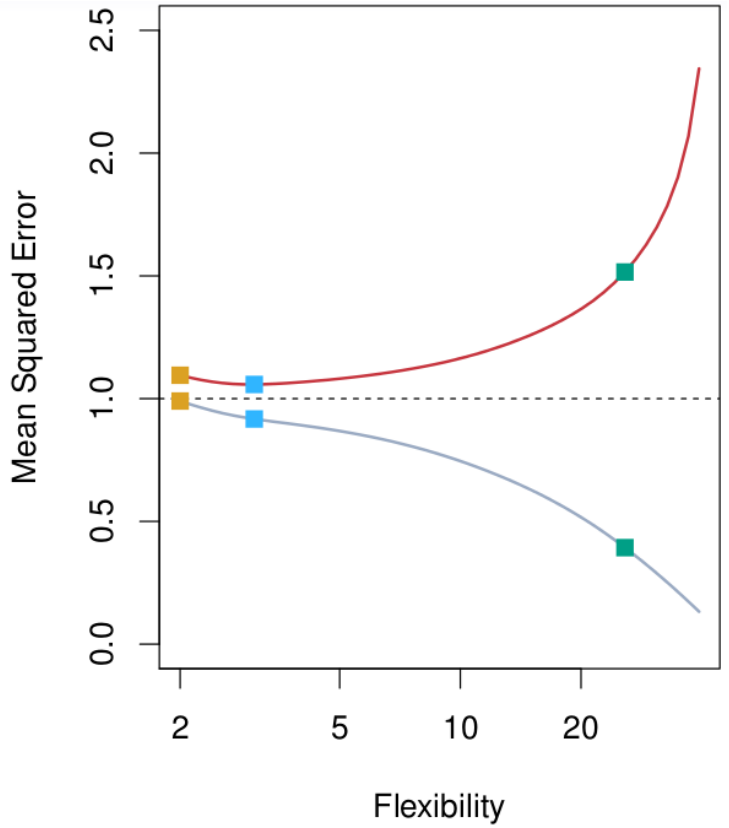


Fra James, Witten, Hastie og Tibshirani:
Sort kurve: Data genererende struktur
Orange linje: Lineær regression
Grøn og Blå kurver: Splines

