ETC3250/5250 Introduction to Machine Learning

Week 9: K-means and hierarchical clustering

Professor Di Cook

etc3250.clayton-x@monash.edu

Department of Econometrics and Business Statistics

Overview

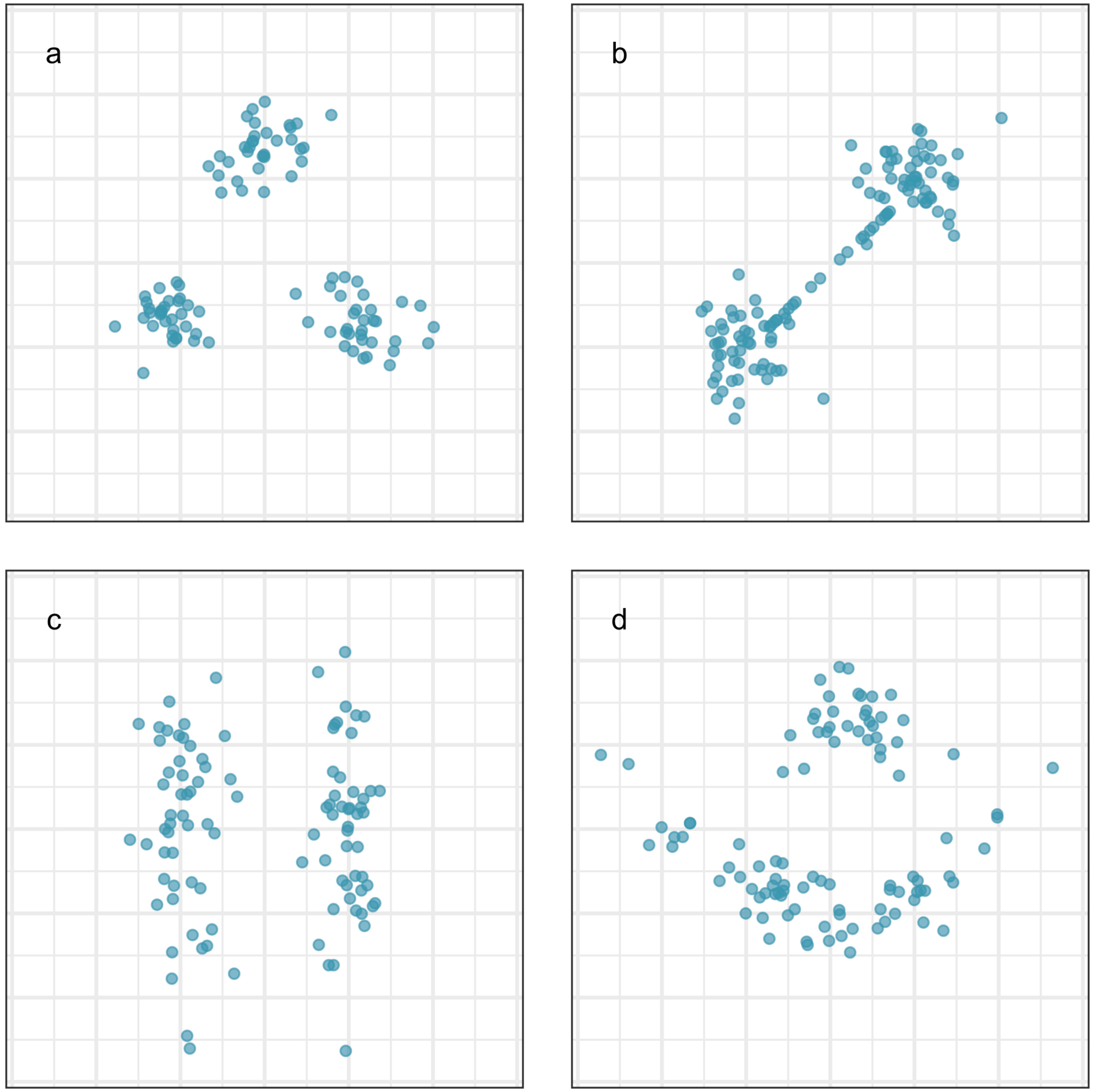
We will cover:

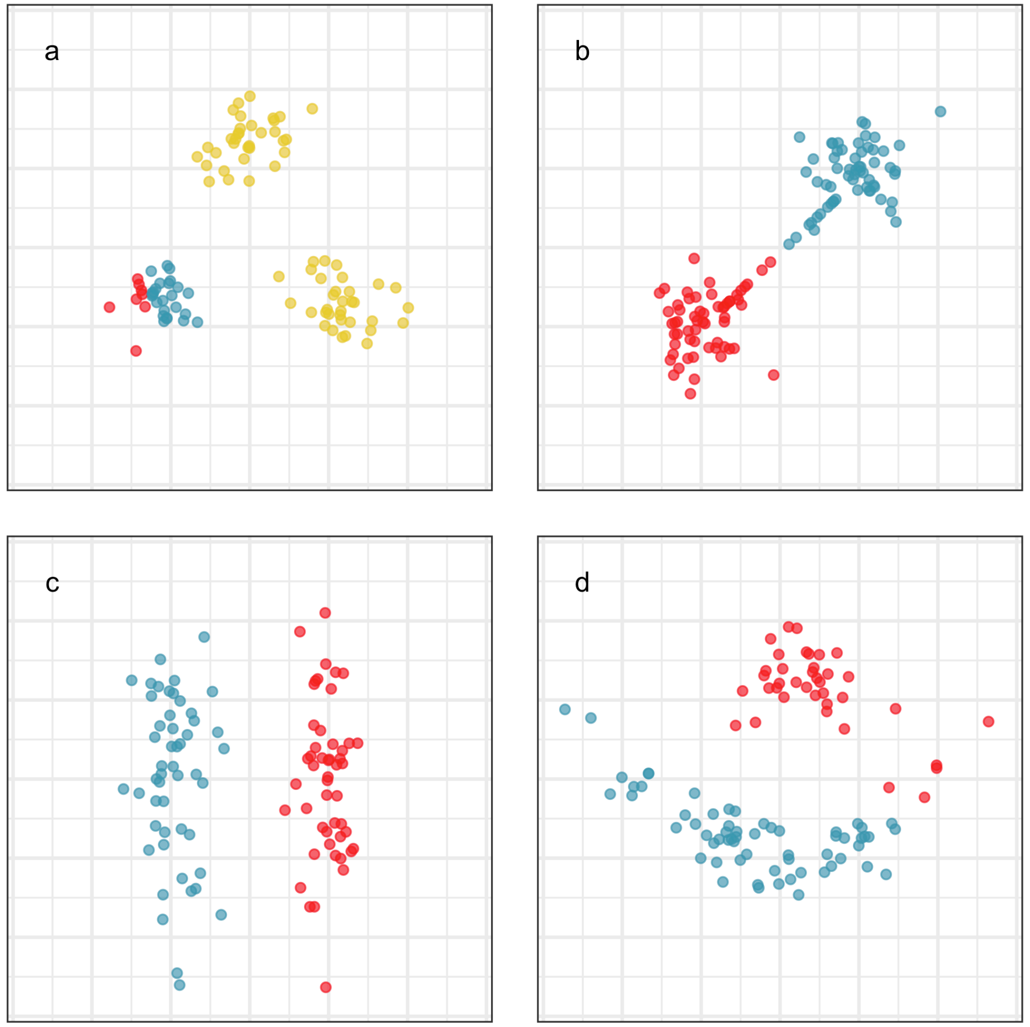
- Defining distance measure
- k-means algorithm
- Hierarchical algorithms
- Making and using dendrograms

