K-means
og hierarkisk clustering

Lecture 2

Torben Tvedebrink
tvede@math.aau.dk

Institut for Matematiske Fag

AALBORG UNIVERSITY
DENMARK

Self-study opgaver
Lotte, Kenneth og Nikolaj

The floor is yours!

Torben Tvedebrink
tvede@math.aau.dk

Supervised vs. unsupervised learning

Supervised learning er data mining problemer hvor vi kender responsen Y og en række kovariater X. Vi ønsker at være i stand til at lave *inferens* eller *prediktion*, således vores information X giver hhv. indsigt i en sammenhæng eller lav prediktions fejl.

Unsupervised learning optræder når vi kun har X og ingen klar respons Y. I visse tilfælde kan hver vektor i X betragtes som respons, men dette er lidt misvisende. Vi kan således være interesserede i at gruppere vores data i klustre, hvor medlemmerne af hvert kluster er mere ens end elementer fra øvrige klustre.

Semi-supervised learning *er en kombination hvor vi kender Y for m observationer men ikke for de $n - m$ resterende observationer (ikke umiddelbart en del af dette kursus).*

Torben Tvedebrink
tvede@math.aau.dk

C4.5 vs. K-means

Learning tree (som C4.5) knytter sig til *supervised learning*, hvor hver observation i træningsdatasættet har en klasse (fx. syg/rask, ja/nej, . . .).

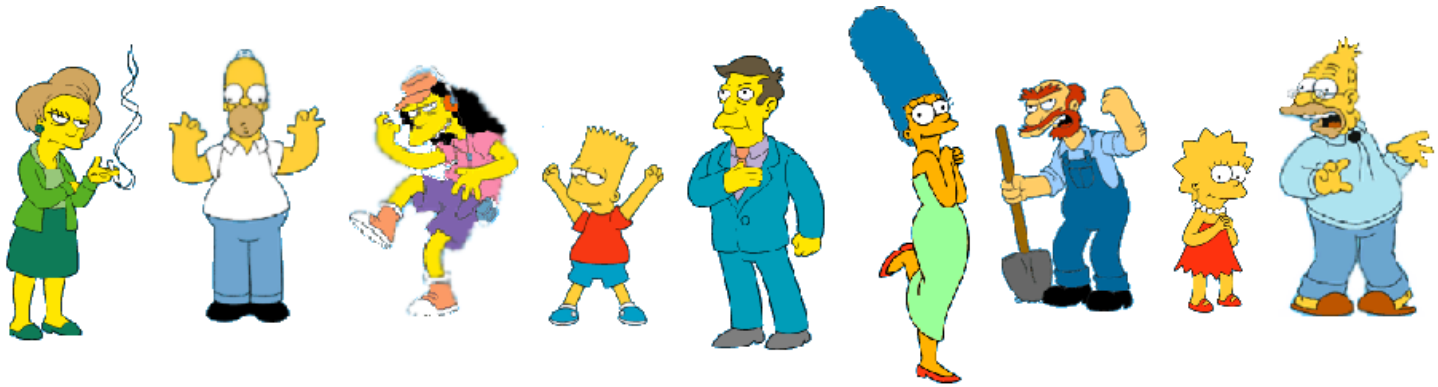
I konstruktionen af træet søges attributter som opdeler klasser bedst muligt.

Cluster analysis (på dansk: klyngeanalyse) forsøger at gruppere observationer som er ens i samme cluster (så tæt på hinanden som muligt), og observationer der er forskellige i andre clustre (så langt fra hinanden som muligt).

Denne opdeling foregår oftes som *unsupervised learning*, idet klassen for observationerne er ukendt.

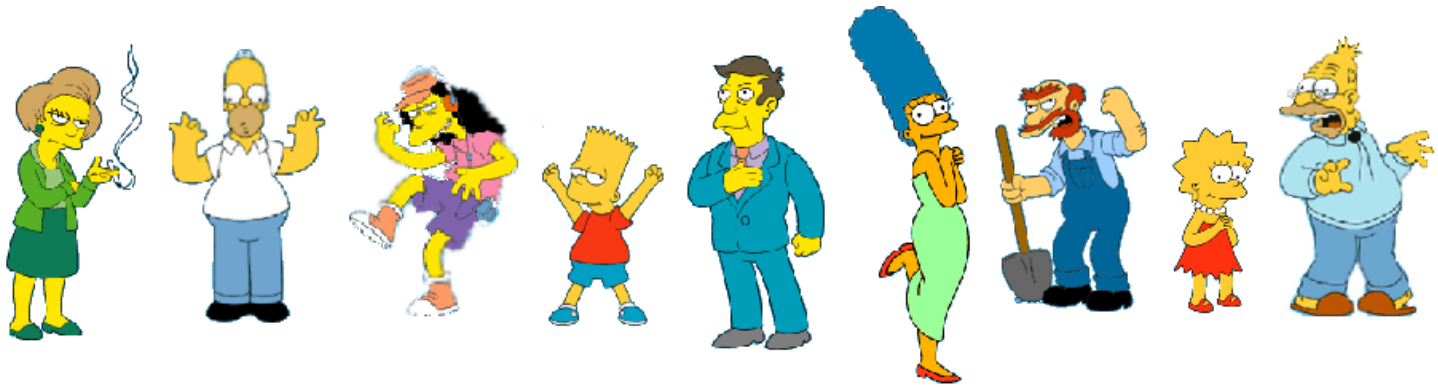
Torben Tvedebrink
tvede@math.aau.dk

Hvad er en naturlig gruppering?

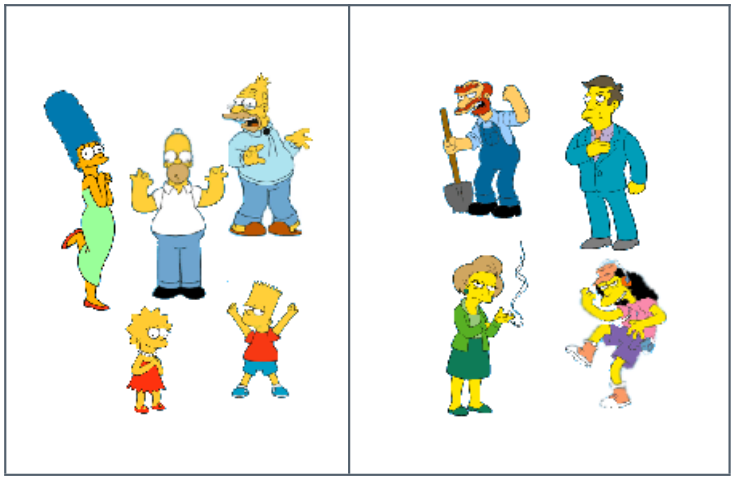


Torben Tvedebrink
tvede@math.aau.dk

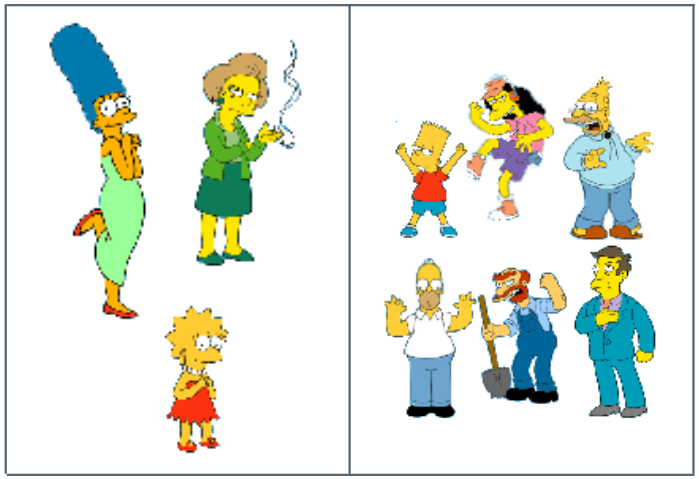
Hvad er en naturlig gruppering?



Clustering er subjektiv



Simpson's Skoleansatte



Kvinder Mænd

Torben Tvedebrink
tvede@math.aau.dk

Hvad er forskellighed?

Torben Tvedebrink
tvede@math.aau.dk

Forskellighed (dissimilarity)

Til at afgøre hvilke objekter der ens - og hvilke der er forskellige, behøves et mål for "(dis)similarity".

Lad $\boldsymbol{x}_i$ være den i'te række/observation. En dissimilarity afstand, D, skal opfylde:

- $D(\boldsymbol{x}_i, \boldsymbol{x}_j) = D(\boldsymbol{x}_j, \boldsymbol{x}_i)$
- $D(\boldsymbol{x}_i, \boldsymbol{x}_i) = 0$
- $D(\boldsymbol{x}_i, \boldsymbol{x}_j) = 0$ hvis og kun hvis $\boldsymbol{x}_i = \boldsymbol{x}_j$.
- $D(\boldsymbol{x}_i, \boldsymbol{x}_j) \leq D(\boldsymbol{x}_i, \boldsymbol{x}_k) + D(\boldsymbol{x}_k, \boldsymbol{x}_j)$

Bemærk at trekantsuligheden gør D til en metrik - denne betingelse behøver ikke at være opfyldt for alle clustering algoritmer.

Torben Tvedebrink
tvede@math.aau.dk

Beregne parvise "dissimilarities" i R

I R kan vi beregne "dissimilarities" mellem objekter/observationer i en data.frame meget enkelt vha. `dist`-funktionen:

```
> data <- data.frame(x1 = c(0,1,3), x2=c(0,1,4))
> (res <- dist(data, method="euclidian", diag=TRUE, upper=TRUE))

         1        2        3
1 0.000000 1.414214 5.000000
2 1.414214 0.000000 3.605551
3 5.000000 3.605551 0.000000

> class(res)
[1] "dist"
```


Torben Tvedebrink
tvede@math.aau.dk

proxy
R-pakke

R-pakken `proxy` udvider `dist`-funktionen med en masse forskellige afstandsmål.

Fx. behøver vi ikke blot at være interesserede i en Euklidiske afstand, men også *Manhattan* afstanden som er $D(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_1 = \sum_{k=1}^{p} |x_{ik} - x_{jk}|$.

[Prøv det for `data` – forrige slide]

Torben Tvedebrink
tvede@math.aau.dk

proxy
R-pakke

R-pakken `proxy` udvider `dist`-funktionen med en masse forskellige afstandsmål.

```
library(proxy)
> summary(pr_DB)
* Similarity measures:
Braun-Blanquet, Chi-squared, correlation, cosine, Cramer,
Dice, eJaccard, Fager, Faith, fJaccard, Gower, Hamman,
Jaccard, Kulczynski1, Kulczynski2, Michael, Mountford, Mozley,
Ochiai, Pearson, Phi, Phi-squared, Russel, simple matching,
Simpson, Stiles, Tanimoto, Tschuprow, Yule, Yule2

* Distance measures:
Bhjattacharyya, Bray, Canberra, Chord, divergence, Euclidean,
Geodesic, Hellinger, Kullback, Levenshtein, Mahalanobis,
Manhattan, Minkowski, Podani, Soergel, supremum, Wave,
Whittaker
```

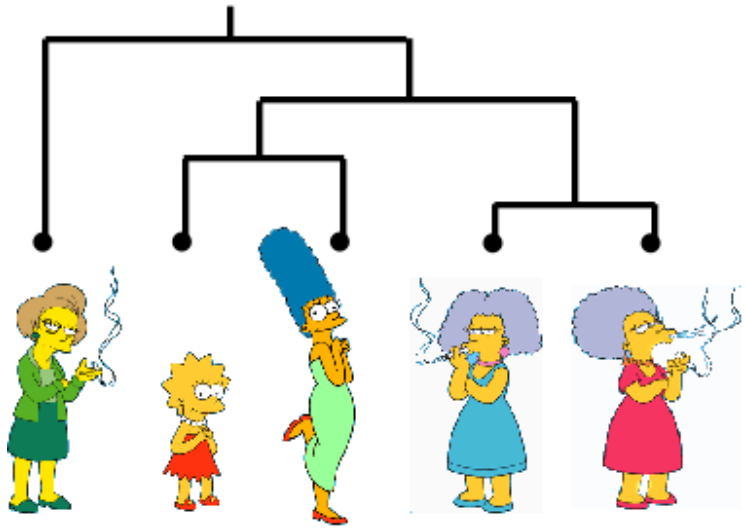

Torben Tvedebrink
tvede@math.aau.dk

To typer af clustering

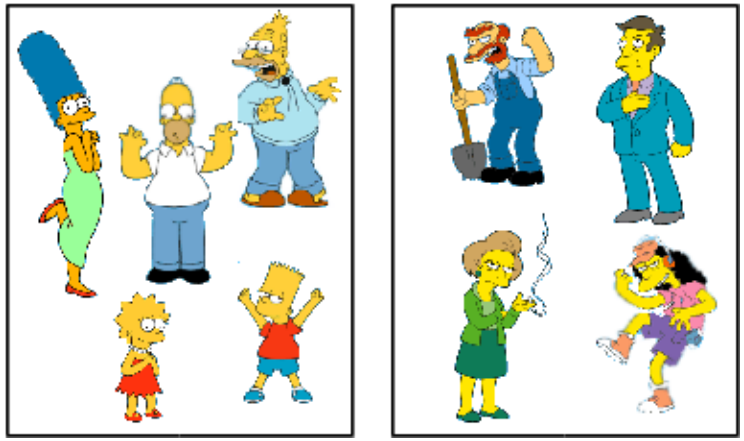
Der findes overordnet to typer af clustering metoder: **hierarkisk** og **partitionering**.

Forskellen er at hierarkisk clustering opdeler clustre i underclustre, mens partitionering opdeler data i disjunkte clustre.

Hierarkisk



Partitionering



Torben Tvedebrink
tvede@math.aau.dk

To typer af clustering

Der findes overordnet to typer af clustering metoder: **hierarkisk** og **partitionering**.

- **Hierarkisk**
  - *Agglomerative* Buttom-up metode som starter med hver observation i eget cluster, og sammensmelter mest *ens* clustere (forskellige mål for *ens*)
  - *Divisive* Starter med alle observationer i et cluster som derefter rekursivt opdeles i de to mest *forskellige* enheder (forskellige mål for *forskel*)
- **Partitionering**
  - Laver disjunkte clustre hvor hvert cluster intern er så ens som muligt, og så forskellig fra de øvrige som muligt (fx. *K*-means).