Torben Tvedebrink
tvede@math.aau.dk

Eksempler

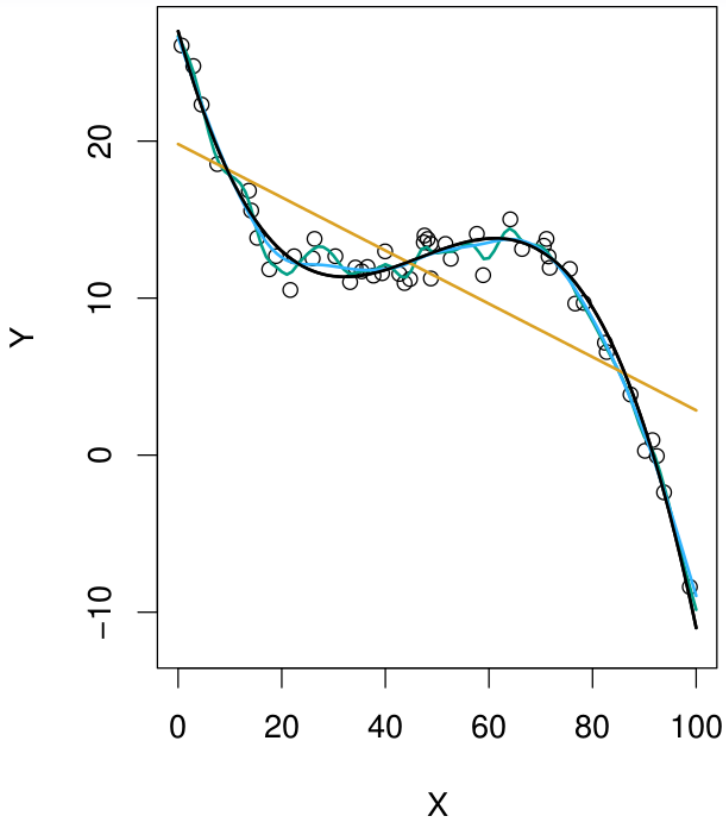


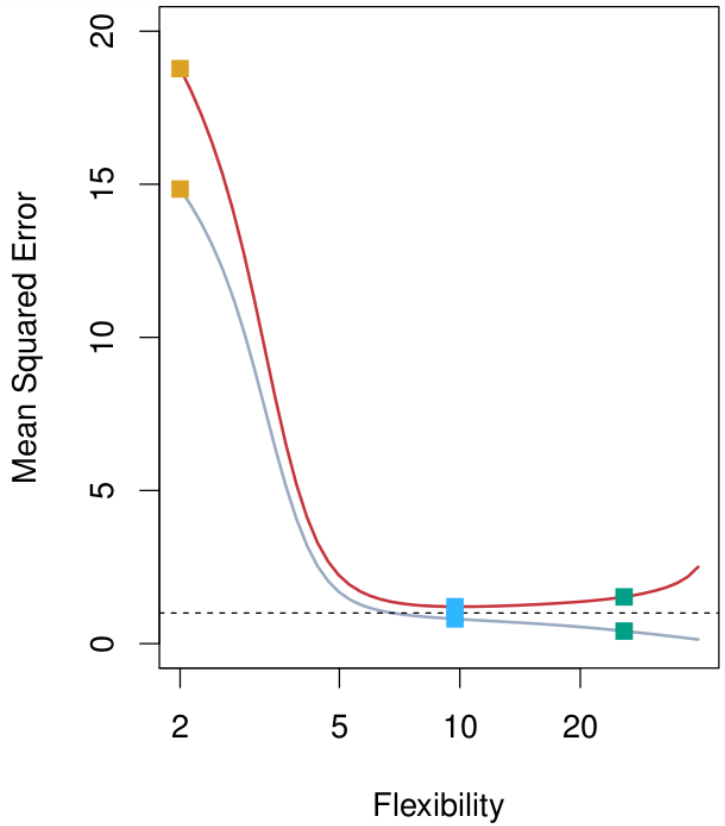


Fra James, Witten, Hastie og Tibshirani:
Sort kurve: Data genererende struktur
Orange linje: Lineær regression
Grøn og Blå kurver: Splines