Cluster analysis

- The aim of cluster analysis is to group cases (objects) according to their similarity on the variables. It is also often called unsupervised classification, meaning that classification is the ultimate goal, but the classes (groups) are not known ahead of time.
- Hence the first task in cluster analysis is to construct the class information. To determine closeness we start with measuring the interpoint distances.

Cluster these!





It's *easy* if we can see the clusters, but what an algorithm sees might be quite different.

Seeing the clusters using spin-and-brush

```r
library(detourr)
set.seed(645)
detour(p_std[,1:4],
       tour_aes(projection = bl:bm)) |>
       tour_path(grand_tour(2), fps = 60,
                 max_bases=40) |>
       show_scatter(alpha = 0.7,
                    axes = FALSE)
```

How do you measure “close”?

Common interpoint distance measures

Let $A = (x_{a1}, x_{a2}, \dots, x_{ap})$ and $B = (x_{b1}, x_{b2}, \dots, x_{bp})$.

Euclidean

$$d_{EUC}(A,B) = \sqrt{\sum_{j=1}^{p}(x_{aj} - x_{bj})^2}$$

$$= \sqrt{((X_A - X_B)^\top (X_A - X_B))}$$

Other distance metrics

- Mahalanobis (or statistical) distance: $\sqrt{((X_A - X_B)^\top S^{-1} (X_A - X_B))}$
- Manhattan: $\sum_{j=1}^{p} |(X_{aj} - X_{bj})|$
- Minkowski: $(\sum_{j=1}^{p} |(X_{aj} - X_{bj})|^m)^{1/m}$

Count data

- Canberra: $\frac{1}{n_z} \sum_{j=1}^{p} \frac{X_{aj}-X_{bj}}{X_{aj}+X_{bj}}$
- Bray-Curtis: $\frac{\sum_{j=1}^{p} |X_{aj}-X_{bj}|}{\sum_{j=1}^{p} (X_{aj}+X_{bj})}$

Categorical variables

- 1- simple matching coefficient: $1 - (\#matches)/p$
- Convert to dummy variables, and use Euclidean distance