Torben Tvedebrink
tvede@math.aau.dk

K-means

K-means algoritmen er meget simpel og beror på en opdeling af data i K clustre hvor *within* kvadratsummen ønskes så lille som mulig:

$$\sum_{i=1}^{n} \min_{j=1,\ldots,k} \|\boldsymbol{x}_i - \boldsymbol{c}_j\|_2^2$$

Torben Tvedebrink
tvede@math.aau.dk

K-means

K-means algoritmen er meget simpel og beror på en opdeling af data i K clustre hvor *within* kvadratsummen ønskes så lille som mulig:

$$\sum_{i=1}^{n} \min_{j=1,\ldots,k} \|\boldsymbol{x}_i - \boldsymbol{c}_j\|_2^2$$

Algoritme:

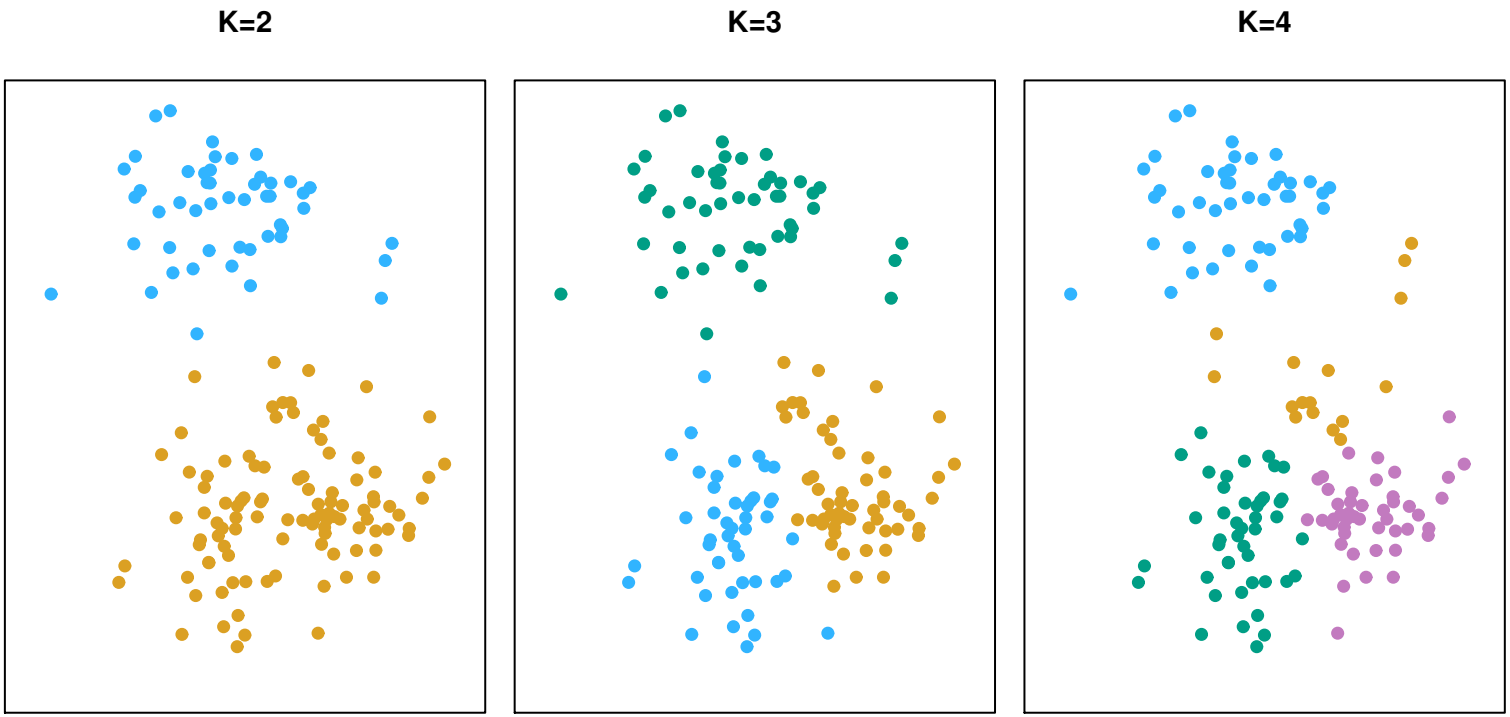
0.1 Vælg antal clustre K

0.2 Vælg tilfældigt K cluster repræsentanter tilfældigt (fx. K data punkter)

1. Hver observation, $\boldsymbol{x}_i$, allokeres til nærmeste center, $\boldsymbol{c}_j$.
2. Opdater $\boldsymbol{c}_j = \frac{1}{n_j} \sum_{j=1}^{n_j} \boldsymbol{x}_{i_j}$, hvor $\boldsymbol{x}_{i_1}, \ldots, \boldsymbol{x}_{i_{n_j}}$ tilhører kluster j.
3. Gentag 1-2 indtil konvergens (ingen ændring i allokering).

Torben Tvedebrink
tvede@math.aau.dk

Antal clustere?



Torben Tvedebrink
tvede@math.aau.dk

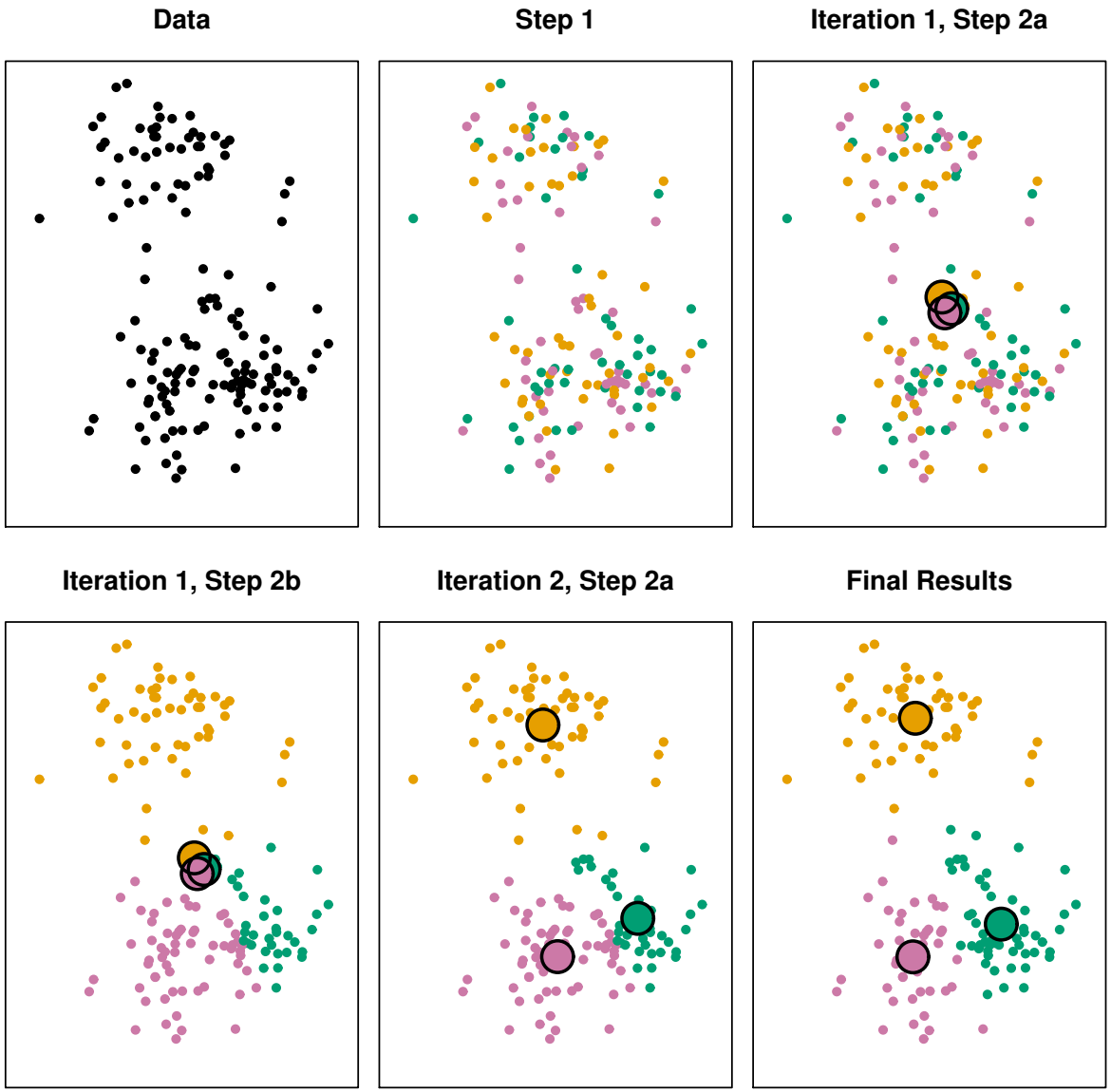
Problemer med K-means

- Skal beslutte antal clustre, K, før analyse
- Følsom overfor start værdier (se simulations eksempel)
- Følsom over for outliers
- ...

Torben Tvedebrink
tvede@math.aau.dk

Algoritmens basale skridt

ISLR, Figure 10.6, p. 389



Torben Tvedebrink
tvede@math.aau.dk

Simuleret data

Simulér data fra fire bivariat normal fordelinger med hvert sit centrum, $\boldsymbol{\mu}_i$, og varians, $\sigma_i^2 I_2$.

Som i bogen (pp. 27-29) vælges

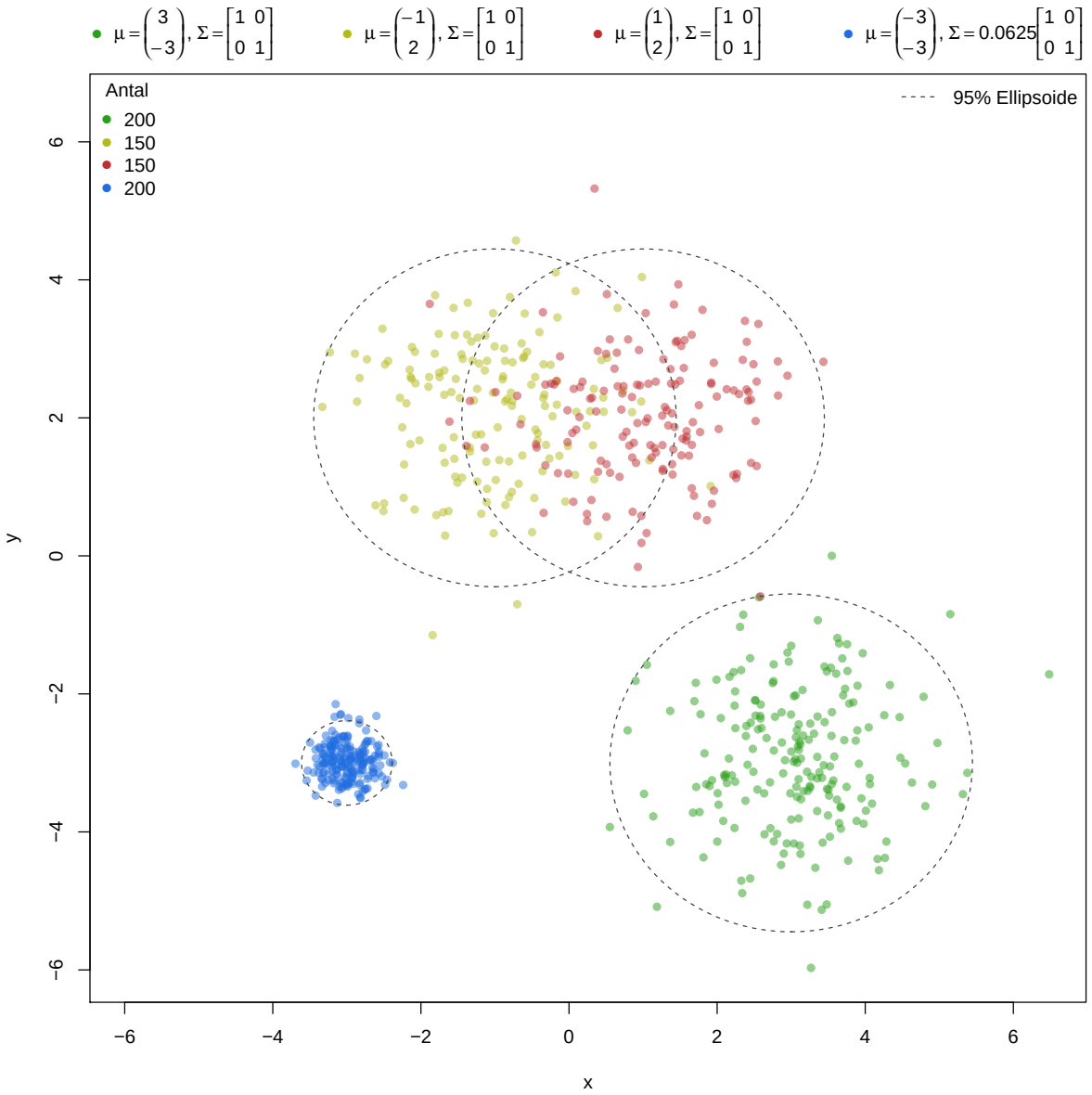
$$\boldsymbol{\mu}_1 = \begin{pmatrix} -3 \\ -3 \end{pmatrix} \quad \boldsymbol{\mu}_2 = \begin{pmatrix} 3 \\ -3 \end{pmatrix} \quad \boldsymbol{\mu}_3 = \begin{pmatrix} -1 \\ 2 \end{pmatrix} \quad \boldsymbol{\mu}_4 = \begin{pmatrix} 1 \\ 2 \end{pmatrix},$$

$$\sigma_1^2 = 0.0625 \quad \sigma_2^2 = 1 \quad \sigma_3^2 = 1 \quad \sigma_4^2 = 1$$