Torben Tvedebrink
tvede@math.aau.dk

Eksempler

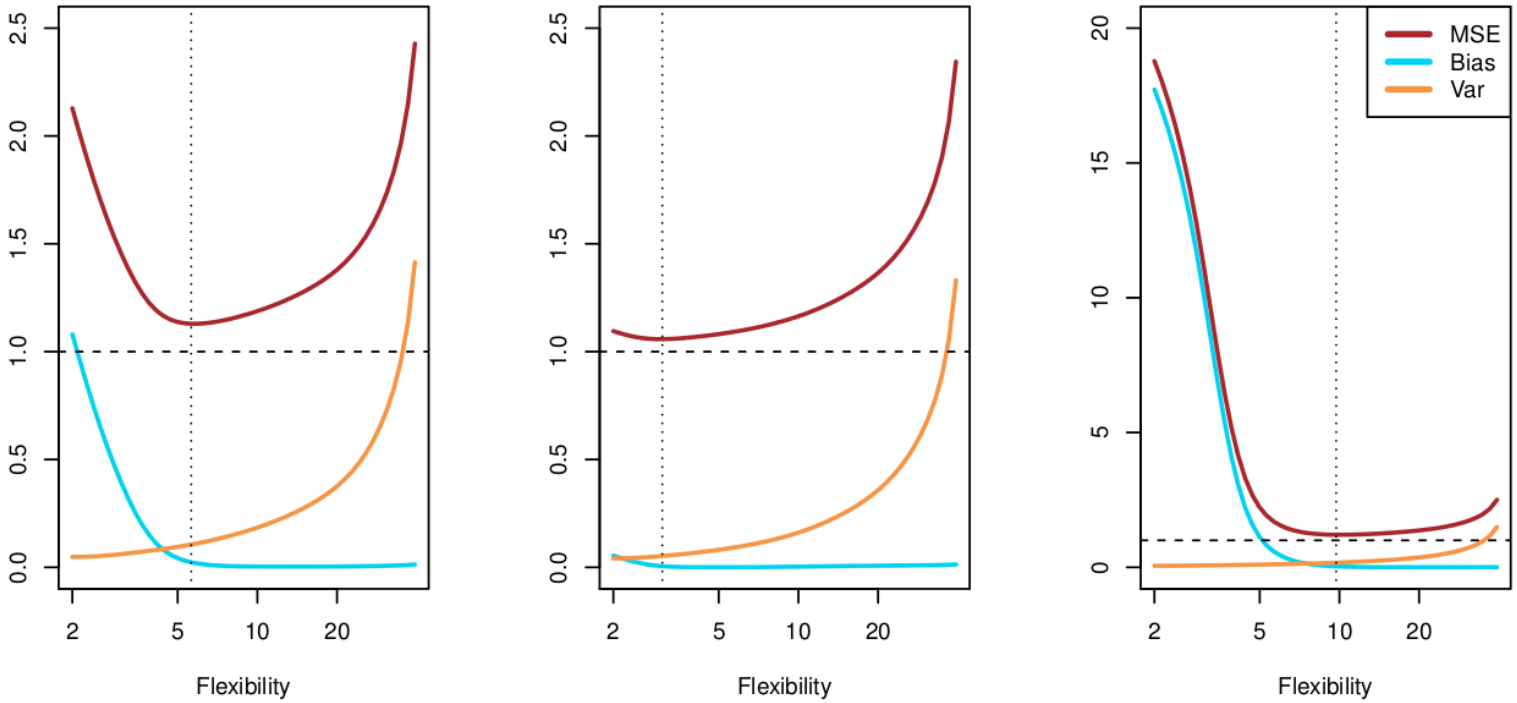




Fra James, Witten, Hastie og Tibshirani:
Sort kurve: Data genererende struktur
Orange linje: Lineær regression
Grøn og Blå kurver: Splines

Torben Tvedebrink
tvede@math.aau.dk

Eksempler



Fra James, Witten, Hastie og Tibshirani:
Sort kurve: Data genererende struktur
Orange linje: Lineær regression
Grøn og Blå kurver: Splines

Torben Tvedebrink
tvede@math.aau.dk

Klassifikation

Vi ønsker at klassificere observationer (fx. kunder, produkter) ud fra en række data/informationer (features, attributter, ...), for i fremtiden at kunne klassificere nye observationer baseret på disse data i deres mest sandsynlige klasse.

Klassifikationer repræsenteres ofte som træer, hvor hver rute fra *rod* til *blad* er en fællesmængde af udsagn, mens træet i sig selv er en forening af disse *ruter*.

Torben Tvedebrink
tvede@math.aau.dk

Beslutningstræ

Beslutningstræs læring (Decision tree learning) fungerer for:

- Variabeltilstande er kategoriske (virker dog også for numeriske variable)
- Den endelige beslutning er kategorisk (ja/nej, syg/rask, ...)
- Data kan indeholde fejl/støj
- Der er manglende attribut værdier i træningsdata

Torben Tvedebrink
tvede@math.aau.dk

C4.5-algoritmen

Utallige algoritmer udfører denne type klassificering - C4.5 (og C5.0) gør dette.

C4.5 er efterfølgeren til ID3, som en af de ældste (og mest effektive) klassifikations algoritmer.

Således *benchmarkes* nye metoder ofte i forhold til ID3/C4.5.

Torben Tvedebrink
tvede@math.aau.dk

Algoritme: C4.5

Input: an attribute-valued dataset D

1. Tree $= \{\}$
2. **if** D is "pure" OR other stopping criteria met **then**
3. terminate
4. **end if**
5. **for all** attribute $a \in D$ **do**
6. Compute information-theoretic criteria if split on a
7. **end for**
8. a_{best} = Best attribute according to above criteria
9. Tree = Create a decision node that tests a_{best} in the root
10. D_v = Induced sub-datasets from D based on a_{best}
11. **for all** D_v **do**
12. $\text{Tree}_v = C4.5(D_v)$
13. Attach Tree_v to the corresponding branch of Tree
14. **end for**
15. **return** Tree

Torben Tvedebrink
tvede@math.aau.dk

Entropi - et mål for viden/"kaos"

Entropien for et givet datasæt D, hvor p_i er andelen af cases med med klasse i bland c mulige klasser, er givet ved

$$Entropi(D) = \sum_{i=1}^{c} -p_i \log_2 p_i,$$

hvor $0 \log_2 0 = 0$ per definition.

Bemærk, hvis $p_i = 1$ og $p_j = 0$, $j \neq i$ er entropien 0, mens den er maksimal hvis $p_i = 1/c$ for alle $i \neq j$ (størst usikkerhed).

Generelt $-\log_2(1/c) = \log_2(c)$, og for $c = 2$ er entropien maksimalt er 1.

Torben Tvedebrink
tvede@math.aau.dk

Eksempel
Golf data

Day	Outlook	Temperature	Humidity	Wind	*Play golf?*
1	sunny	hot	high	weak	no
2	sunny	hot	high	strong	no
3	overcast	hot	high	weak	yes
4	rain	mild	high	weak	yes
5	rain	cool	normal	weak	yes
6	rain	cool	normal	strong	no
7	overcast	cool	normal	strong	yes
8	sunny	mild	high	weak	no
9	sunny	cool	normal	weak	yes
10	rain	mild	normal	weak	yes
11	sunny	mild	normal	strong	yes
12	overcast	mild	high	strong	yes
13	overcast	hot	normal	weak	yes
14	rain	mild	high	strong	no