Mixed variable types

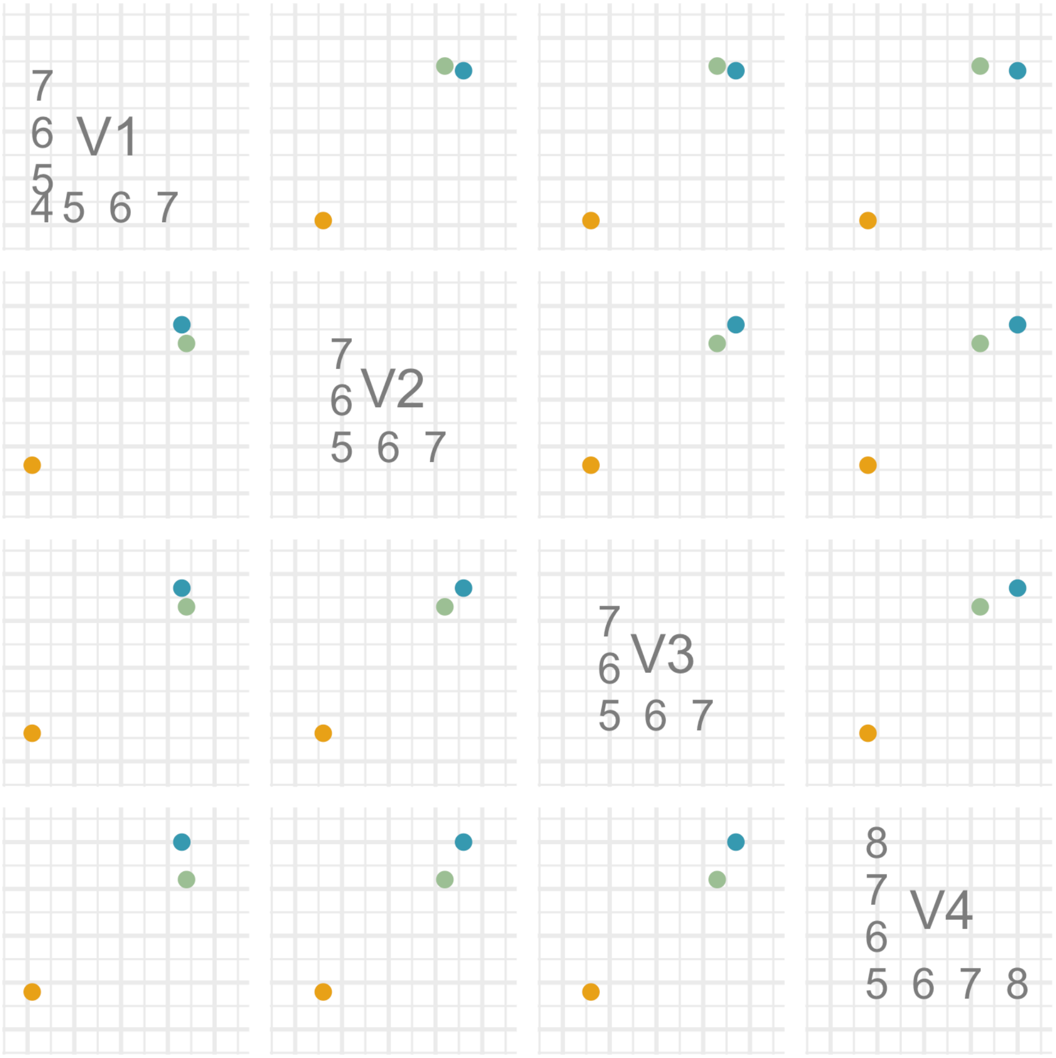
- Gower's distance

Other

- Hamming: all binary variables, number of variables at which values are different.
- Cosine: $\frac{\sum_{j=1}^{p} X_{aj}X_{bj}}{||X_a||||X_b||}$ (How does this compare to a calculation of correlation??)

Distance calculations

```
   V1  V2  V3  V4 point
1 7.3 7.6 7.7 8.0    a1
2 7.4 7.2 7.3 7.2    a2
3 4.1 4.6 4.6 4.8    a3
```



Compute Euclidean distance between a_1 and a_2.

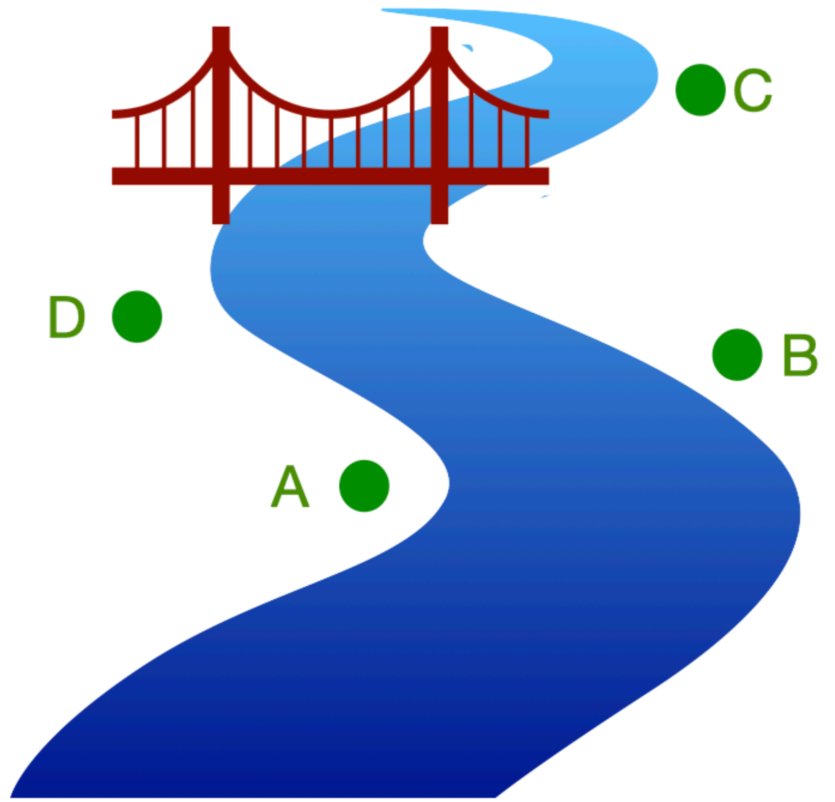
🔑 Standardise your variables!!!!

Could you compute a correlation distance? $d_cor = 1 - |r|$ Is a_1 close to a_3 than a_2?

Basic rules of a distance metric

Anything can be a distance if it follows these rules:

1. $\mathrm{d}(\mathrm{A}, \mathrm{B}) \geq 0$
2. $\mathrm{d}(\mathrm{A}, \mathrm{A}) = 0$
3. $\mathrm{d}(\mathrm{A}, \mathrm{B}) = \mathrm{d}(\mathrm{B}, \mathrm{A})$
4. Metric dissimilarity satisfies
   $\mathrm{d}(\mathrm{A}, \mathrm{B}) \leq \mathrm{d}(\mathrm{A}, \mathrm{C}) + \mathrm{d}(\mathrm{C}, \mathrm{B})$



- If both points on left bank, or both on right bank, use Euclidean distance.
- If on opposite sides, Euclidean distances to bridge, plus length of bridge crossing.
- Does this satisfy the rules?

Dissimilarity vs similarity

- Distance is a **dissimilarity** measure because small means close and large means far.
- Correlation is a **similarity** measure because the larger the value the closer the objects. It can be converted to a dissimilarity with a transformation.

k-means clustering

k-means clustering - algorithm (1/8)