$$n_1 = 200 \quad n_2 = 200 \quad n_3 = 150 \quad n_4 = 150.$$

Torben Tvedebrink
tvede@math.aau.dk

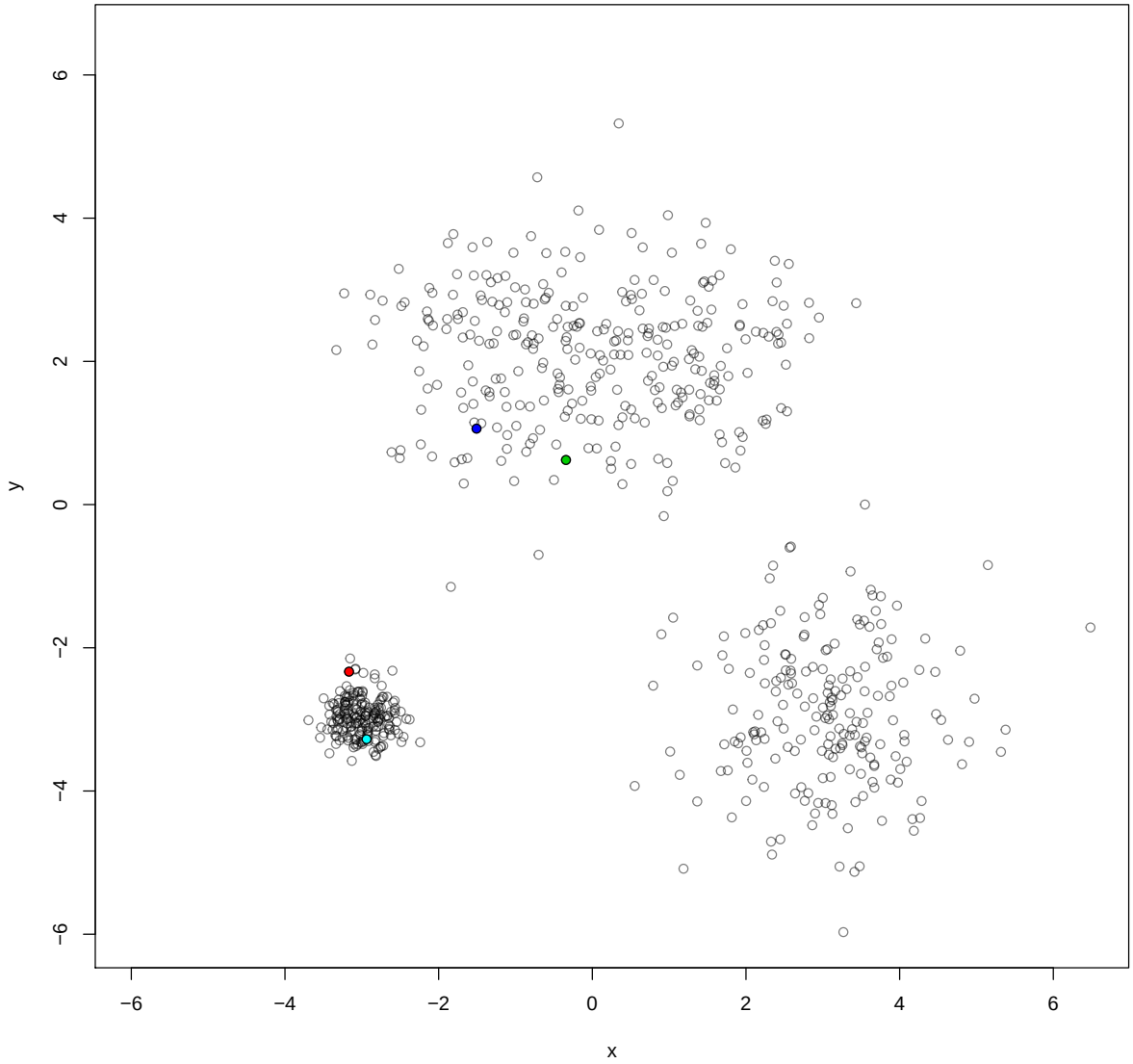
Simuleret data



Torben Tvedebrink
tvede@math.aau.dk

K-means på simulationerne

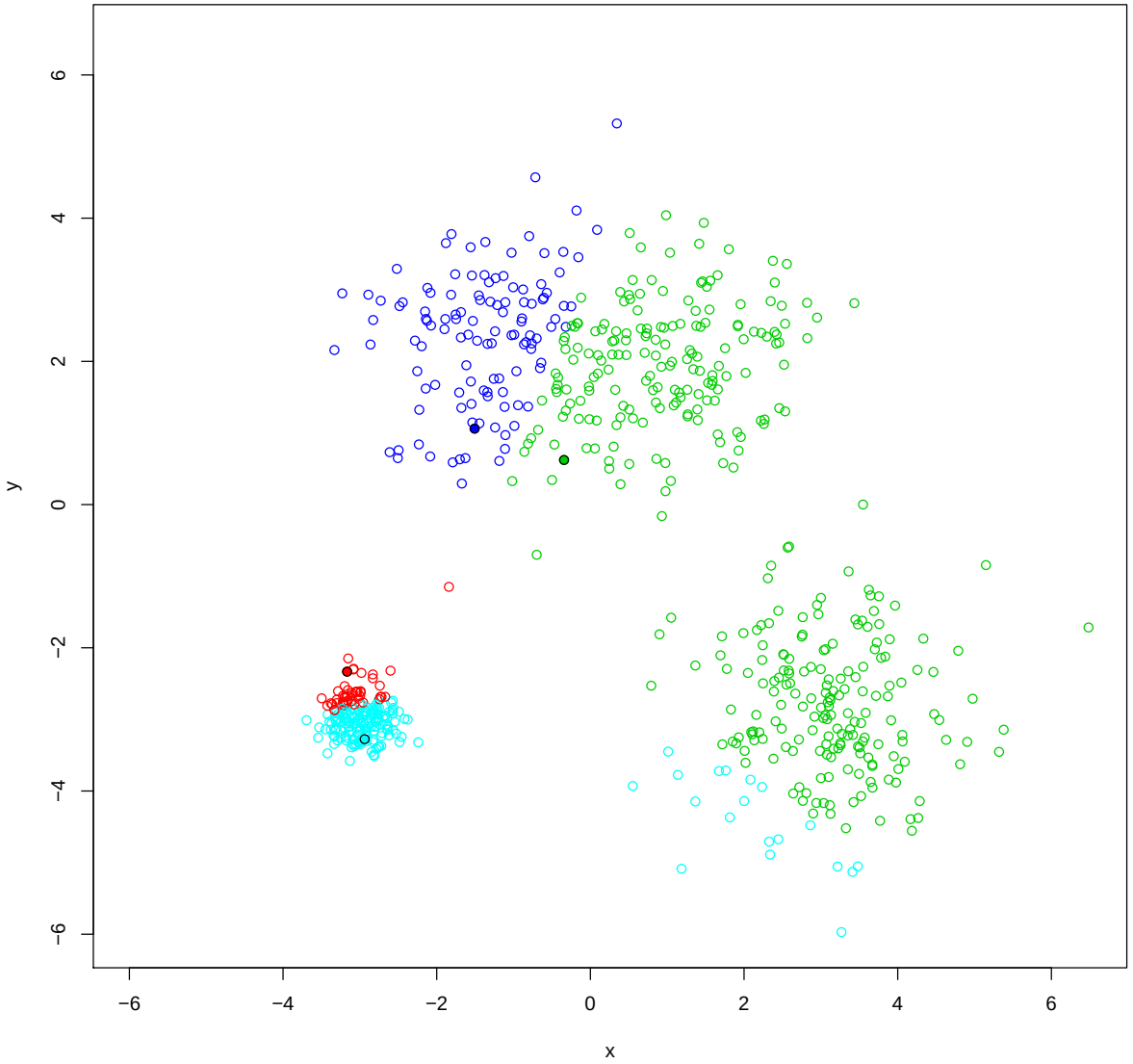
Begyndelsesværdier 1



Torben Tvedebrink
tvede@math.aau.dk

K-means på simulationerne

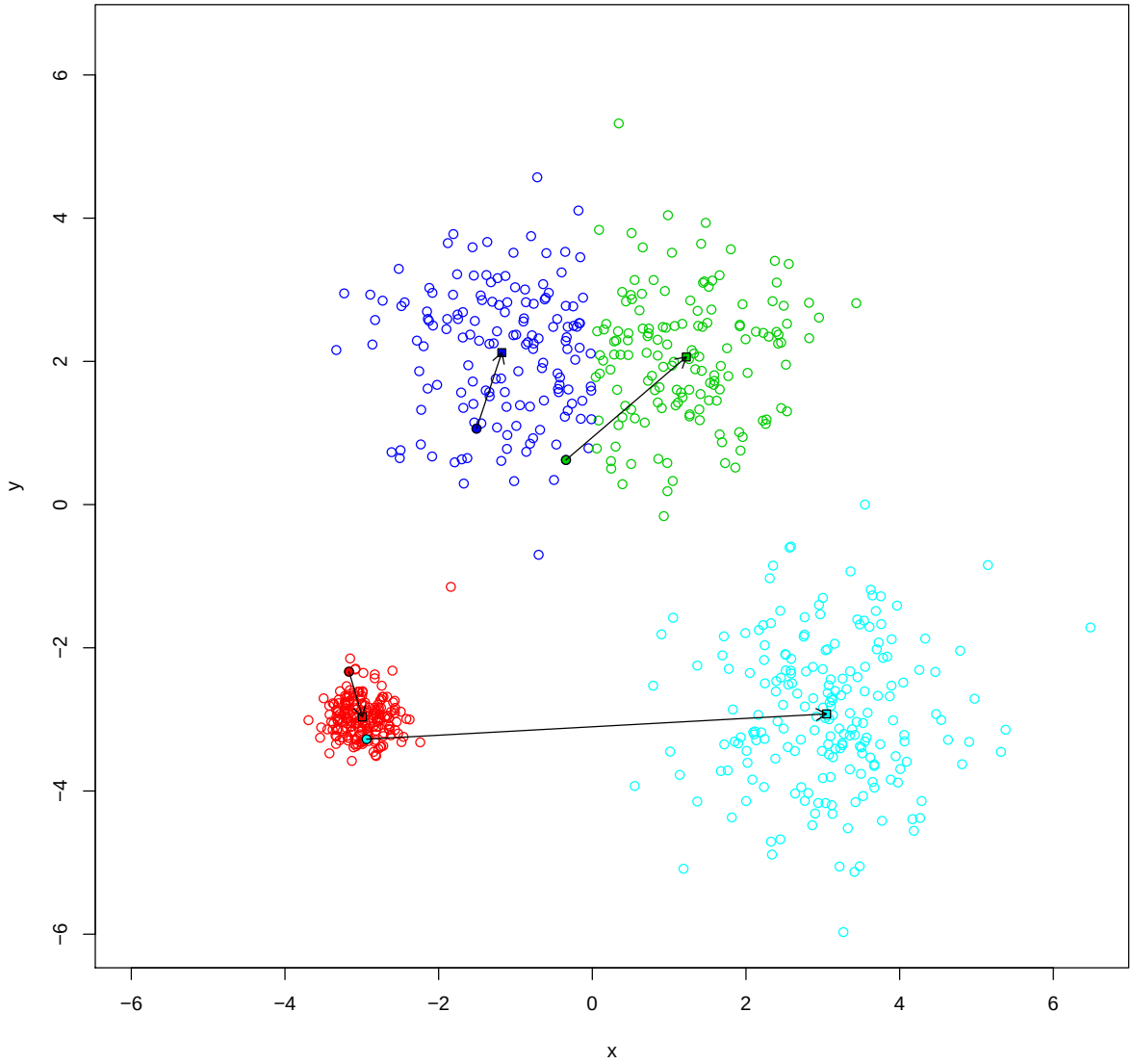
Begyndelsesværdier 1



Torben Tvedebrink
tvede@math.aau.dk

K-means på simulationerne

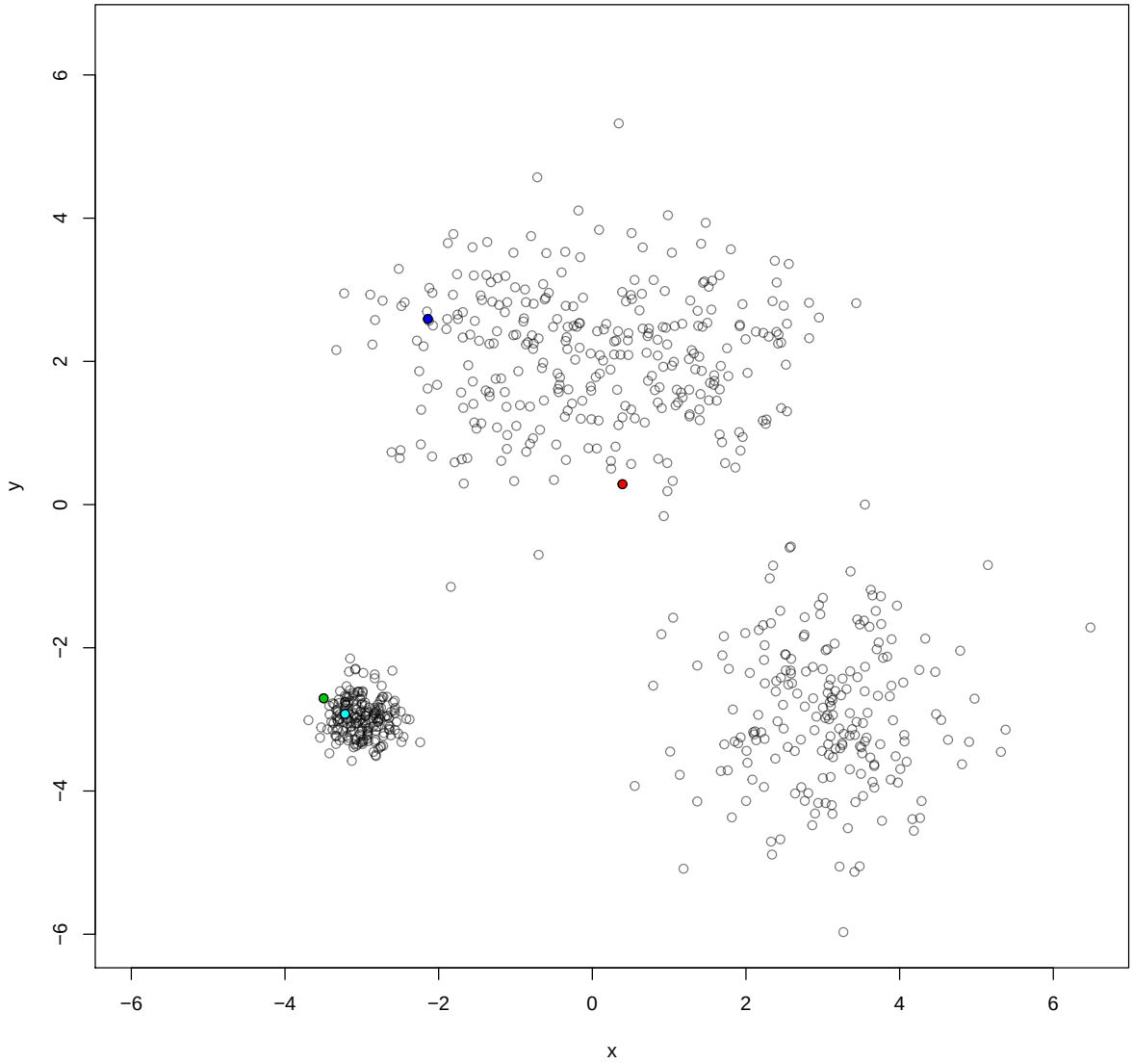
Begyndelsesværdier 1



Torben Tvedebrink
tvede@math.aau.dk

K-means på simulationerne

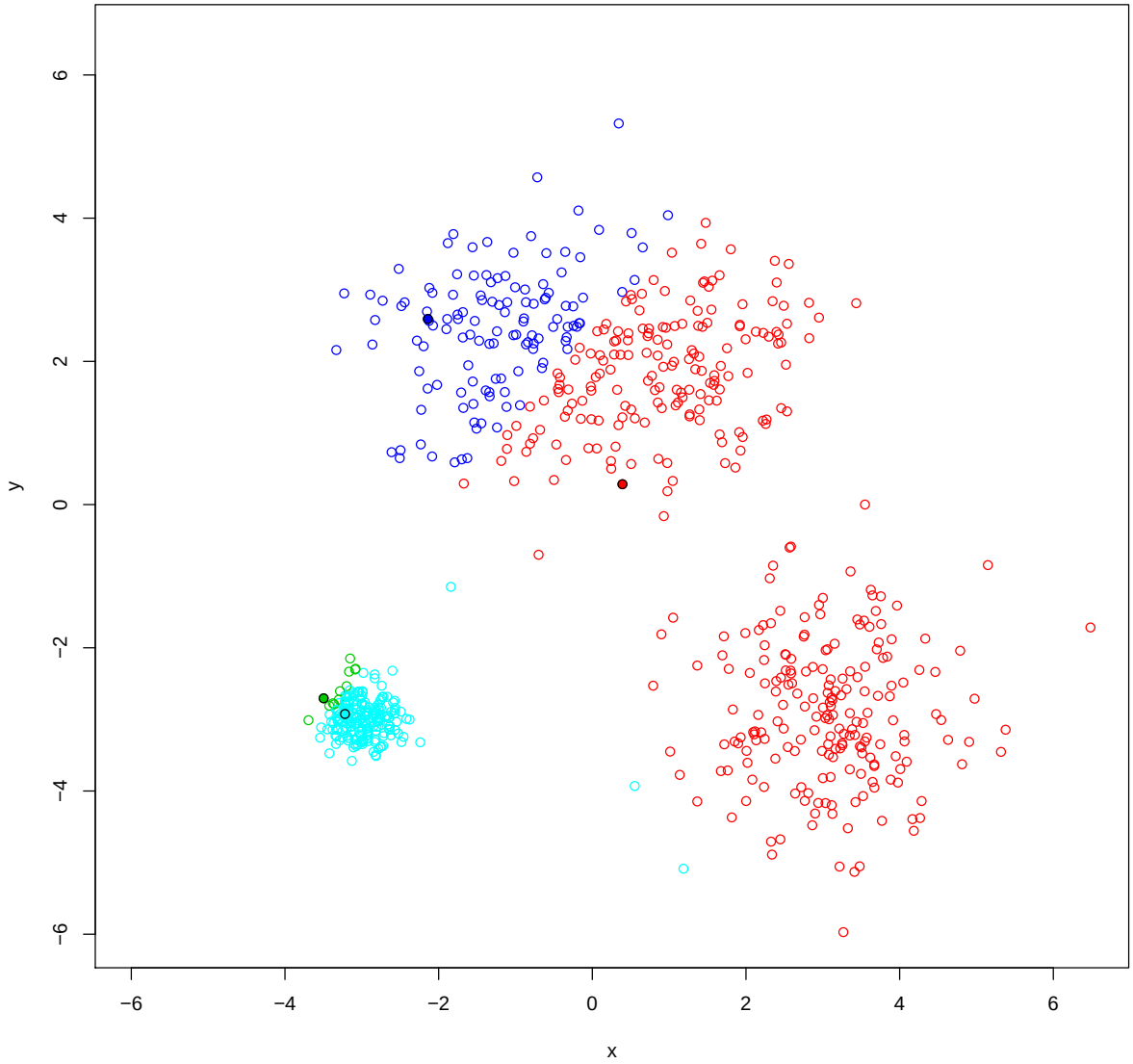
Begyndelsesværdier 2



Torben Tvedebrink
tvede@math.aau.dk

K-means på simulationerne