Torben Tvedebrink
tvede@math.aau.dk

Eksempel
Golf data

Day	Outlook	Temperature	Humidity	Wind	*Play golf?*
1	sunny	hot	high	weak	no
2	sunny	hot	high	strong	no
3	overcast	hot	high	weak	yes
4	rain	mild	high	weak	yes
5	rain	cool	normal	weak	yes
6	rain	cool	normal	strong	no
7	overcast	cool	normal	strong	yes
8	sunny	mild	high	weak	no
9	sunny	cool	normal	weak	yes
10	rain	mild	normal	weak	yes
11	sunny	mild	normal	strong	yes
12	overcast	mild	high	strong	yes
13	overcast	hot	normal	weak	yes
14	rain	mild	high	strong	no

$$Entropi(D) = -(9/14)\log_2(9/14) - (5/14)\log_2(5/14) = 0.94$$

Torben Tvedebrink
tvede@math.aau.dk

Information-theoretic criteria

Ønsket er at have så lille en entropi som muligt - mindst usikkerhed.

Informations *gain* er den forventede mindskelse i entropi ved opdeling baseret på en bestemt attribut, A:

$$Gain(D, A) = Entropi(D) - \sum_{v \in val(A)} \frac{|D_v|}{|D|} Entropi(D_v),$$

hvor D_v er datasættet med attribute A fixeret på værdi v blandt As mulige værdier, $val(A)$.

Gain(D,A) ønskes så stor som muligt.

Torben Tvedebrink
tvede@math.aau.dk

Information-theoretic criteria

Ønsket er at have så lille en entropi som muligt - mindst usikkerhed.

Informations *gain* er den forventede mindskelse i entropi ved opdeling baseret på en bestemt attribut, A:

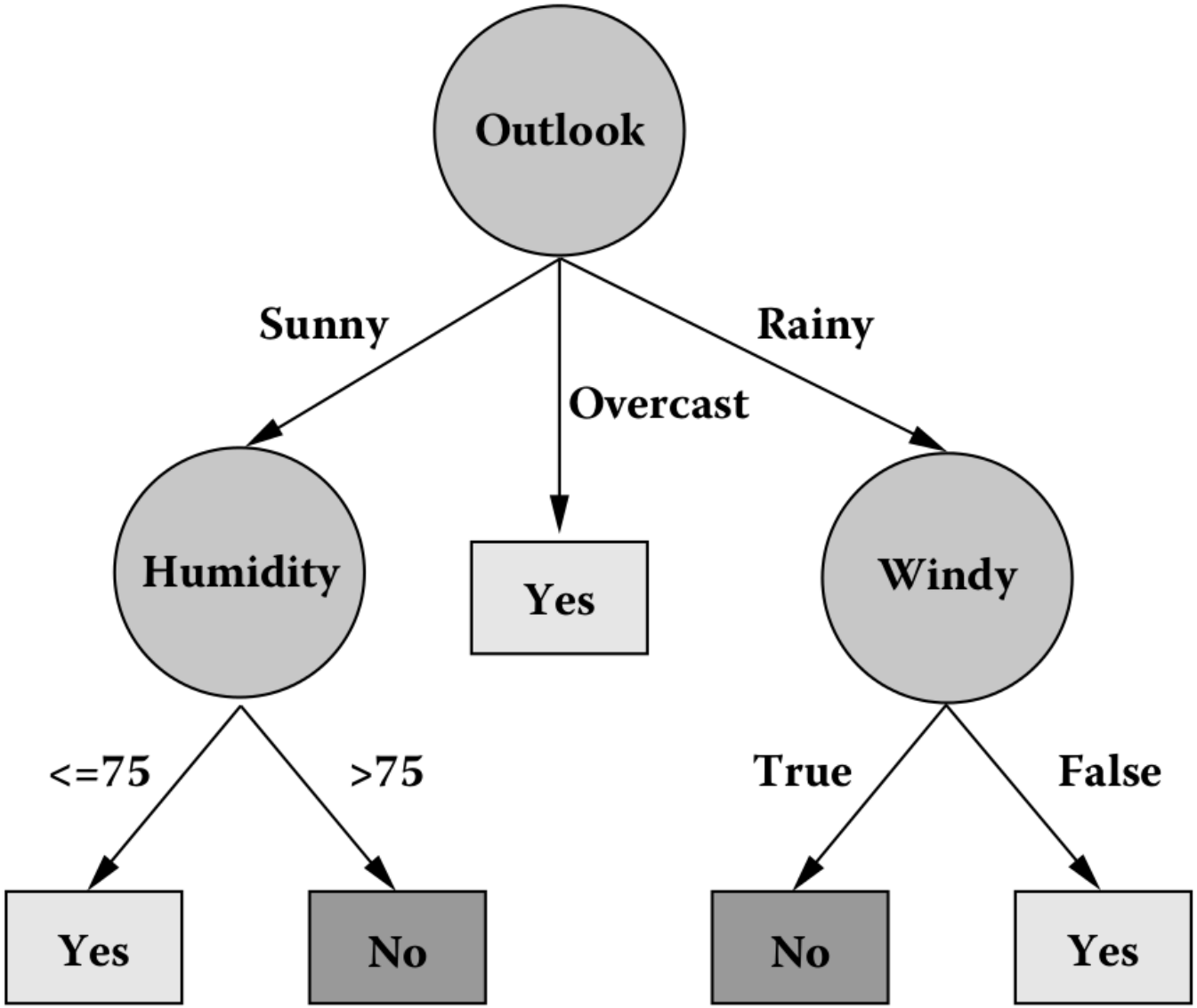
$$Gain(D, A) = Entropi(D) - \sum_{v \in val(A)} \frac{|D_v|}{|D|} Entropi(D_v),$$