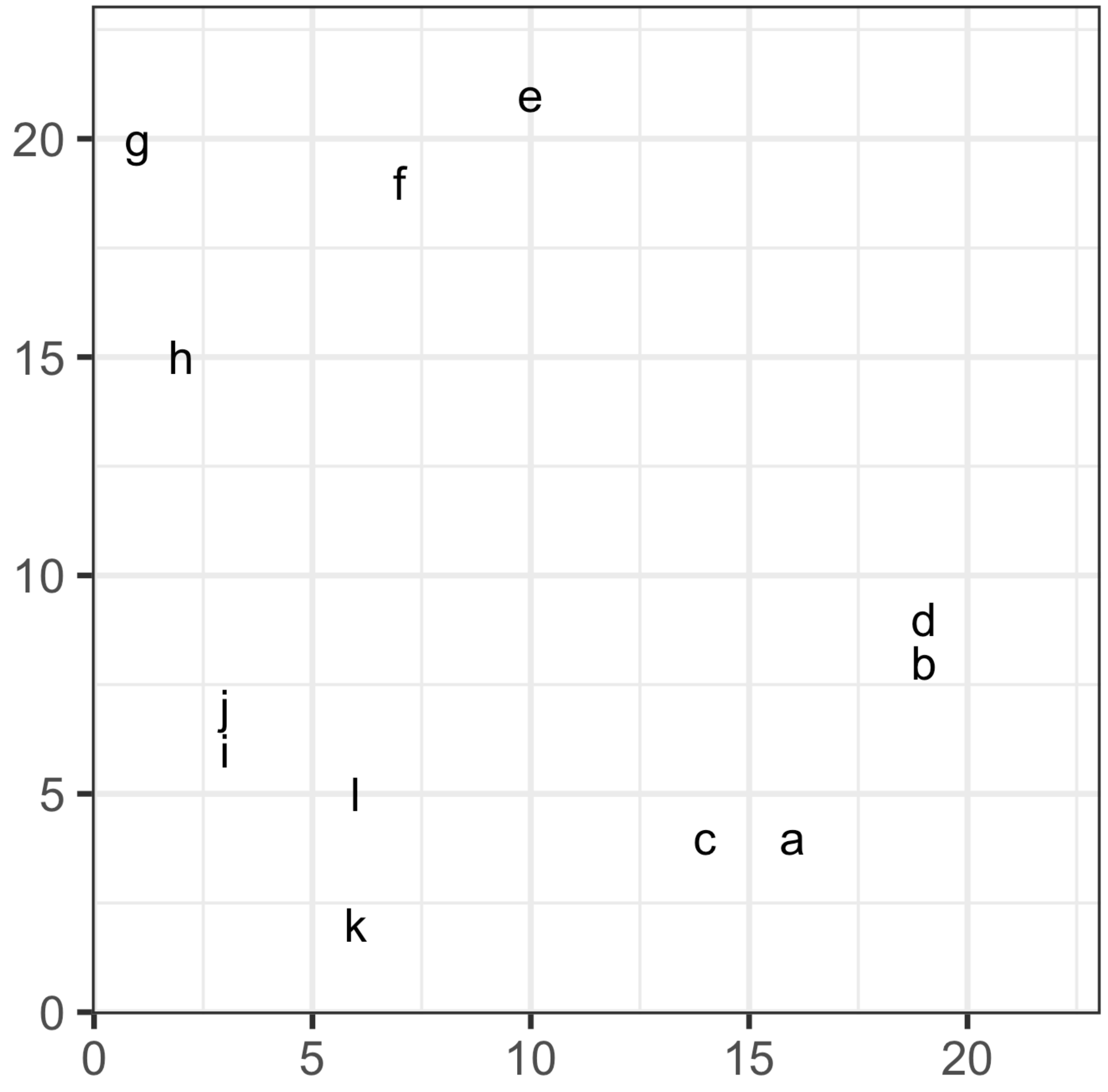
This is an iterative procedure. To use it the number of clusters, k, must be decided first. The stages of the iteration are:

1. Initialize by either (a) partitioning the data into k groups, and compute the k group means or (b) an initial set of k points as the first estimate of the cluster means (seed points).
2. Loop over all observations reassigning them to the group with the closest mean.
3. Recompute group means.
4. Iterate steps 2 and 3 until convergence.

Thean C. Lim's blog post

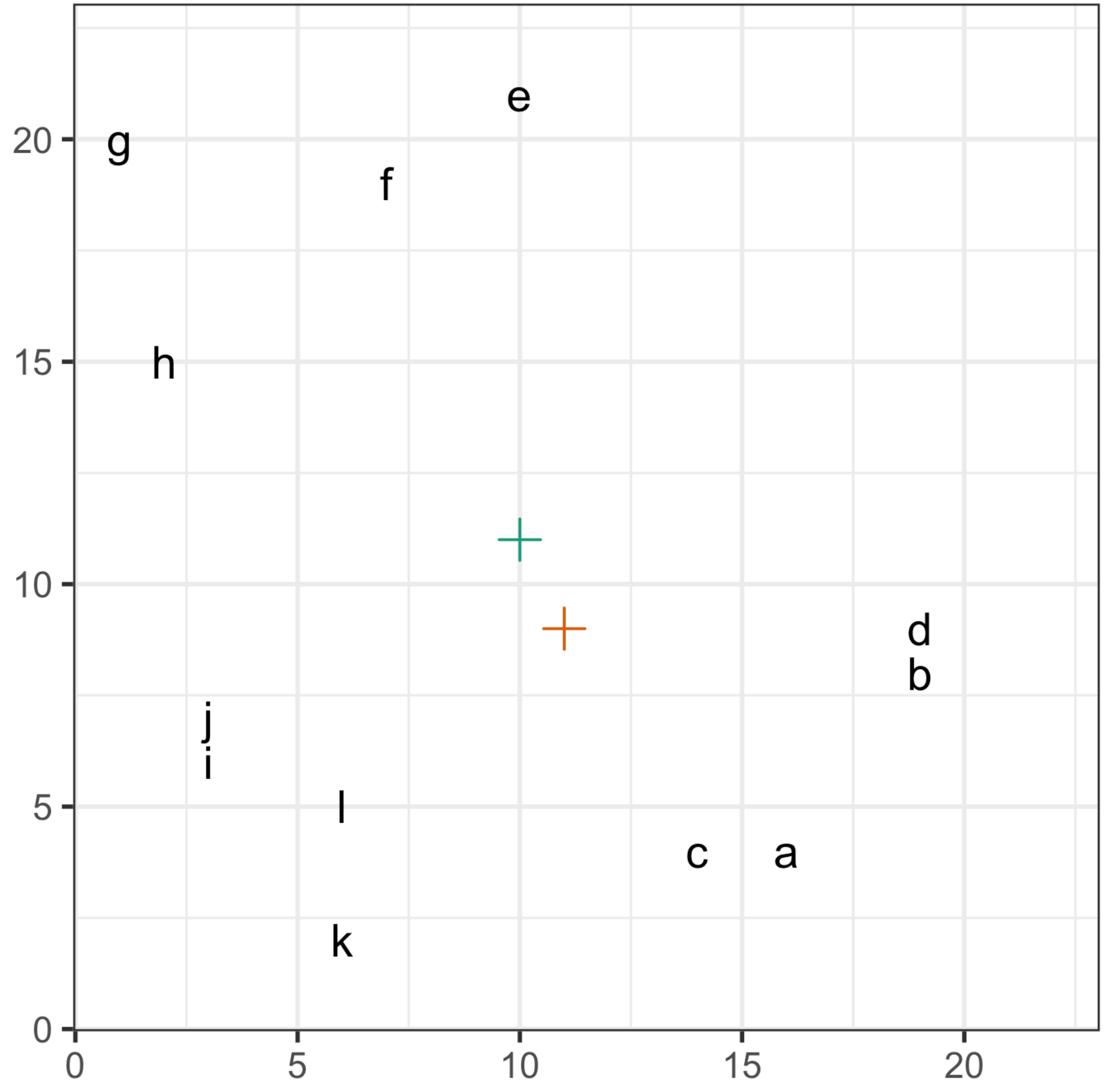
k-means clustering - algorithm (2/8)