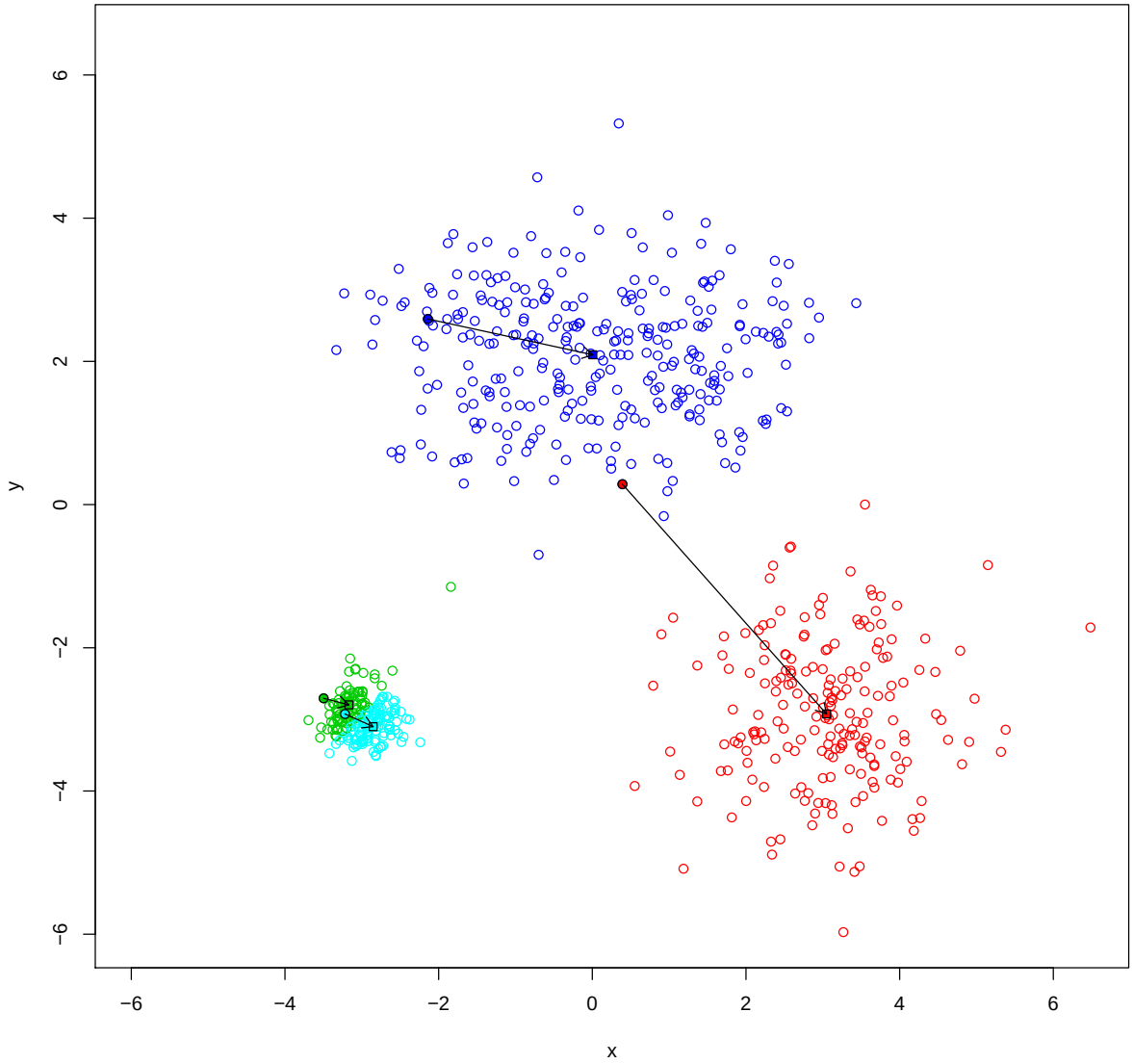
Begyndelsesværdier 2



Torben Tvedebrink
tvede@math.aau.dk

K-means på simulationerne

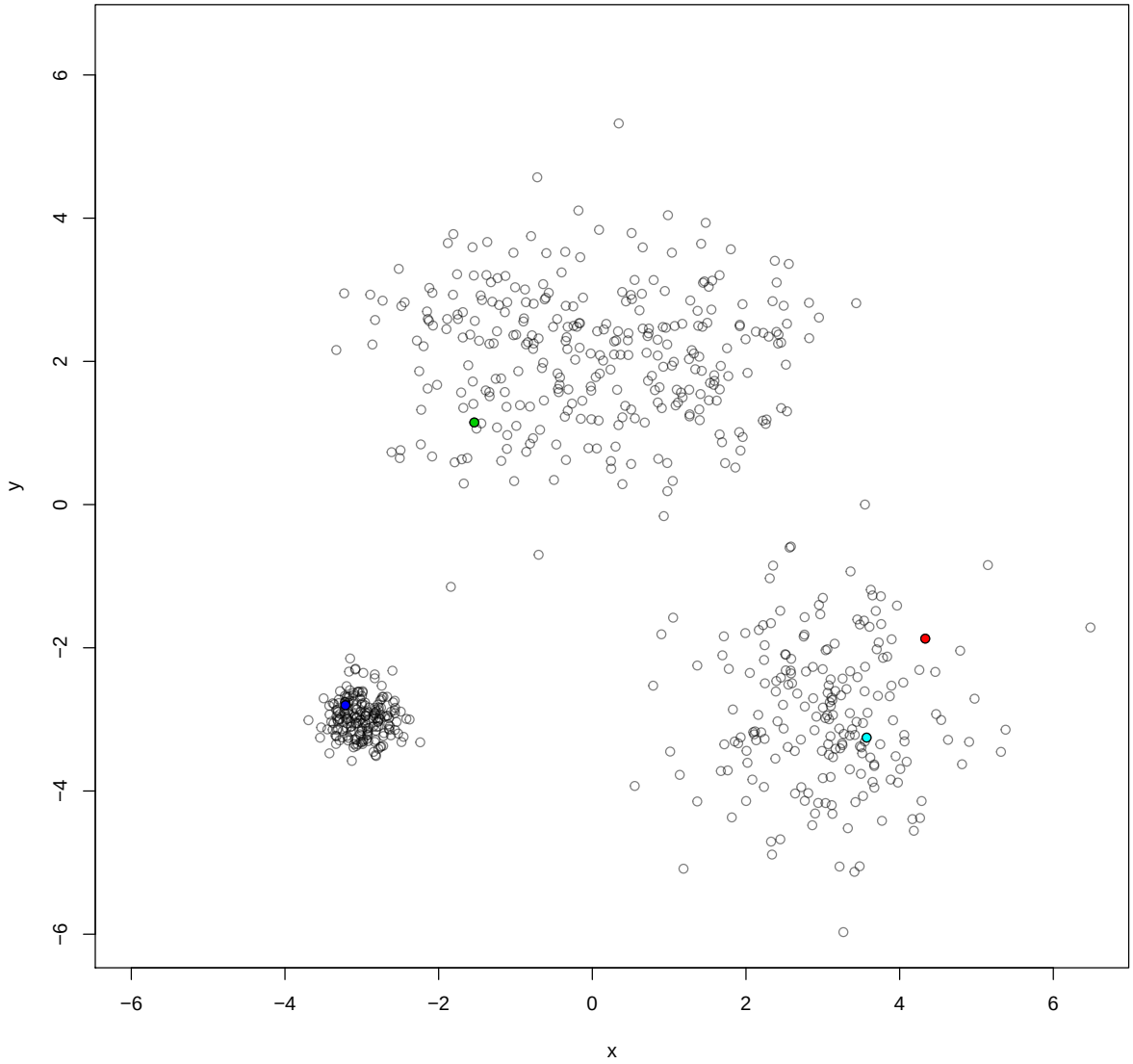
Begyndelsesværdier 2



Torben Tvedebrink
tvede@math.aau.dk

K-means på simulationerne

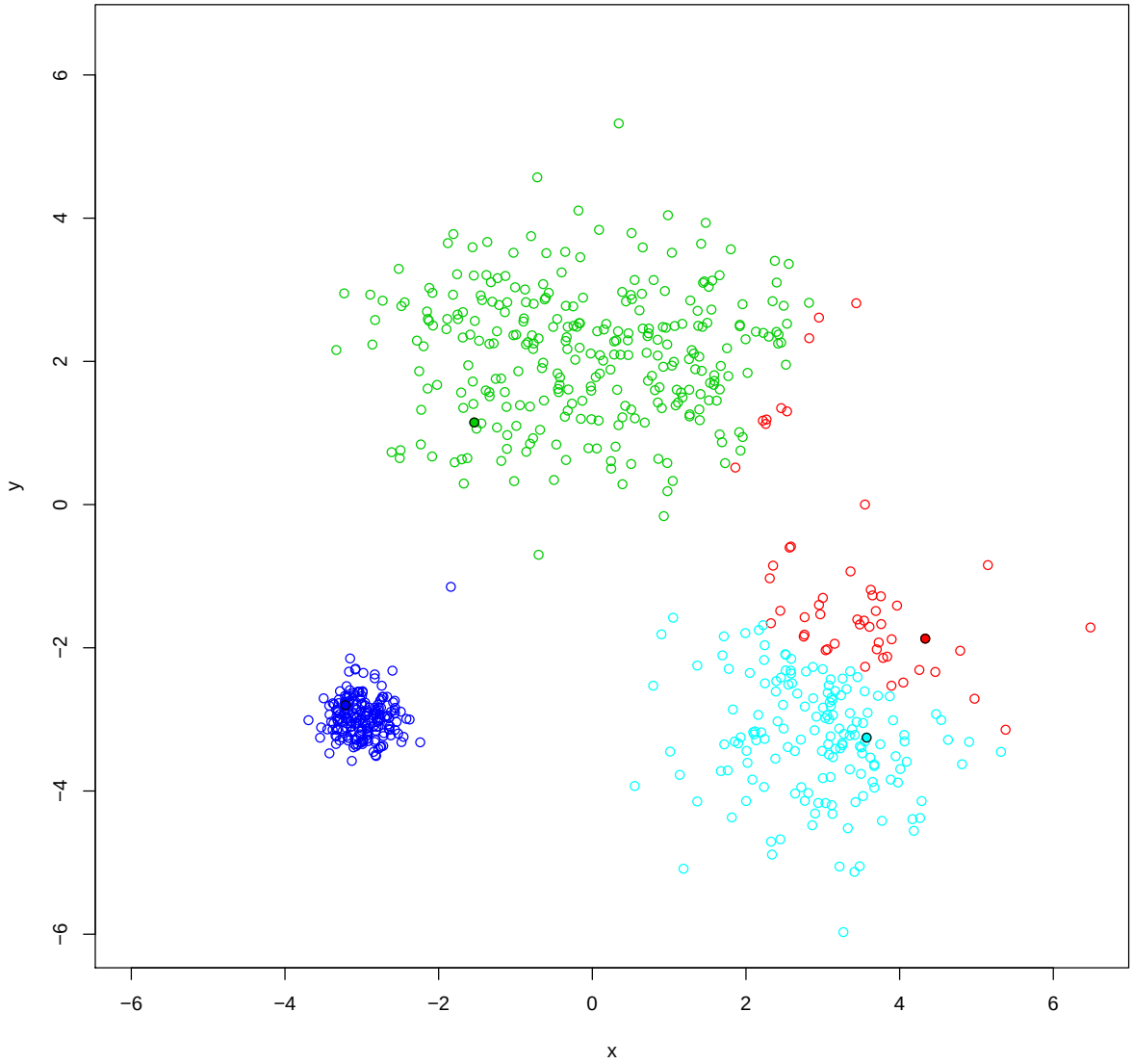
Begyndelsesværdier 3



Torben Tvedebrink
tvede@math.aau.dk

K-means på simulationerne

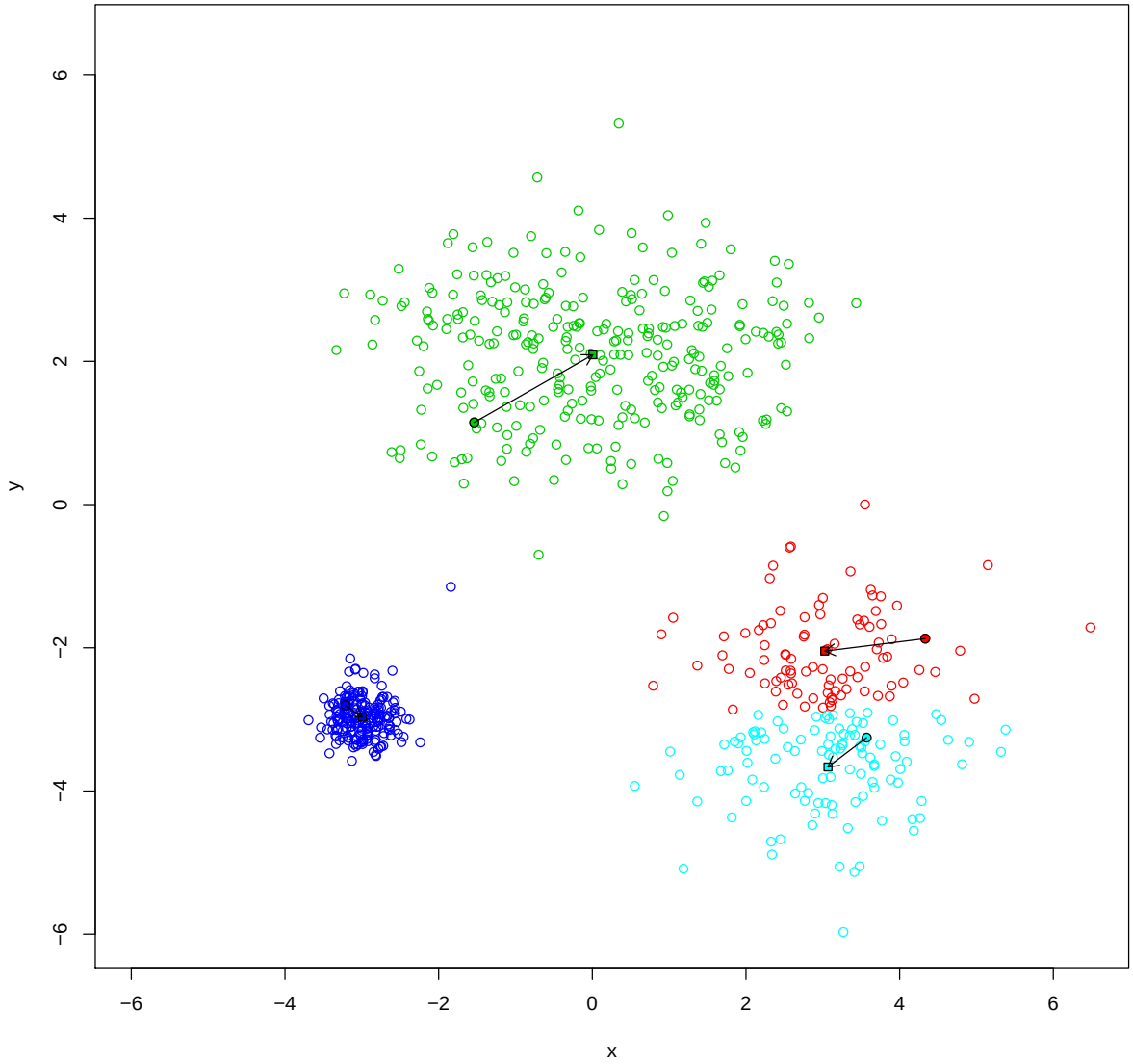
Begyndelsesværdier 3



Torben Tvedebrink
tvede@math.aau.dk

K-means på simulationerne

Begyndelsesværdier 3



Torben Tvedebrink
tvede@math.aau.dk

K-means i R

K-means er tilgængelig i R uden yderligere pakker ved at bruge `kmeans`. Se på dokumentationen for `kmeans` for at gøre jer bekendte med argumenterne.