hvor D_v er datasættet med attribute A fixeret på værdi v blandt As mulige værdier, $val(A)$.

$$gain(D, \texttt{"outlook"}) = 0.247$$
$$gain(D, \texttt{"wind"}) = 0.048$$

Torben Tvedebrink
tvede@math.aau.dk

Klassifikationstræ
Golf-data



Torben Tvedebrink
tvede@math.aau.dk

C5.0 i R

```
summary(C5.0(play~.,data=golf))
summary(C5.0(play~.,data=golf,rules=TRUE))

head(cgolf)
   outlook temperature temp humidity humid   wind play
1    sunny         hot   85     high    85   weak   no
2    sunny         hot   80     high    90 strong   no
3 overcast         hot   83     high    78   weak  yes
4     rain        mild   70     high    96   weak  yes
5     rain        cool   68   normal    80   weak  yes
6     rain        cool   65   normal    70 strong   no

cgolf <- cgolf[,c("outlook","temp","humid","wind","play")]
summary(C5.0(play~.,data=cgolf))
```


Torben Tvedebrink
tvede@math.aau.dk

Data eksempel

`churnTrain` data fra `library(C50)` har ikke noget at gøre med at kærne smør, men derimod kundeafgang.

Torben Tvedebrink
tvede@math.aau.dk

R kode

```
## Look at http://www.rulequest.com/see5-unix.html
## Loads the library for C5.0 (successor of C4.5)
library(C50)

## Load and look at the top rows of churnTrain data
data(churn)
head(churnTrain)

## Construnct the decision tree using the C5.0 algorithm with
## churn as response and remaining variables as predictors
treemodel <- C5.0(churn ~.,data=churnTrain)

## Ruletree
rulemodel <- C5.0(churn ~.,data=churnTrain, rule=TRUE)

## Print summary of the fitted tree model
summary(treemodel)
```


Torben Tvedebrink
tvede@math.aau.dk

KISS princippet

KISS er et akronym for **K**eep **I**t **S**imple, **S**tupid!

Dette princip gælder indenfor mange videnskaber, herunder statistik, og i særdeleshed for data mining.

Vi bør som udgangspunkt stræbe efter simple beskrivelser af data. Dette gør det nemmere at fortolke modeller - og for data mining mere sandsynligt at kunne generalisere ens model/strukturer til nye datasæt.

Torben Tvedebrink
tvede@math.aau.dk

Håndtering af numeriske variable

Ligefrem og umiddelbar løsning:

- Split for hver mulig værdi (midtpunkter mellem unikke værdier)
- Vælg det bedste split punkt jf. informations kriterie
- Informations gain på split er således informations gain for attribut
- *Er oplagt mere beregningskrævende end for kategoriske variable*

Torben Tvedebrink
tvede@math.aau.dk

GainRatio

For at undgå at split på variable med mange niveauer (fx. datoer) benyttes i C4.5 *GainRatio*:

$$GainRatio(D, A) = \frac{Gain(D, A)}{Entropi(D, A)},$$

hvor $Entropi(D, A)$ bestemmes ved

$$Entropi(D, A) = -\sum_{v \in value\{A\}} \frac{|D_v|}{|D|} \log_2 \frac{|D_v|}{|D|}.$$

Dvs. for variable med mange numeriske værdier vil entropien blive høj, idet næsten alle observationer vil være unikke.

Denne konstruktion *straffer* variable med 'uniforme' splits - husk $Entropi(D)$ er størst for uniform fordeling. *Se opgave*

Torben Tvedebrink
tvede@math.aau.dk

Manglende observationer

Kan vi blot betragte `NA` som en ny kategori?

Ex. Ved gen ekspressions analyser er visse observationer manglende pga. for lav/høj måling.

Ex. Hvis `Gravid=NA` er det ikke samme 'information' for mænd (burde være `Nej`) som for 25-årige kvinder (burde være `Måske`).