lbl	x1	x2
a	16	4
b	19	8
c	14	4
d	19	9
e	10	21
f	7	19
g	1	20
h	2	15
i	3	6
j	3	7
k	6	2
l	6	5



k-means clustering - algorithm (3/8)

Select $k = 2$, and set initial seed means $\bar{x}_1 = (10, 11)$, $\bar{x}_2 = (11, 9)$



k-means clustering - algorithm (4/8)

Compute distances (d_1, d_2) between each observation and each mean,

$\bar{x}_1 = (10, 11)$, $\bar{x}_2 = (11, 9)$

lbl	x1	x2	d1	d2
a	16	4	9.2	7.1
b	19	8	9.5	8.1
c	14	4	8.1	5.8
d	19	9	9.2	8.0
e	10	21	10.0	12.0
f	7	19	8.5	10.8
g	1	20	12.7	14.9
h	2	15	8.9	10.8
i	3	6	8.6	8.5
j	3	7	8.1	8.2
k	6	2	9.8	8.6
l	6	5	7.2	6.4

k-means clustering - algorithm (5/8)

Assign each observation to a cluster, based on which mean is closest.

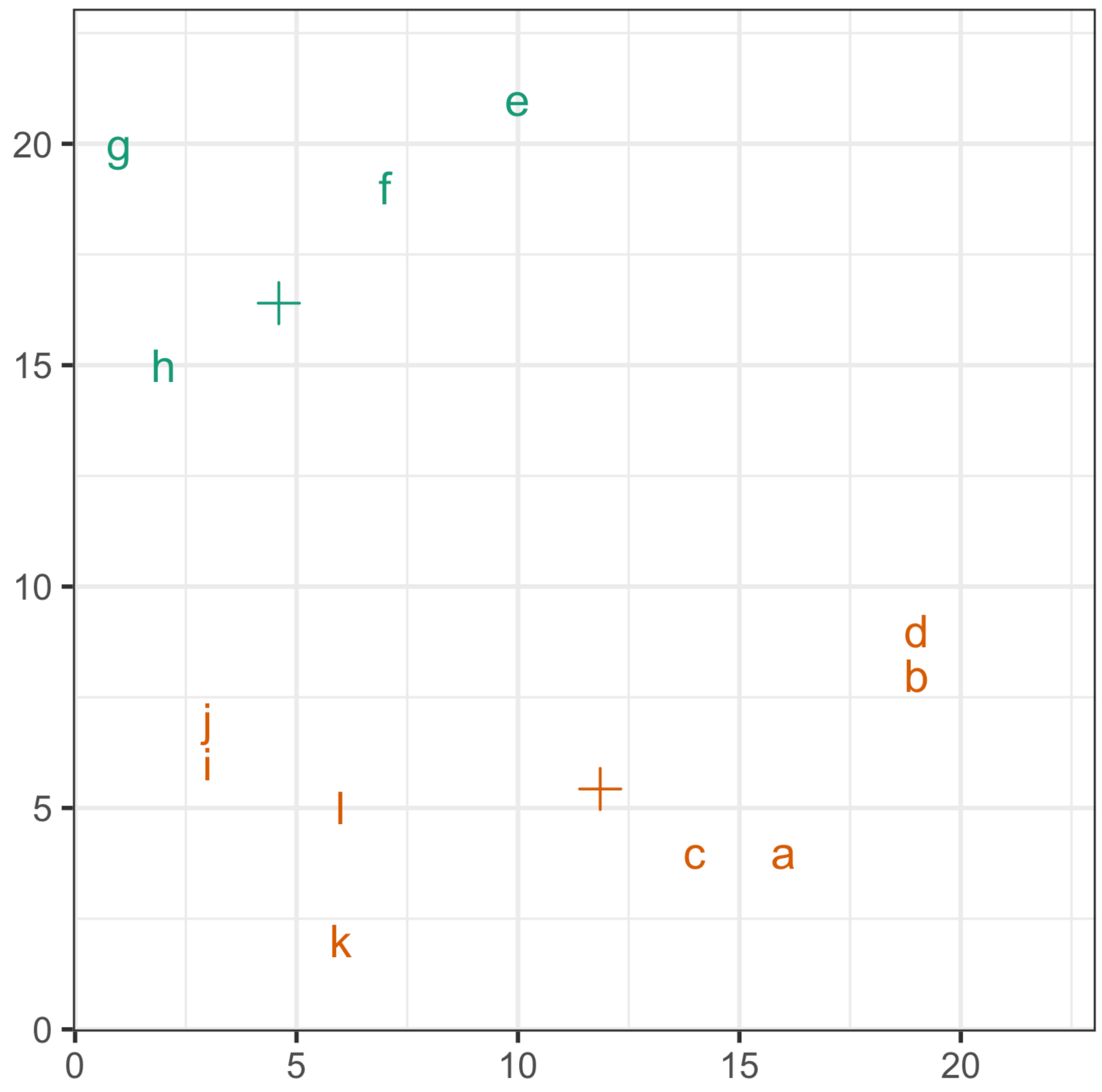
lbl	x1	x2	d1	d2	cl
a	16	4	9.2	7.1	2
b	19	8	9.5	8.1	2
c	14	4	8.1	5.8	2
d	19	9	9.2	8.0	2
e	10	21	10.0	12.0	1
f	7	19	8.5	10.8	1
g	1	20	12.7	14.9	1
h	2	15	8.9	10.8	1
i	3	6	8.6	8.5	2
j	3	7	8.1	8.2	1
k	6	2	9.8	8.6	2
l	6	5	7.2	6.4	2

k-means clustering - algorithm (6/8)