For eksempel, man kan køre K-means med ti tilfældige sæt af start værdier ved at angive `nstart = 10`.

Vi kan tilgå en totale variabilitet og variabiliteten forklaret ved "modellen" fra outputet fra `kmeans`:
`"totss"` `"withinss"` `"tot.withinss"` `"betweenss"`

Torben Tvedebrink
tvede@math.aau.dk

Sums of squares
Dekomponere variationen i data

I statistik ønsker vi at forklare så meget af variabiliteten ved systematiske komponenter. I clustering er de systematiske komponenter cluster tilhørsforholdet.

Torben Tvedebrink
tvede@math.aau.dk

Sums of squares

Dekomponere variationen i data: SS_{TOT}, SS_W og SS_B

Lad x_i være en skalar (fx. reelle tal – argumentet holder også for vektorer) så har vi:

$$\bar{x} = n^{-1} \sum_{i=1}^{n} x_i \quad \text{og} \quad \bar{x}_k = |C_k|^{-1} \sum_{i \in C_k} x_i$$

Den *totale sum of squares* og *within sums of squares* er da givet ved

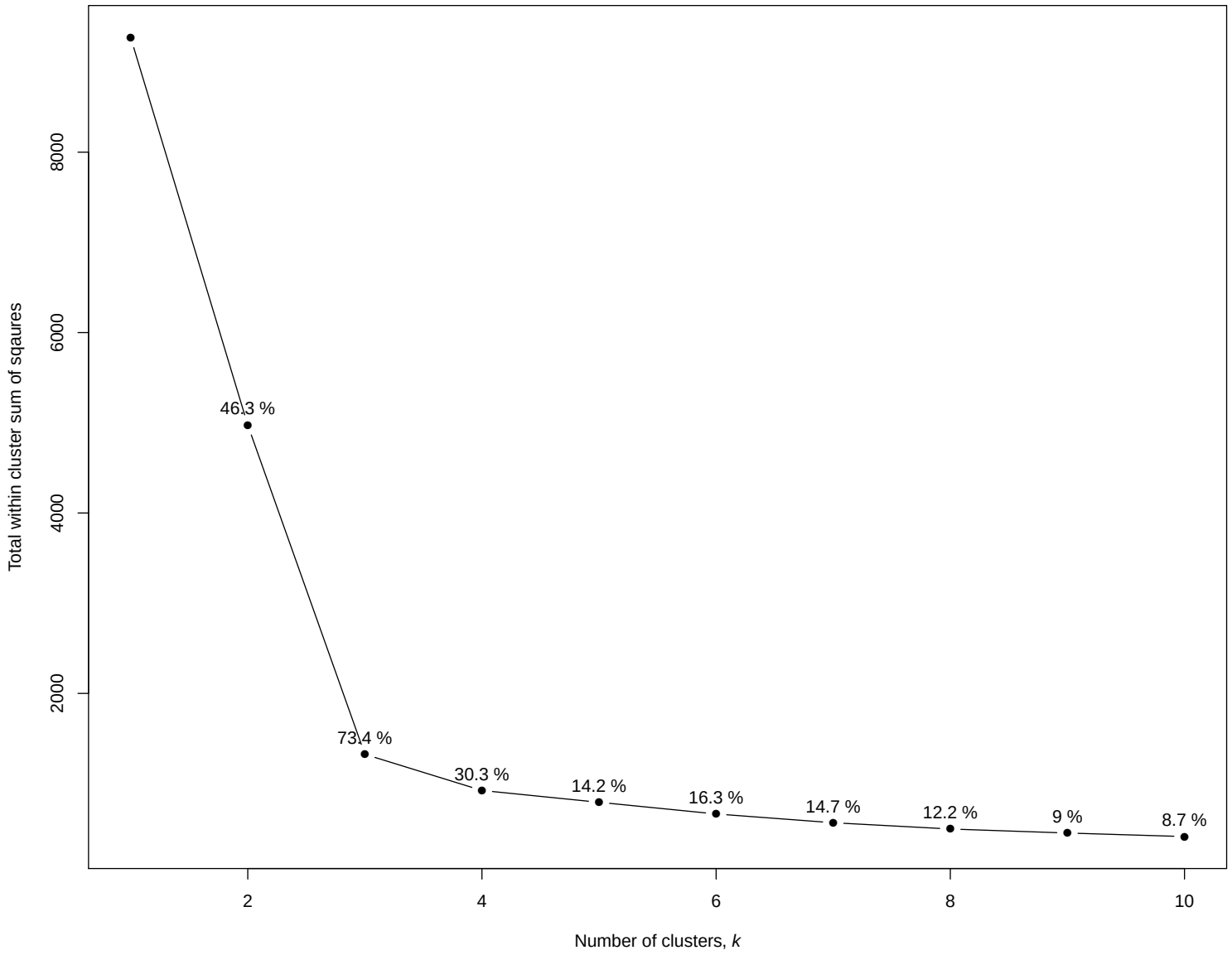
$$SS_{TOT} = \sum_{i=1}^{n} (x_i - \bar{x})^2 \quad \text{og} \quad SS_{W_k} = \sum_{i \in C_k} (x_i - \bar{x}_k)^2$$

Ydermere, *between sums of squares* er defineret som

$$SS_W = \sum_{k=1}^{K} SS_{W_k} \quad \text{and} \quad SS_B = SS - SS_W$$

Torben Tvedebrink
tvede@math.aau.dk

Sums of squares
Simulations eksemplet fortsat



Torben Tvedebrink
tvede@math.aau.dk

CH Index

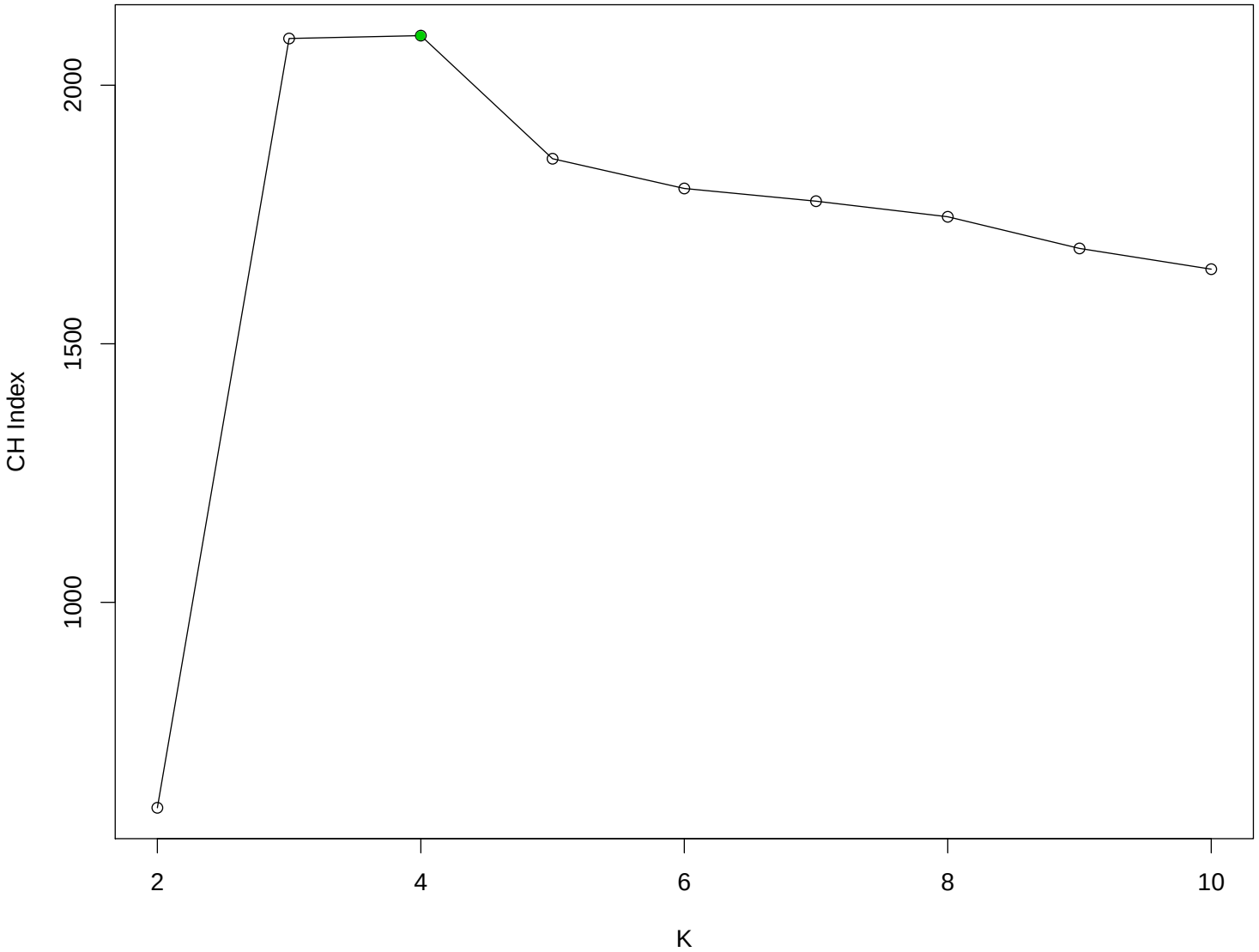
Til at vælge antal clustre foreslog Calinski and Harabasz (1974), "A dendrite method for cluster analysis" dette index, som ønskes størst muligt:

$$CH(K) = \frac{B(K)/(K-1)}{W(K)/(n-K)} = \frac{SS_B/(K-1)}{SS_W/(n-K)}$$

hvor $B(K)$ og $W(K)$ er hhv. between-sum-of-squares og within-sum-of-squares for K clustre.

Torben Tvedebrink
tvede@math.aau.dk

CH Index
Simulations eksempel fortsat



Torben Tvedebrink
tvede@math.aau.dk

Kør modellen flere gange

SS_W, ISLR, Figure 10.7, p. 390

Torben Tvedebrink
tvede@math.aau.dk

K-means alternativer
K-mediods

I stedet for at bruge gennemsnittet som centre, $\boldsymbol{c}_j$, kan medianen bruges (K-mediods, tilgængeligt via `pam` i `cluster`-pakken).

Torben Tvedebrink
tvede@math.aau.dk

K-means alternativer

X-means

I stedet for at bruge gennemsnittet som centre, $\boldsymbol{c}_j$, kan medianen bruges (K-mediods, tilgængeligt via `pam` i `cluster`-pakken).

En anden metode, X-means, består i at bruge et Informations Kriterie til at bestemme hvor mange klustre data understøtter, dvs. estimere K på baggrund af hvor godt data fitter forskellige modeller. Scoren/kvadratsummen straffes for flere klustre/parametre vha. BIC ved $\frac{q}{2} \log n$, hvor q er antal parametre i modellen og n er antal observationer.

Torben Tvedebrink
tvede@math.aau.dk

K-means alternativer
Soft clustering

I stedet for at bruge gennemsnittet som centre, $\boldsymbol{c}_j$, kan medianen bruges (K-mediods, tilgængeligt via `pam` i `cluster`-pakken).

I modsætning til K-means som udfører *hard clustering* kan man benytte *soft clustering*. Her bliver hvert datapunkt tilskrevet et cluster med en given sandsynlighed – hvor *hard clustering* angiver "sandsynligheder" svarende til 0 og 1.

Vi vender tilbage til dette i lektion 5. En god pakke til dette er `mclust` – med en udførlig vignette:

```
library(mclust)
vignette("mclust")
```


Torben Tvedebrink
tvede@math.aau.dk

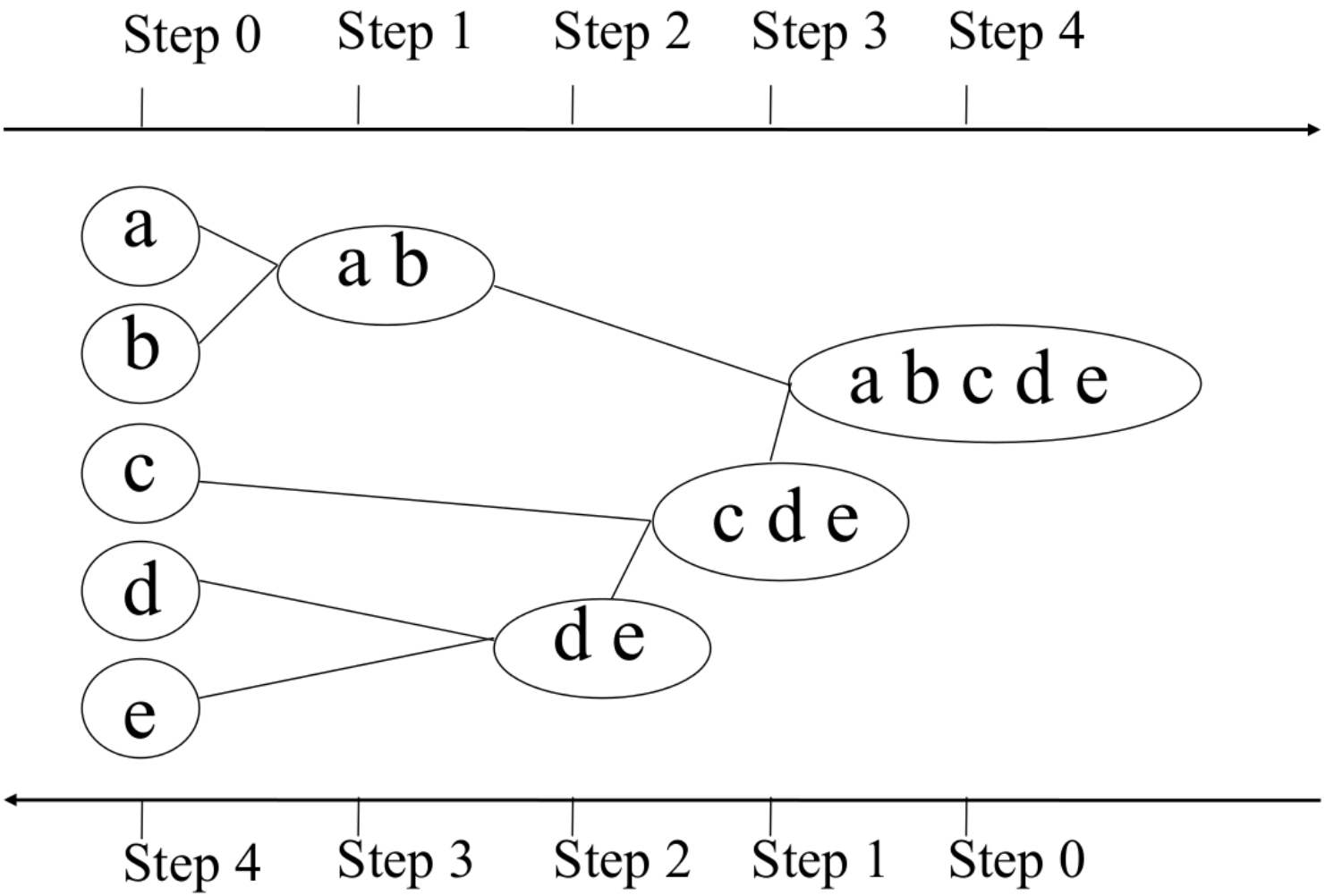
Hierarkiske clustre

- *Agglomerative*
  Buttom-up metode: Starter med n "clustre" og sammensmelter *nærmeste* clustre i hvert trin
- *Divisive*
  Stop-down metode: Starter med ét "cluster" og adskiller hvert cluster indtil n 'blade' er nået.