Torben Tvedebrink
tvede@math.aau.dk

Håndtering af manglende observationer

Problemer:

1. Hvordan vælges passende attribut at splitte på ved `NA`?
2. Når en attribut er valgt - hvilken 'gren' af træet skal `NA` observationer tildeles?
3. Hvordan klassificeres nye data efterfølgende ved `NA` på relevante attributer?

Torben Tvedebrink
tvede@math.aau.dk

Håndtering af manglende observationer

Problemer:

1. Hvordan vælges passende attribut at splitte på ved `NA`?

Løsninger:

a Ignorere data med `NA`
b Bruge den hyppigst forekommende attribut (kategorisk) eller gennemsnit (numerisk)
c Forsøge at imputere variablen baseret på øvrige variable (*model*)

Torben Tvedebrink
tvede@math.aau.dk

Håndtering af manglende observationer

Problemer:

2. Når en attribut er valgt - hvilken 'gren' af træet skal `NA` observationer tildeles?

Løsninger:

a Ignorere data med `NA`
b hyppigst forekommende attribut (kategorisk) eller gennemsnit (numerisk)
c Angiv (som fraktion) til hver del-datasæt ud fra disses størrelse
  c.1 Evt. kun til del-datasættet med flest observationer (som fraktion i forhold til størrelser)
d Lav egen 'gren' til `NA` observationer
e Tilskriv mest sandsynlige værdi givet øvrige variable (*model*).

Torben Tvedebrink
tvede@math.aau.dk

Håndtering af manglende observationer

Problemer:

3. Hvordan klassificeres nye data efterfølgende ved `NA` på relevante attributer?

Løsninger:

a Hvis `NA`-'gren' findes følges denne
b Følg hyppigst forekommende gren
c Imputer værdi baseret på øvrige variable (*model*)
d Stop og baser kun på (betinget) sandsynlighed på det pågældende sted i træet