Recompute means, and re-assign the cluster membership

$\bar{x}_1 = (5, 16)$, $\bar{x}_2 = (12, 5)$

lbl	x1	x2	d1	d2	cl
a	16	4	16.8	4.4	2
b	19	8	16.7	7.6	2
c	14	4	15.6	2.6	2
d	19	9	16.2	8.0	2
e	10	21	7.1	15.7	1
f	7	19	3.5	14.4	1
g	1	20	5.1	18.2	1
h	2	15	3.0	13.7	1
i	3	6	10.5	8.9	2
j	3	7	9.5	9.0	2
k	6	2	14.5	6.8	2
l	6	5	11.5	5.9	2

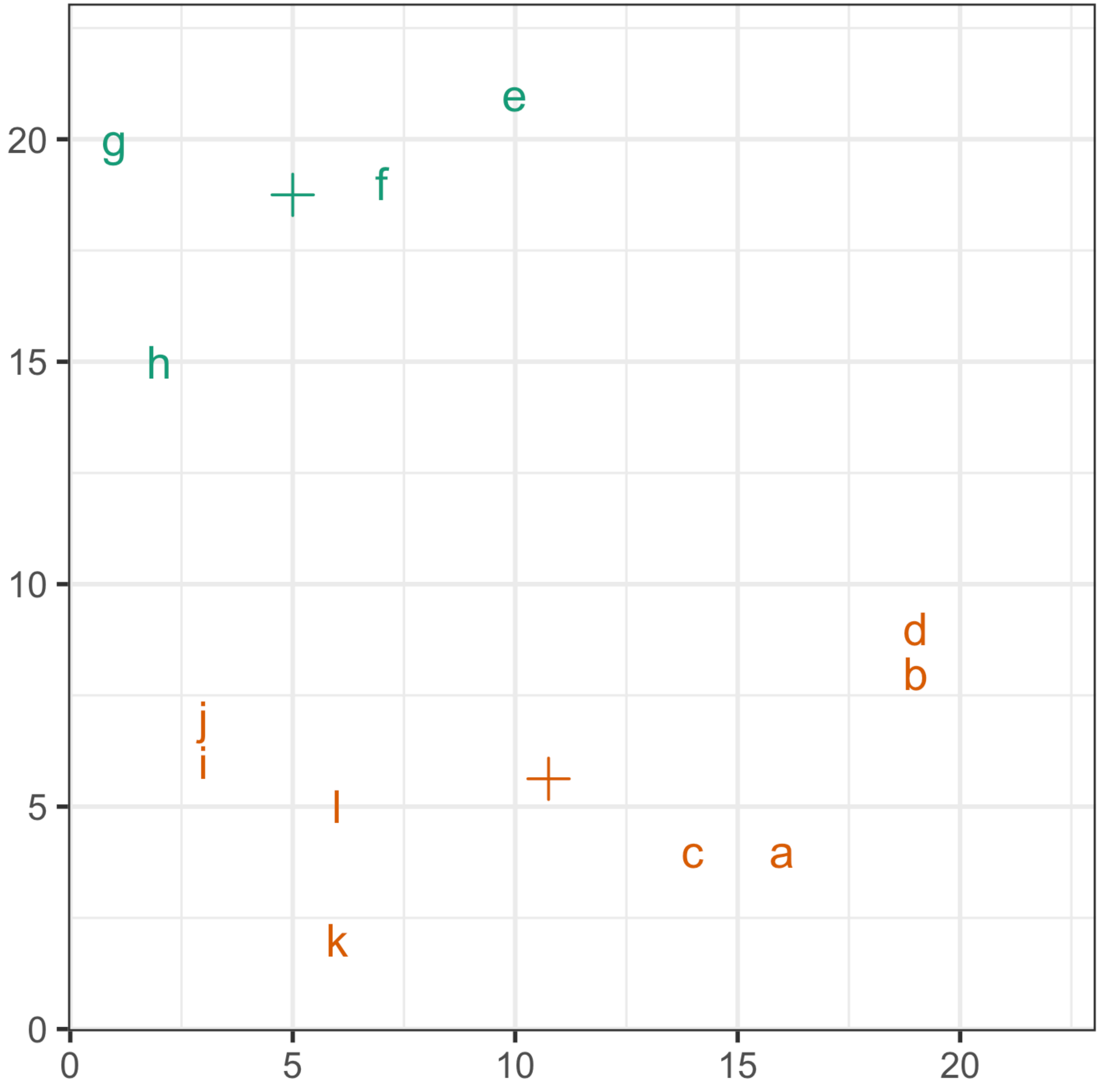


k-means clustering - algorithm (7/8)

Recompute means, and re-assign the cluster membership

$\bar{x}_1 = (5, 19)$, $\bar{x}_2 = (11, 6)$

lbl	x1	x2	d1	d2	cl
a	16	4	18.4	5.5	2
b	19	8	17.7	8.6	2
c	14	4	17.3	3.6	2
d	19	9	17.1	8.9	2
e	10	21	5.5	15.4	1
f	7	19	2.0	13.9	1
g	1	20	4.2	17.4	1
h	2	15	4.8	12.8	1
i	3	6	12.9	7.8	2
j	3	7	11.9	7.9	2
k	6	2	16.8	6.0	2
l	6	5	13.8	4.8	2