Torben Tvedebrink
tvede@math.aau.dk

Hierarkiske clustre

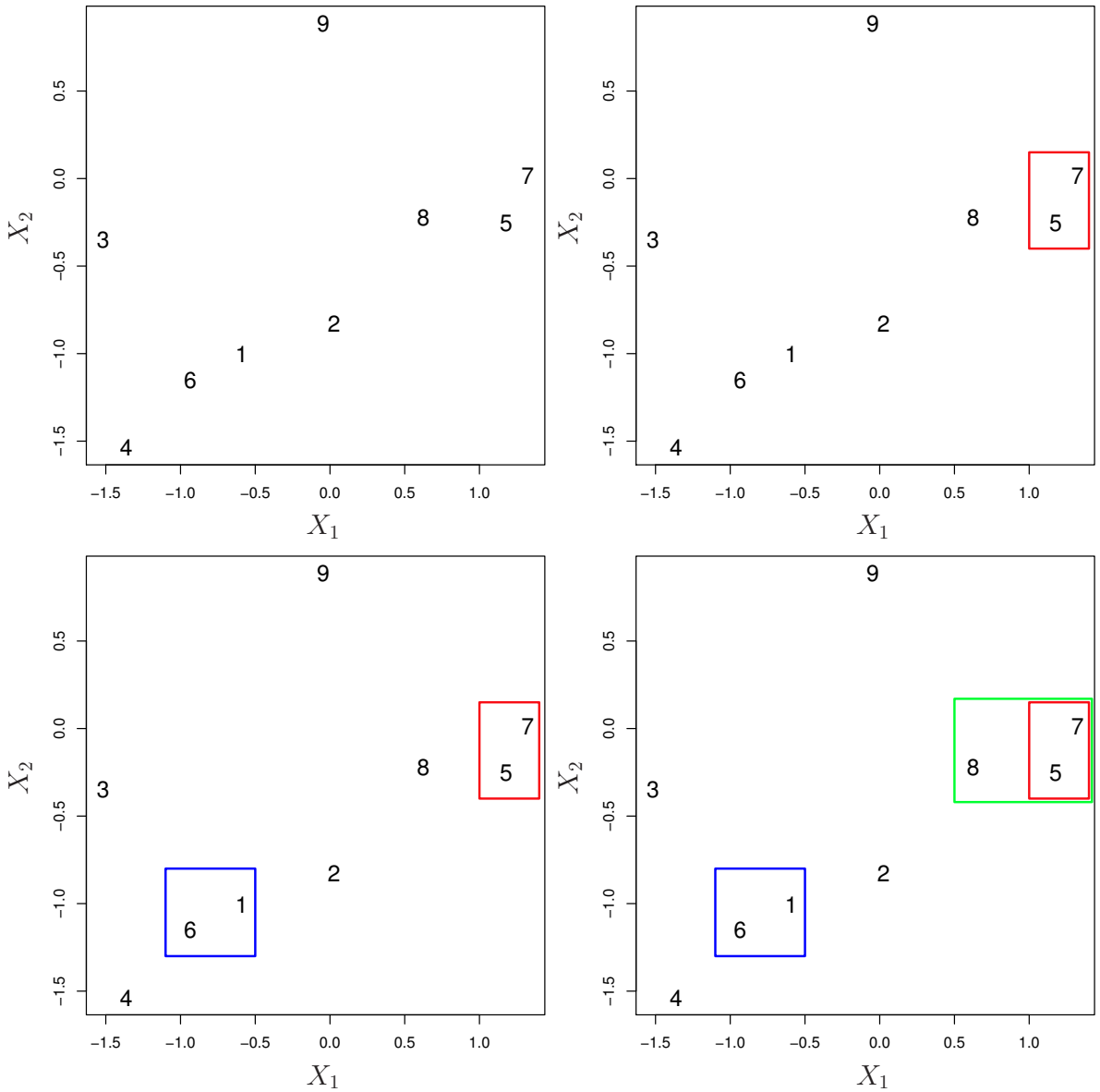
Agglomerative



Divisive

Torben Tvedebrink
tvede@math.aau.dk

Eksempel
Complete link



Torben Tvedebrink
tvede@math.aau.dk

Dissimilarity mellem grupper
Linkages

- **Single linkage** (nærmeste nabo) - afstanden mellem de to nærmeste elementer fra forskellige clustre

$$d_{\text{Single linkage}}(C_k, C_{k'}) = \min_{i \in C_k, i' \in C_{k'}} d_{ii'}.$$

- **Complete linkage** (fjerneste nabo) - afstanden mellem de to fjerneste elementer fra forskellige clustre

$$d_{\text{Complete linkage}}(C_k, C_{k'}) = \max_{i \in C_k, i' \in C_{k'}} d_{ii'}.$$

- **Average linkage** - den gennemsnitlige afstand i mellem alle par af punkter i hvert cluster

$$d_{\text{Average linkage}}(C_k, C_{k'}) = \frac{1}{|C_k| \cdot |C_{k'}|} \sum_{i \in C_k, i' \in C_{k'}} d_{ii'}.$$

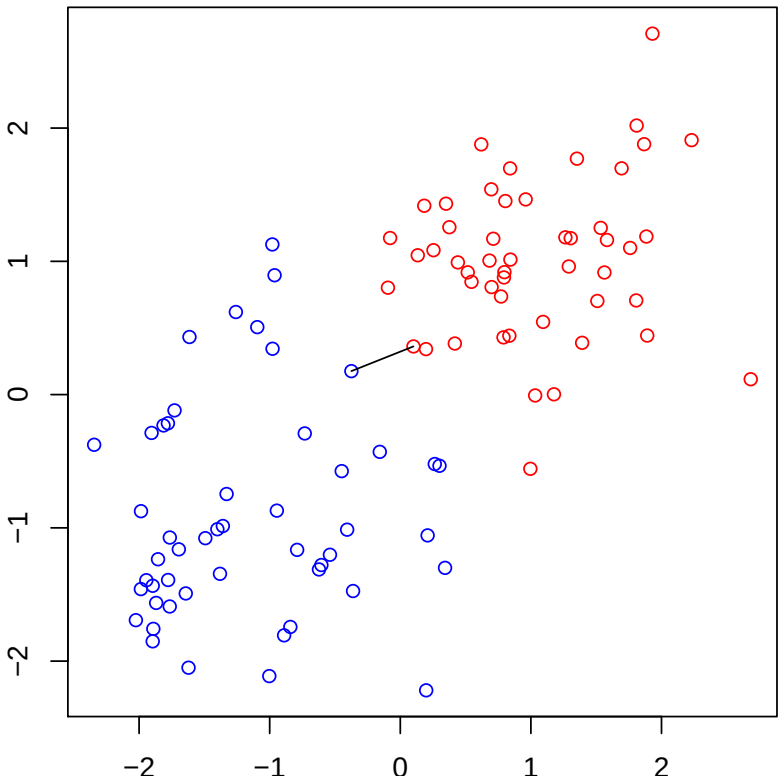
- **Ward's method** - minimerer SS_W når to clustre sammensmeltes.
- ...

Torben Tvedebrink
tvede@math.aau.dk

Single link

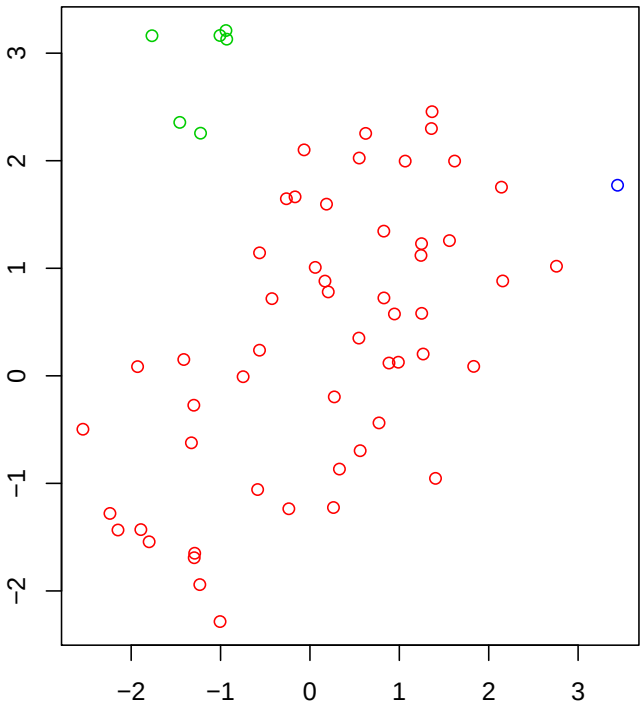
Single linkage (nærmeste nabo) - afstanden mellem de to nærmeste elementer fra forskellige clustre

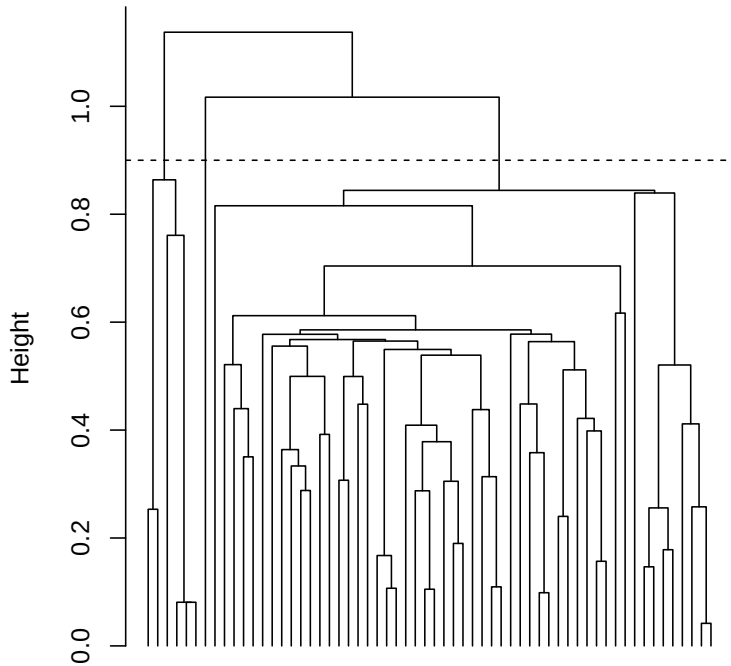
$$d_{\text{Single linkage}}(C_k, C_{k'}) = \min_{i \in C_k, i' \in C_{k'}} d_{ii'}.$$



Torben Tvedebrink
tvede@math.aau.dk

Single link
Eksempel



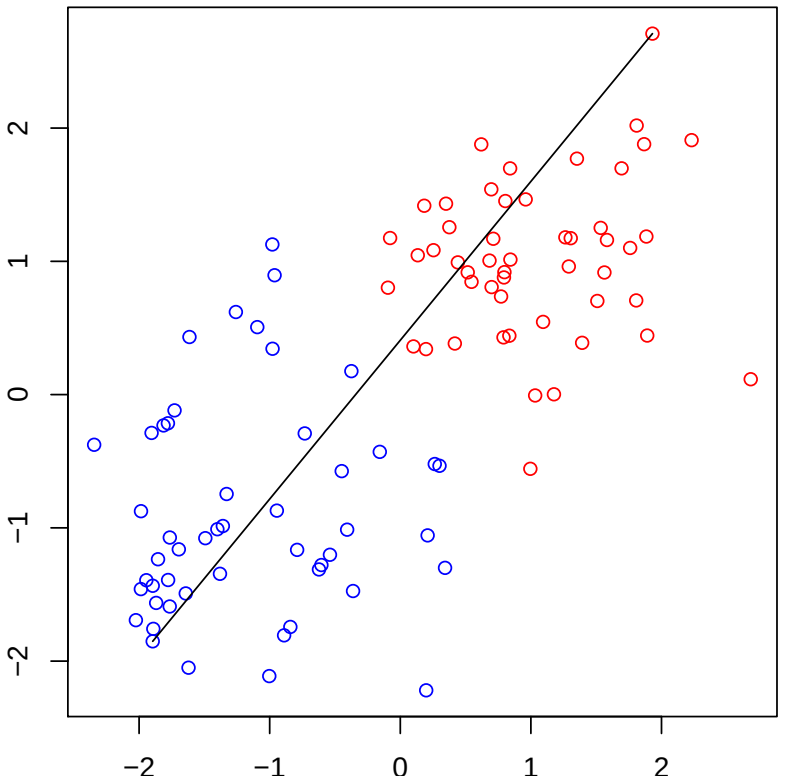


Torben Tvedebrink
tvede@math.aau.dk

Complete link

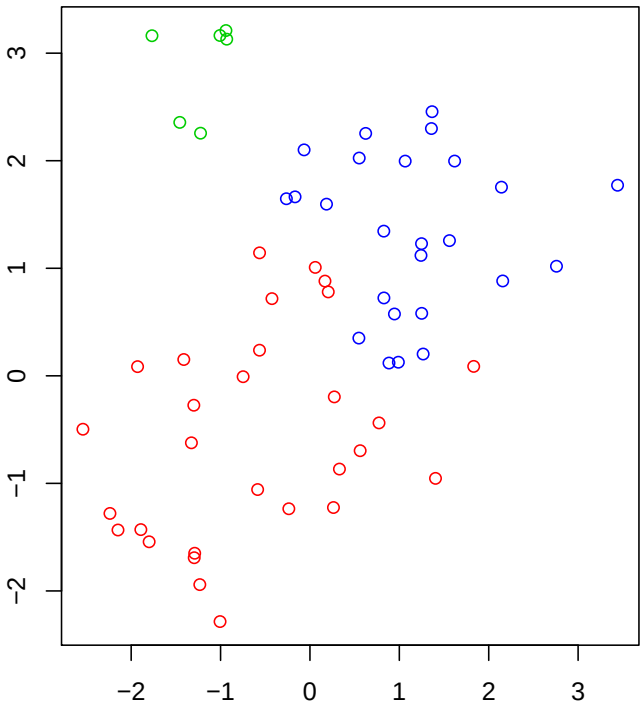
Complete linkage (fjerneste nabo) - afstanden mellem de to fjerneste elementer fra forskellige clustre