Torben Tvedebrink
tvede@math.aau.dk

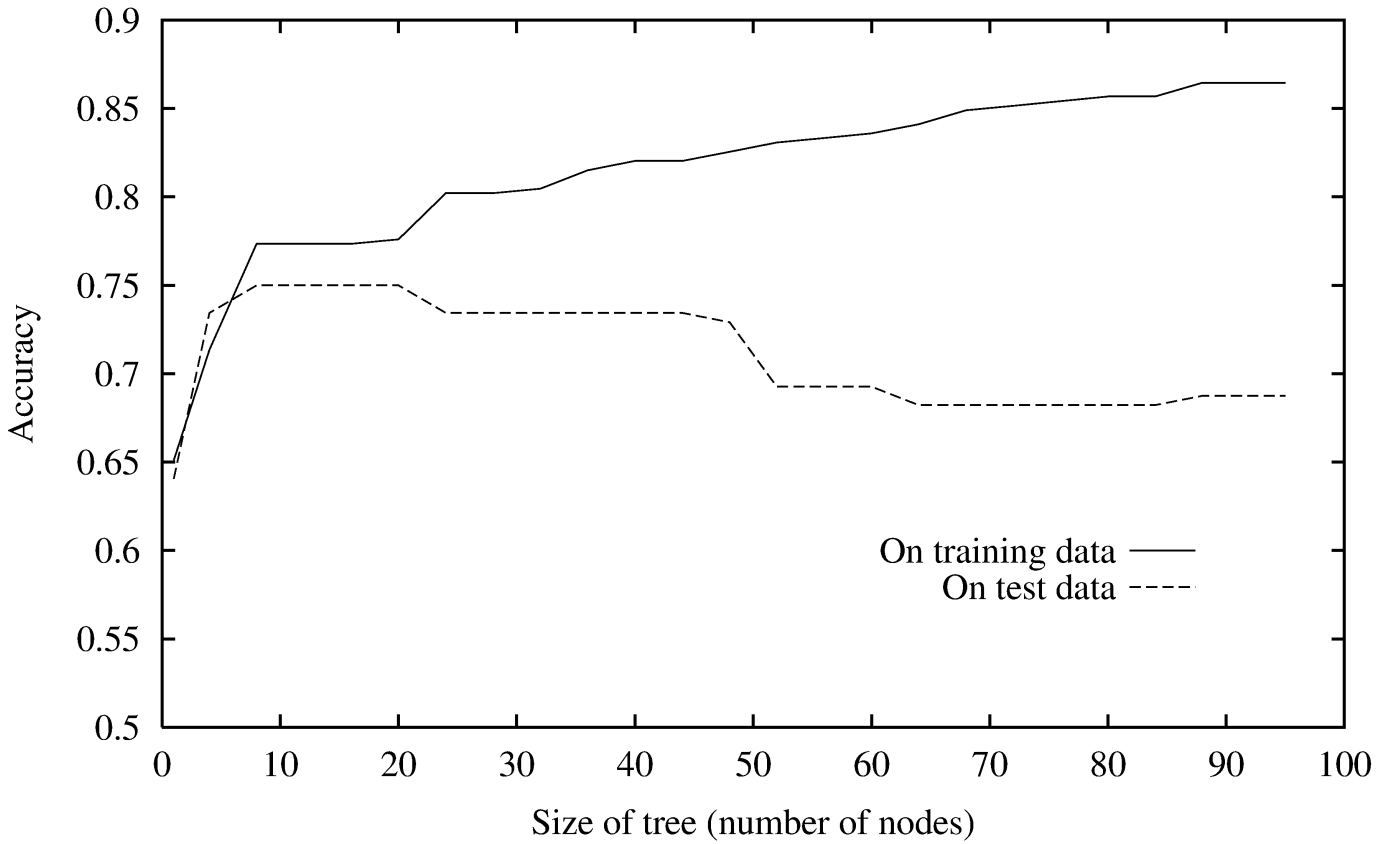
Undgå overfitting

Overfitting

En given hypotese h (et træ) siges at *overfitte* træningsdata hvis der findes en alternativ hypotese h' (et andet træ) således at h har en mindre fejl end h' for træningsdata, men h' har en mindre fejl end h for den generelle problemstilling (mulige realisationer af data).

Torben Tvedebrink
tvede@math.aau.dk

Træningsdata og valideringsdata



Model for patienter med diabetes

Torben Tvedebrink
tvede@math.aau.dk

Overfitting

Overfitting kan skyldes . . . :

- Fejl/støj i data - eks. golf data
- For små datasæt kan der tilfældigt være sammenhænge mellem attribut og *target* som eller er uafhængige
- . . .

Torben Tvedebrink
tvede@math.aau.dk

Overfitting

Overfitting kan skyldes . . . :

- Fejl/støj i data - eks. golf data
- For små datasæt kan der tilfældigt være sammenhænge mellem attribut og *target* som eller er uafhængige
- . . .

To umiddelbare måder at undgå overfitting:

- Stop før 'bladene' bliver *for* rene
- Post-pruning af træet efter termination af algoritme