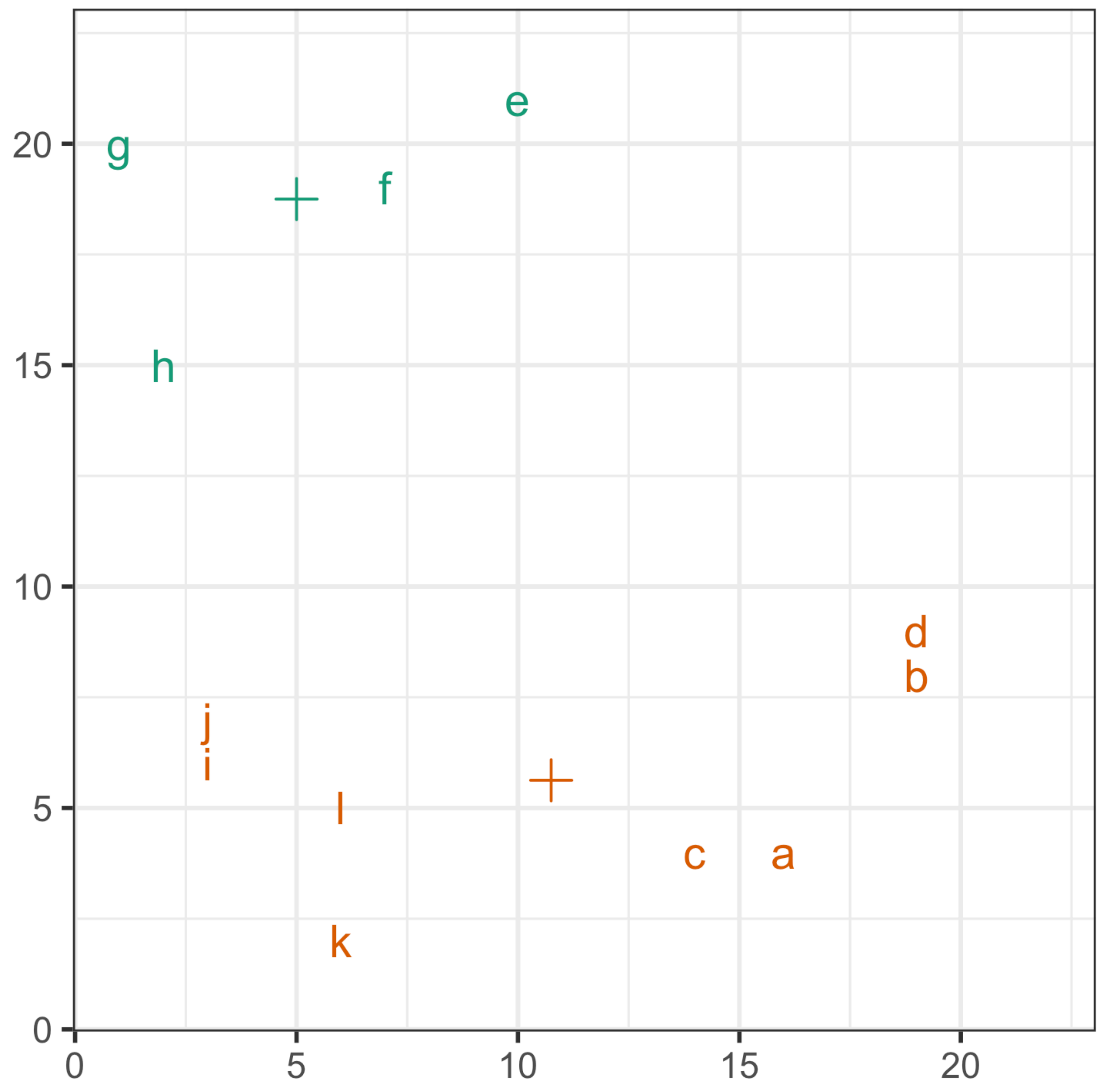


k-means clustering - algorithm (8/8)

Recompute means, and re-assign the cluster membership

$\bar{x}_1 = (5, 19)$, $\bar{x}_2 = (11, 6)$

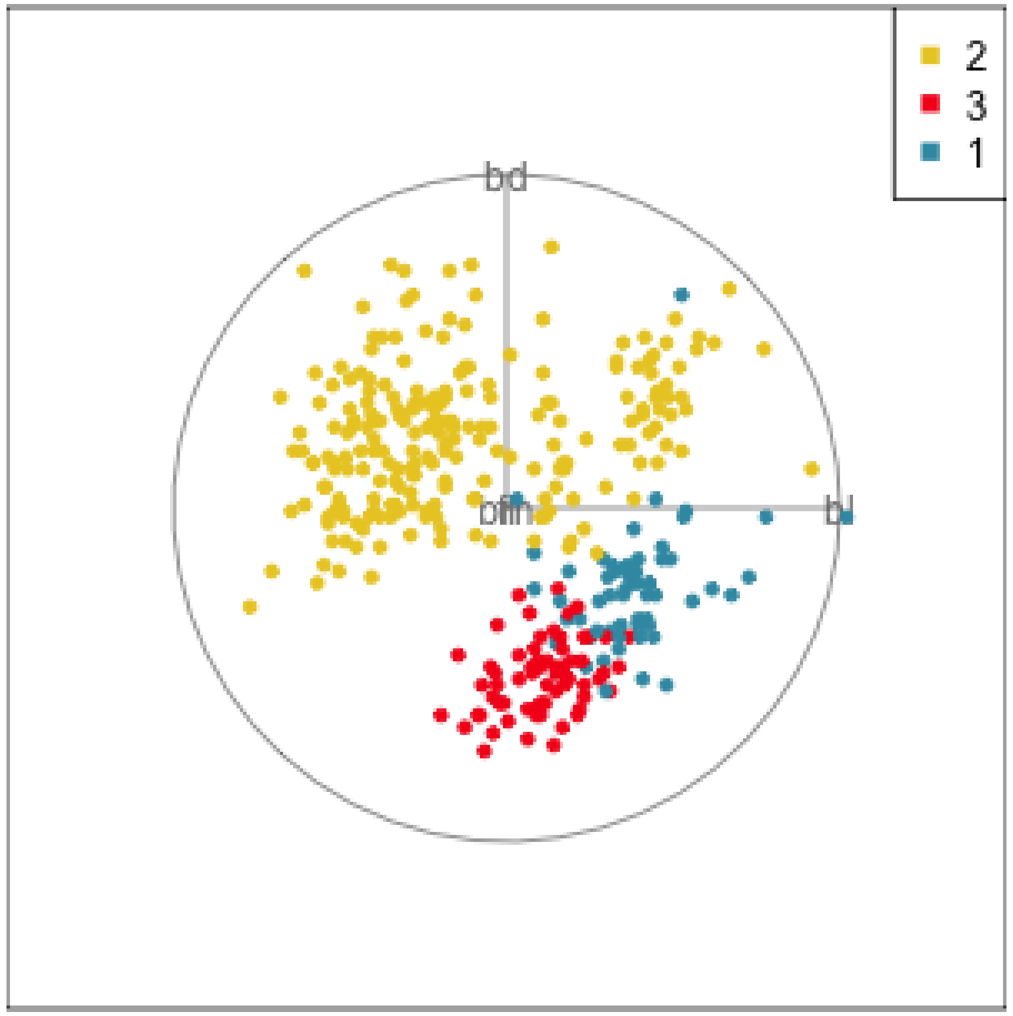
lbl	x1	x2	d1	d2	cl
a	16	4	18.4	5.5	2
b	19	8	17.7	8.6	2
c	14	4	17.3	3.6	2
d	19	9	17.1	8.9	2
e	10	21	5.5	15.4	1
f	7	19	2.0	13.9	1
g	1	20	4.2	17.4	1
h	2	15	4.8	12.8	1
i	3	6	12.9	7.8	2
j	3	7	11.9	7.9	2
k	6	2	16.8	6.0	2
l	6	5	13.8	4.8	2