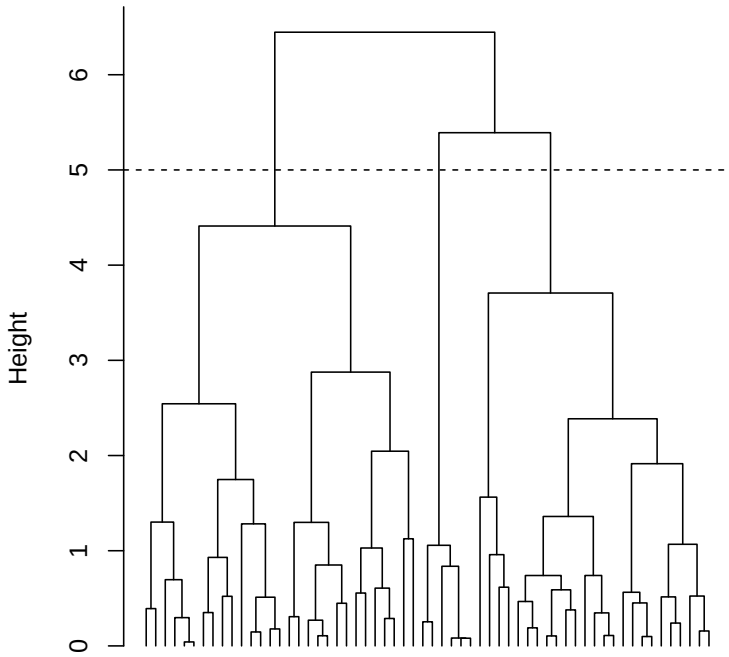
$$d_{\text{Complete linkage}}(C_k, C_{k'}) = \max_{i \in C_k, i' \in C_{k'}} d_{ii'}.$$



Torben Tvedebrink
tvede@math.aau.dk

Complete link
Eksempel



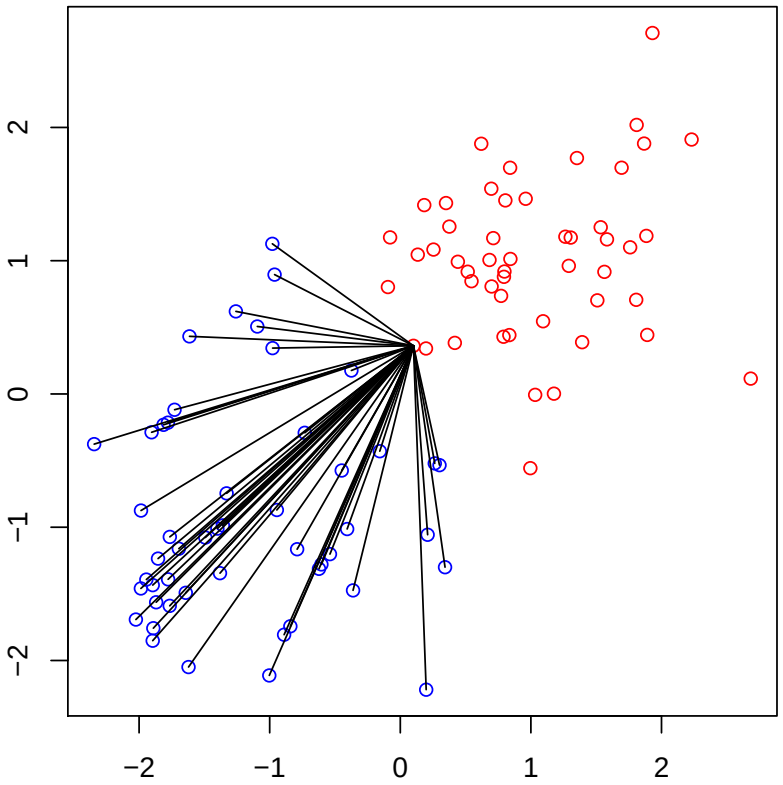


Torben Tvedebrink
tvede@math.aau.dk

Average link

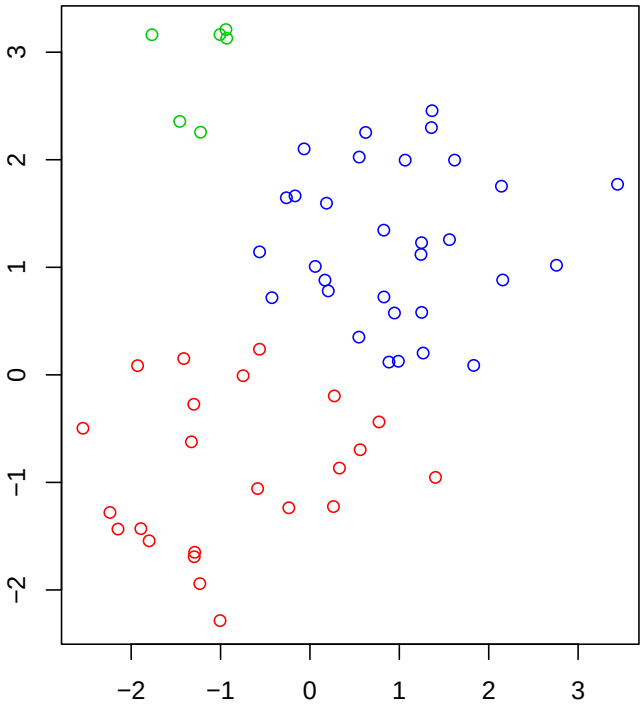
Average linkage - den gennemsnitlige afstand i mellem alle par af punkter i hvert cluster

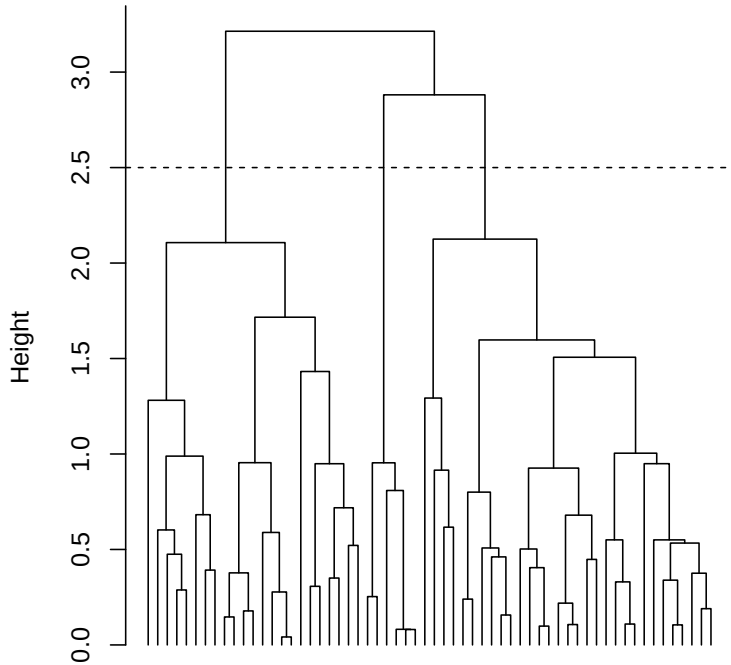
$$d_{\text{Average linkage}}(C_k, C_{k'}) = \frac{1}{|C_k| \cdot |C_{k'}|} \sum_{i \in C_k, i' \in C_{k'}} d_{ii'}.$$



Torben Tvedebrink
tvede@math.aau.dk

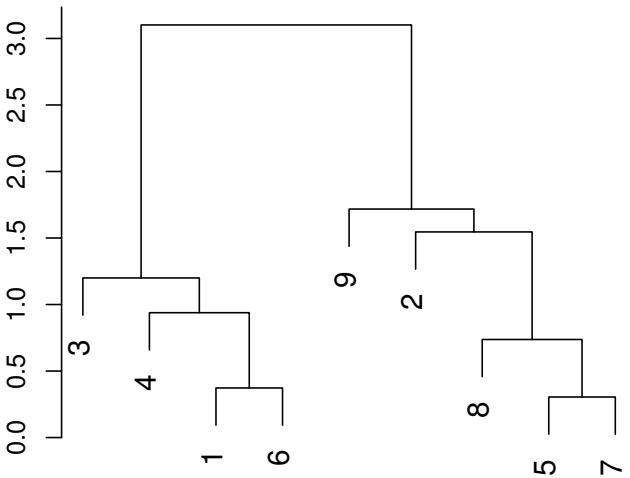
Average link
Eksempel

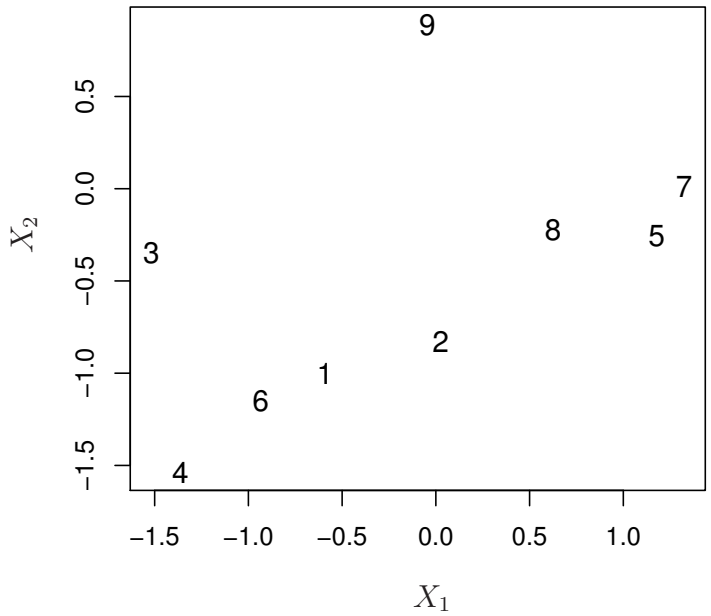




Torben Tvedebrink
tvede@math.aau.dk

Dendrogrammet





Torben Tvedebrink
tvede@math.aau.dk

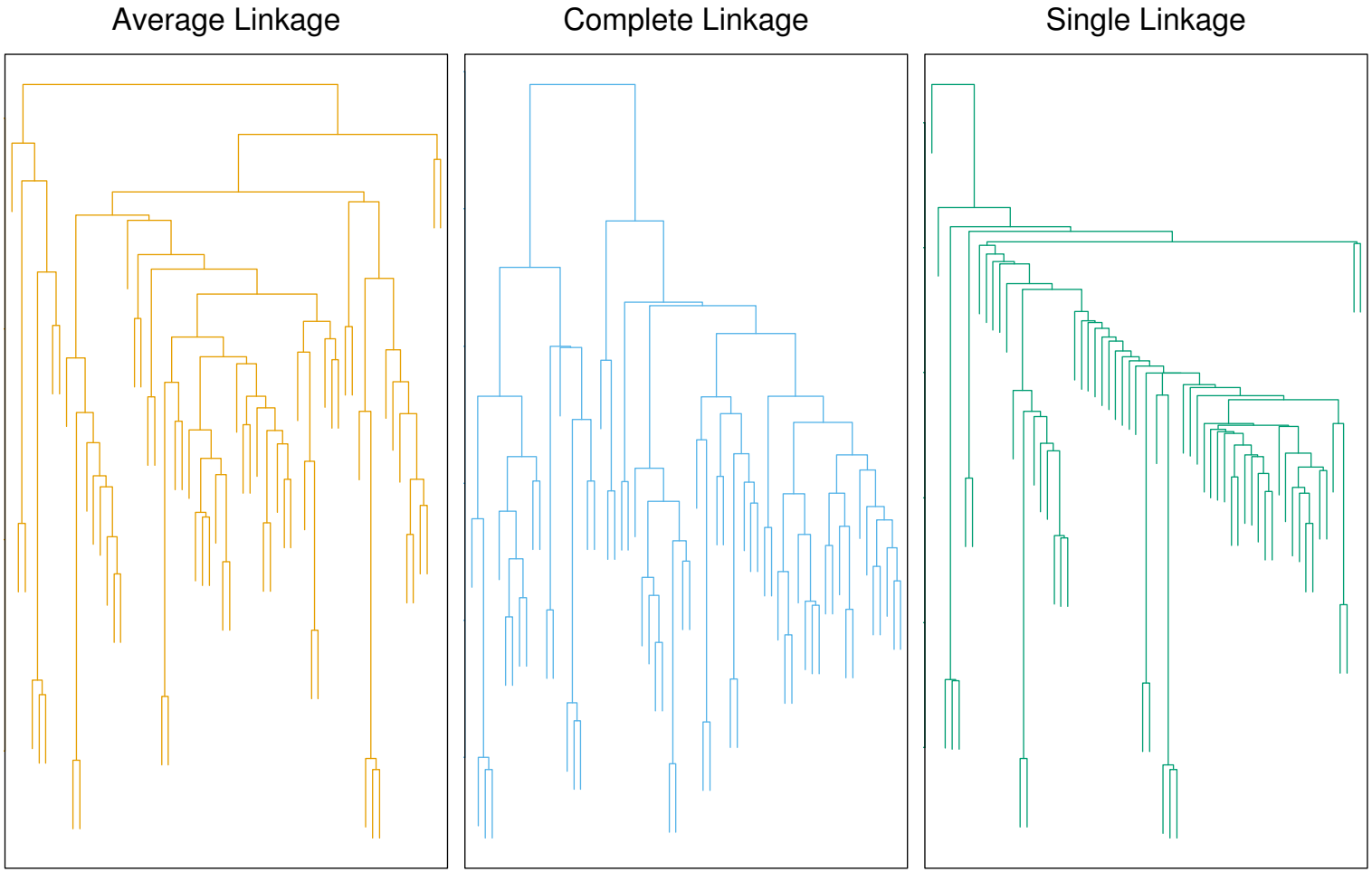
Linkages

Hvert af de foregående linkages giver anledning til forskellige clustre (nogen minder naturligvis om hinanden for simple eksempler).

Alt afhængigt af formålet med cluster analysen vil forskellige mål være at foretrække.