Torben Tvedebrink
tvede@math.aau.dk

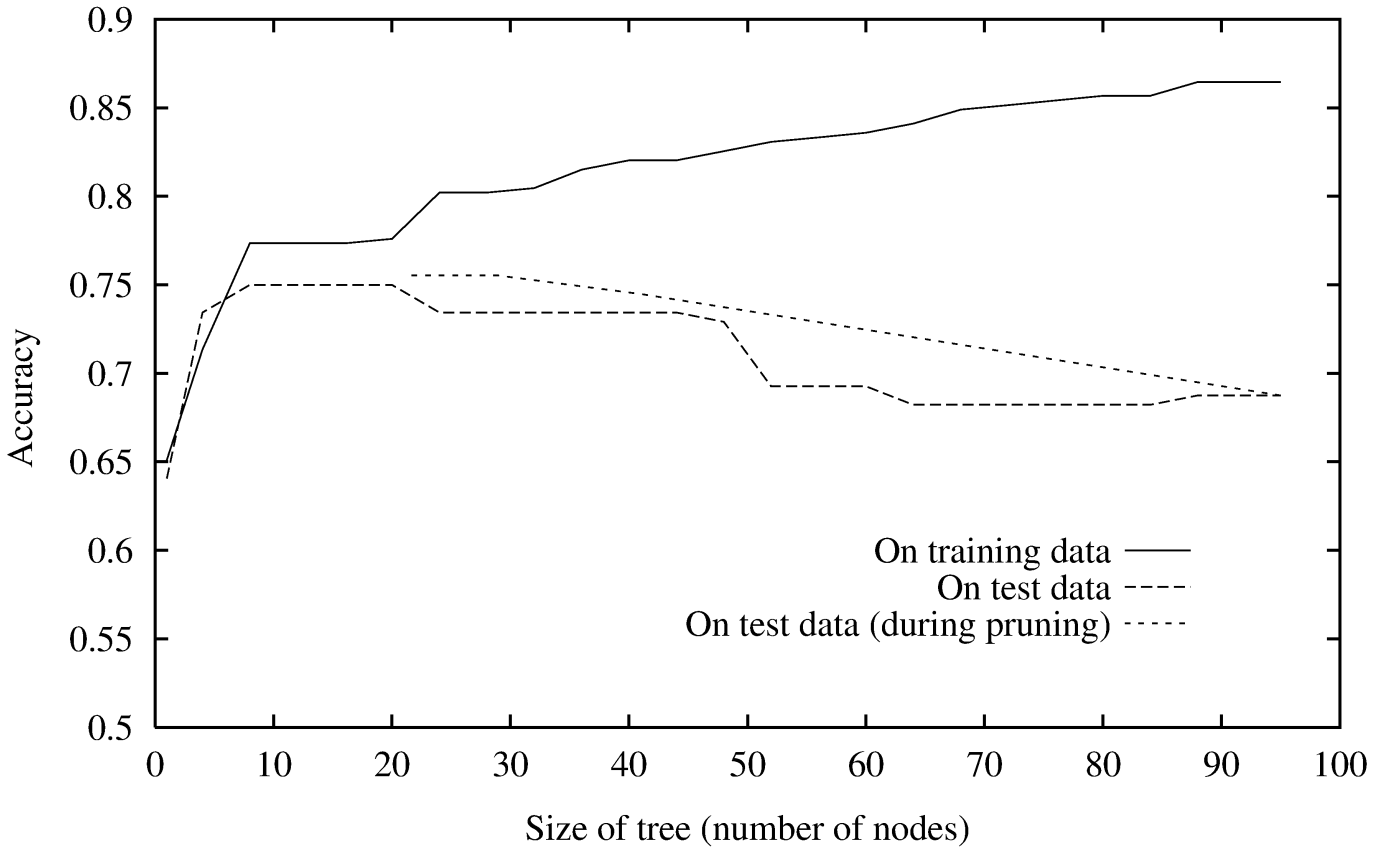
Pruning

Pruning består i at slå et undertræ sammen til et blad, hvor typen tilskrives den hyppigst forekommende klasse.

Knuder (dvs undertræer) fjernes kun vil det simplere træ ikke er dårligere end det oprindelige for valideringssættet.

Torben Tvedebrink
tvede@math.aau.dk

Pruning



Torben Tvedebrink
tvede@math.aau.dk

Estimering af fejlsandsynlighed

Hvis ens datasæt er begrænset kan den forventede fejl forsøges estimeret ud fra en anskuelse om at træet er biased i forhold til træningssættet.

Torben Tvedebrink
tvede@math.aau.dk

Estimering af fejlsandsynlighed

C4.5 benytter sig af *pessimistisk prunining*. Hvis et blad har N observationer, og E af disse er misklassificeret , er et empirisk skøn af fejlen $(E + 0.5)/N$, hvor 0.5 er kontinuitets korrektion.

For et undertræ med L blade som kollapses og observationerne tilskrives den hyppigste klasse er fejlen $(\sum_{l=1}^{L} E_l + L/2)/\sum_{l=1}^{L} N_l$

Hvis undertræet erstattes af et blad (med klassen lig flest observationer) og dette medfører at J observationer misklassificeres. Så prunes træet hvis $J + 0.5$ er indenfor én standard afvigelse af $\sum_{l=1}^{L} E_l + L/2$.

Torben Tvedebrink
tvede@math.aau.dk

Pessimistisk pruning

Hvis vi bruger notationen $\varepsilon(T, D) = E/N$, hvor $E = \sum_{l=1}^{L} E_l$ og $N = \sum_{l=1}^{L} N_l$ som er fejlraten for træ T på datasættet D.

Pga. kontinuitets korrektionen får vi

$$\varepsilon'(T, D) = \varepsilon(T, D) + \frac{L}{2N}$$

Ydermere er pruned(T, t) træet T hvor deltræet med rod i knuden t er erstattet af en knude.

Vi pruner hvis

$$\varepsilon'(\text{pruned}(T, t), D) \leq \varepsilon'(T, D) + \sqrt{\frac{\varepsilon'(T, D)[1 - \varepsilon'(T, D)]}{|D|}}$$

Torben Tvedebrink
tvede@math.aau.dk

Opgaver

PCA, **MDS** og **Varians-bias**:

1. Vis at PCA giver ukorrelerede projektions retninger
2. Indlæs data `data(Auto)` fra `ISLR` og foretag en PCA analyse hvor I bruger `cor` hhv. (a) `TRUE` og (b) `FALSE`.
3. Simuler data (`MDS.R`) og eksperimenter med `cmdscale`.
4. Vis $\mathbb{E}[\text{MSE}] = \mathbb{E}(\varepsilon^2) + \mathbb{E}[(f - \mathbb{E}\{\hat{f}\})^2] + \mathbb{E}[(\mathbb{E}(\hat{f}) - \hat{f})^2]$

C4.5

1. Løs opgaverne 2 og 3 i Wu og Kumar afsnit 1.7.
2. Bestem hvornår entropien for en binær (dikotomisk) responsvariabel er størst mulig.
3. Implementer i R funktionerne: Entropi, Gain og GainRatio.
4. Beregn *gain* for `"temperature"` og `"humidity"` når variablene er kodet som på slide 38.
5. Bestem ligeledes *GainRatio* for `"outlook"` og `"windy"`.

Torben Tvedebrink
tvede@math.aau.dk