Torben Tvedebrink
tvede@math.aau.dk

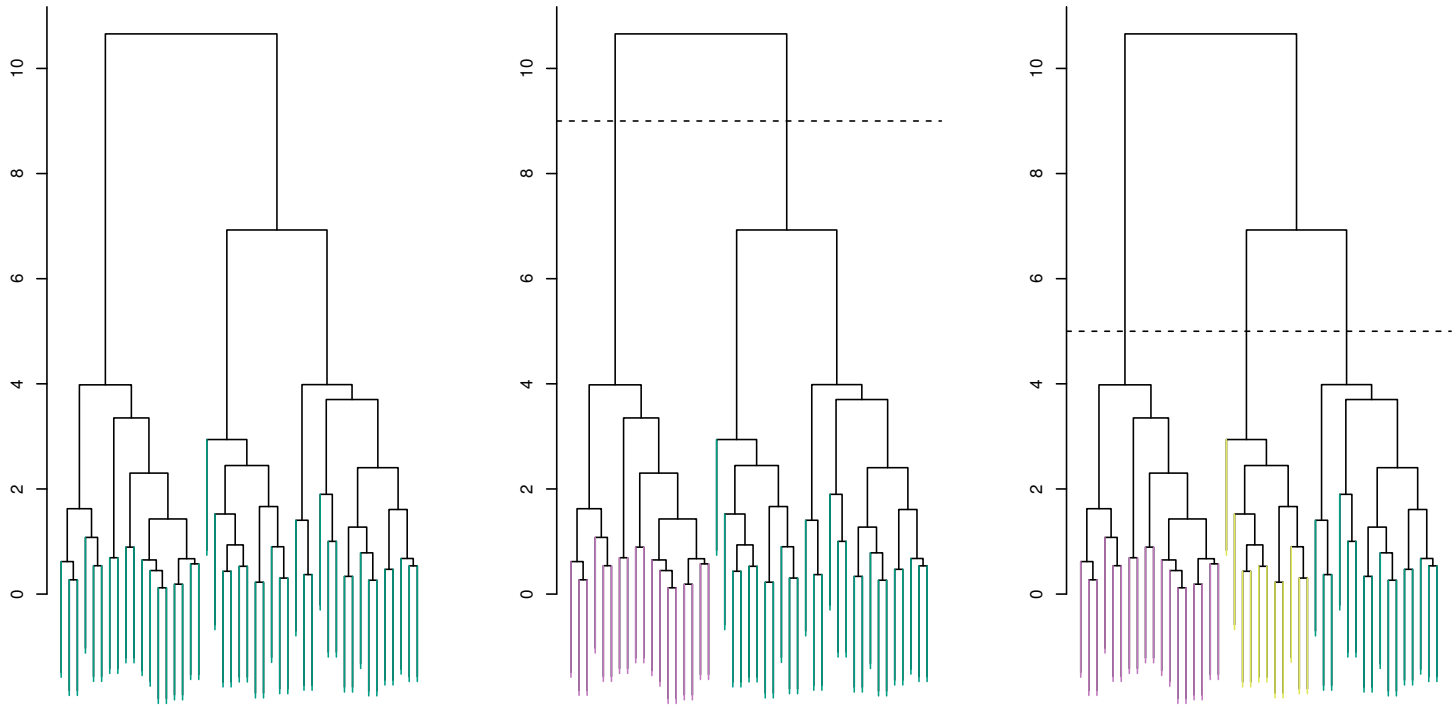
Linkages



Torben Tvedebrink
tvede@math.aau.dk

Partitionering af hierarkiske clustere

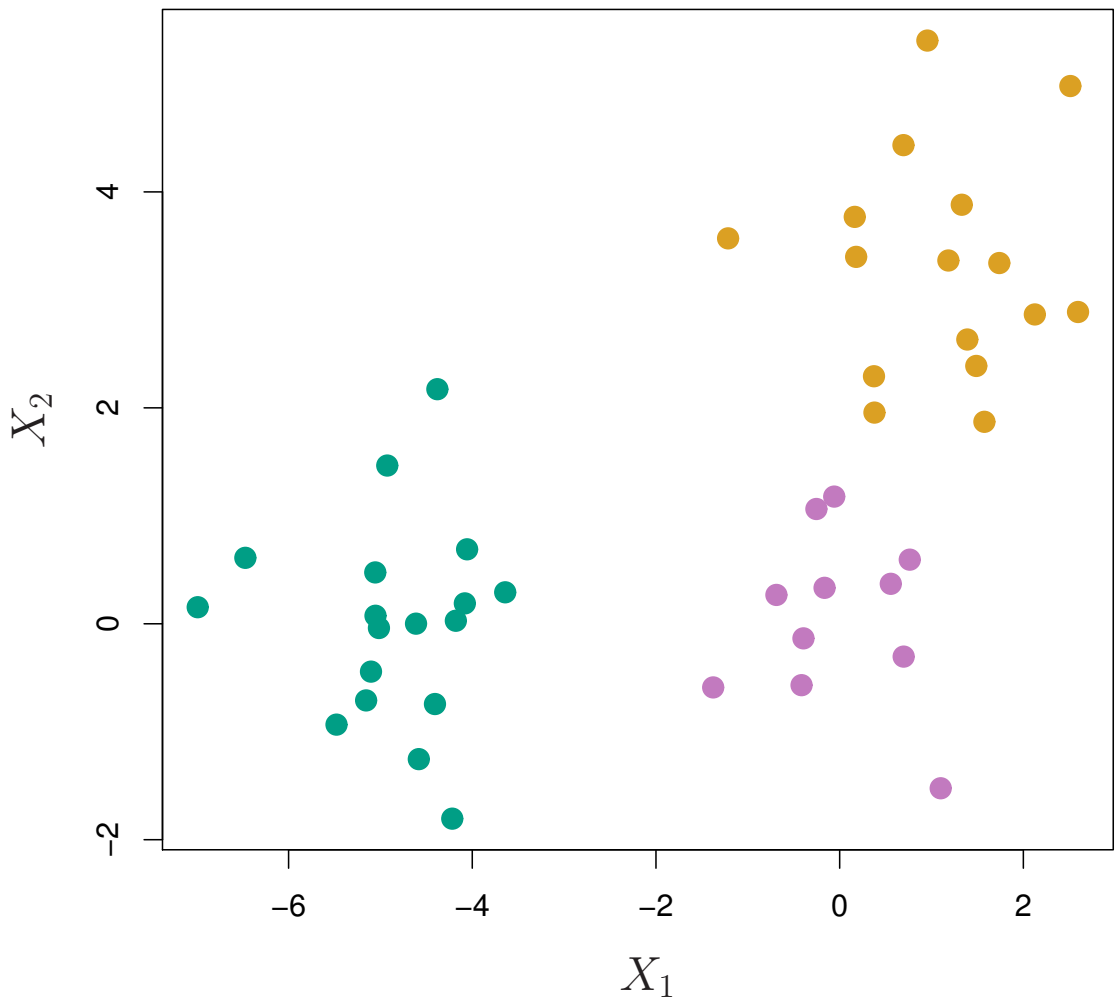
Forskellige cut højder



Torben Tvedebrink
tvede@math.aau.dk

Partitionering af hierarkiske clustere

Ved tre clustre



Torben Tvedebrink
tvede@math.aau.dk

Hierarkisk clustering i R

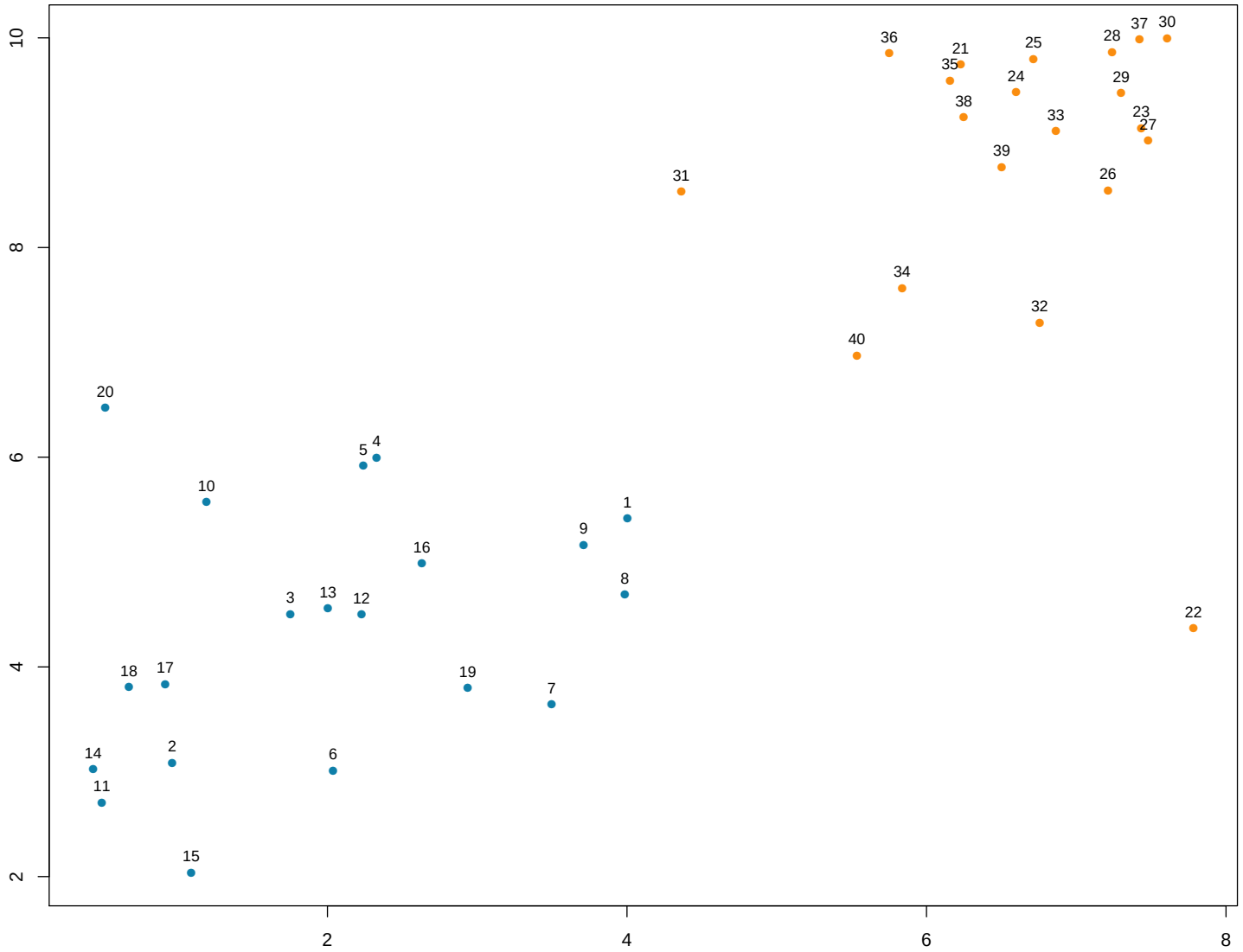
I R kan man lave hierarkisk clustering vha. `hclust` funktionen i R.

`hclust` tager følgende vigtige argumenter: `d` og `method`, hvor `d` er et `dist`-objekt fx. fra `dist`-funktionen. Det andet argument, `method`, refererer til linkage metoden, fx. `"complete", "single", "average", "ward.D2"`

Mange andre hierarkiske clustering metoder er tilgængelige via `cluster` pakken i R. Ved at bruge funktionerne fra denne pakke, se listen af funktioner og datasæts ved `help(package="cluster")`.

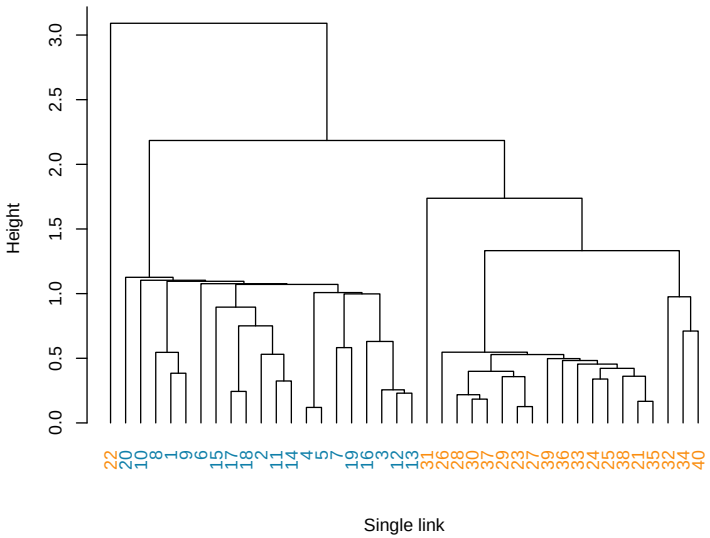
Torben Tvedebrink
tvede@math.aau.dk

Eksempel

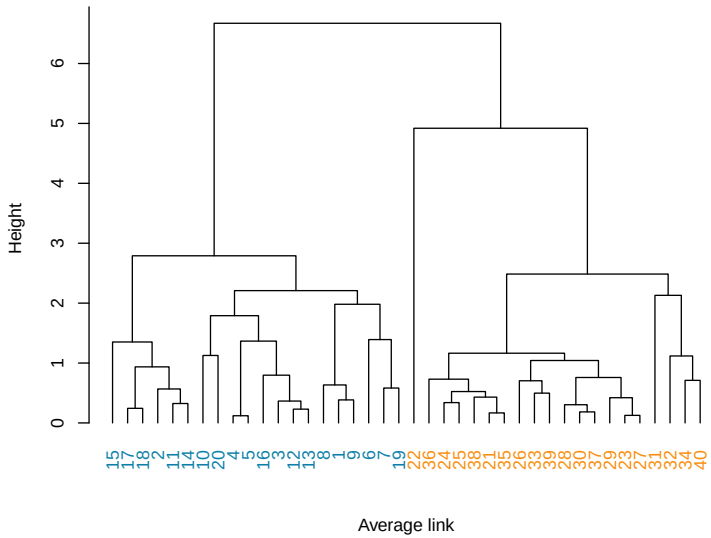


Torben Tvedebrink
tvede@math.aau.dk

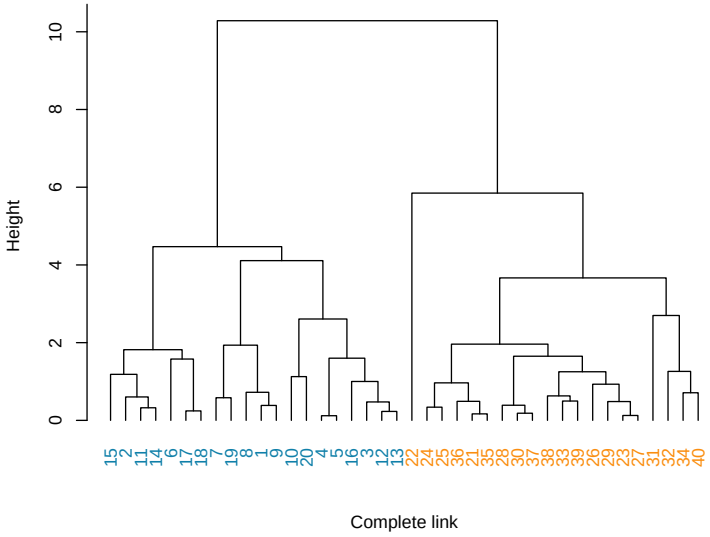
Eksempel



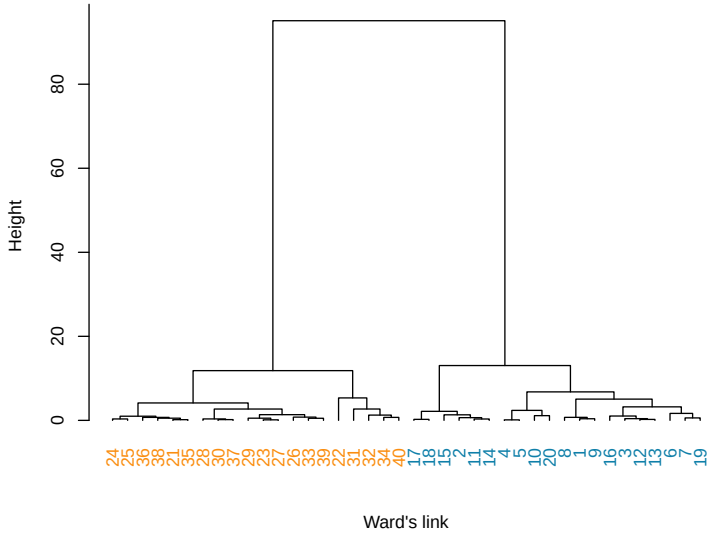
Single link



Average link



Complete link



Ward's link

Torben Tvedebrink
tvede@math.aau.dk

Plot dendogrammet

Efter vi har udført hierarkisk clustering og gemt objektet som et R objekt, fx.:

```
hc <- hclust(dist(USArrests), "ave")
plot(hc)
plot(hc, hang = -1)
```

vi plotter altså ved at bruge `plot`-kommandoen. I R virker mange funktioner på forskellig vis afhængigt af typen af objektet.

Man kan "overskære" træet i en bestemt højde eller antal undertræer vha. `cutree`-kommandoen, ved hhv. `h` eller `k` argumenterne.

Torben Tvedebrink
tvede@math.aau.dk

Opgaver

- Løs opgaver 1–9 fra afsnit 2.7 i Wu og Kumar.
- Antag at data er *række*-standardiseret. Vis at $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2 = 2(1 - r_{ij})$, hvor r_{ij} er korrelationen mellem række i og j.
  Vis ligeledes resultatet numerisk for datasættet `USArrests` (fx. ved et plot).
- Gennemgå analyserne i `MASS4-cluster.pdf` (moodle) – og suppler hvor nødvendigt for at forstå de anvendte metoder.

Torben Tvedebrink
tvede@math.aau.dk