Example: penguins

- We know there are three clusters, but generally we don't know this.
- Will $k = 3$-means clustering see three?
- Fit for various values of k. Add cluster label to data.
- Examine solution in plots of the data.
- Compute cluster metrics.
- NOTE: `set.seed()` because results can depend on initialisation.

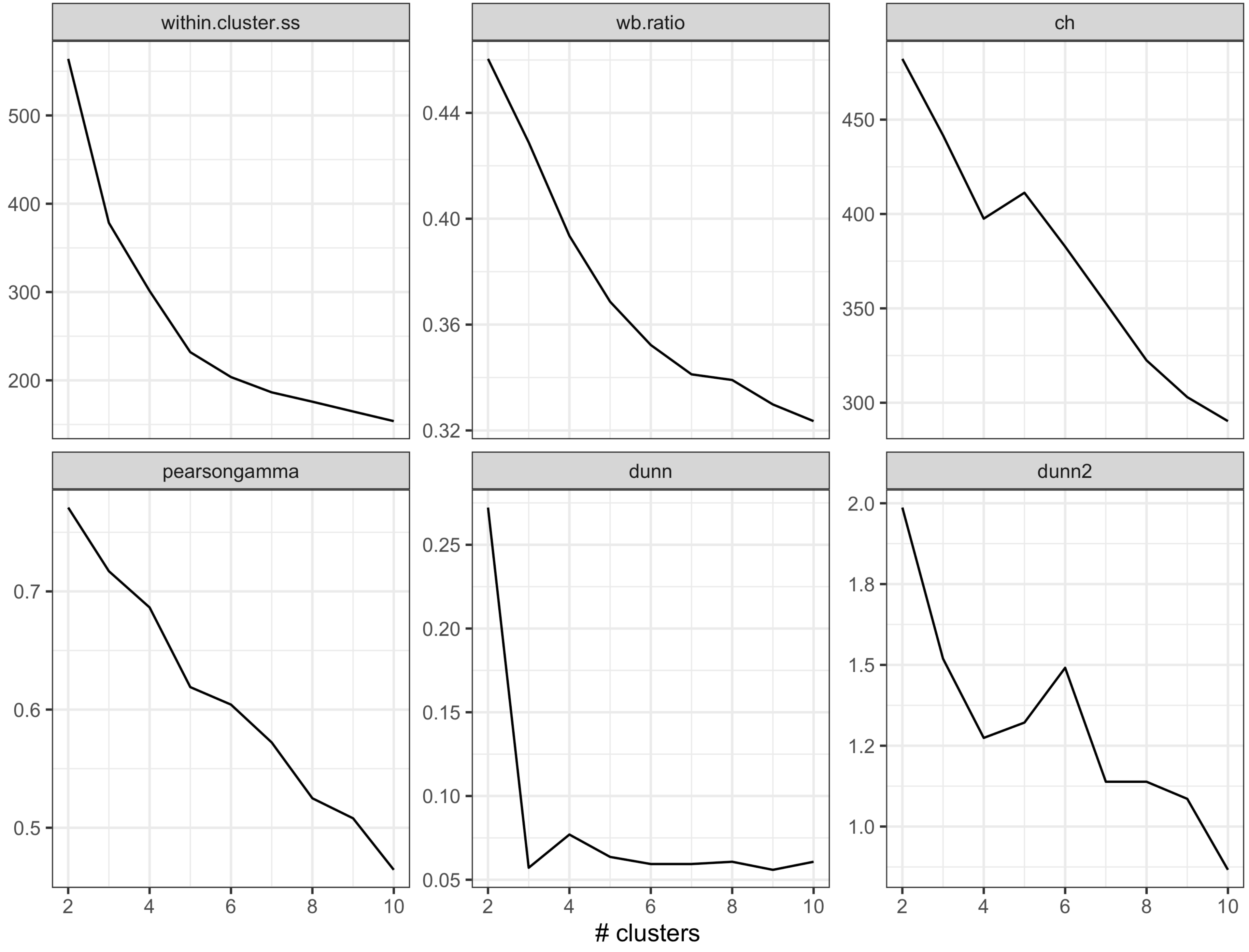
```
set.seed(712)
p_km3 <- kmeans(p_std[,2:5], 3)
p_std_km <- p_std |>
  mutate(cl = factor(p_km3$cluster))
```



Choosing k with cluster statistics (1/2)

- within.cluster.ss: sum of distances within cluster. Want it to be low, but always drops for each additional cluster so look for large drops.
- WBRatio: average within/average between distances. Want it to be low, but always drops for each additional cluster so look for large drops.
- Hubert Gamma: (s+ - s-)/(s+ + s-) where $s+$ =sum of number of within $<$ between, $s- =$ sum of number within $>$ between. Want this to be high.
- Dunn: ratio of (smallest distance between points from different clusters) to (maximum distance of points within any cluster). Want this to be high.
- Calinski-Harabasz Index: $\frac{\sum_{i=1}^{p} B_{ii}/(k-1)}{\sum_{i=1}^{p} W_{ii}/(n-k)}$. Want this to be high.

Choosing k with cluster statistics (2/2)



- Results are inconclusive. No agreement between metrics.
- Not unusual. Stay tuned for nuisance variables and observations.

Hierarchical clustering

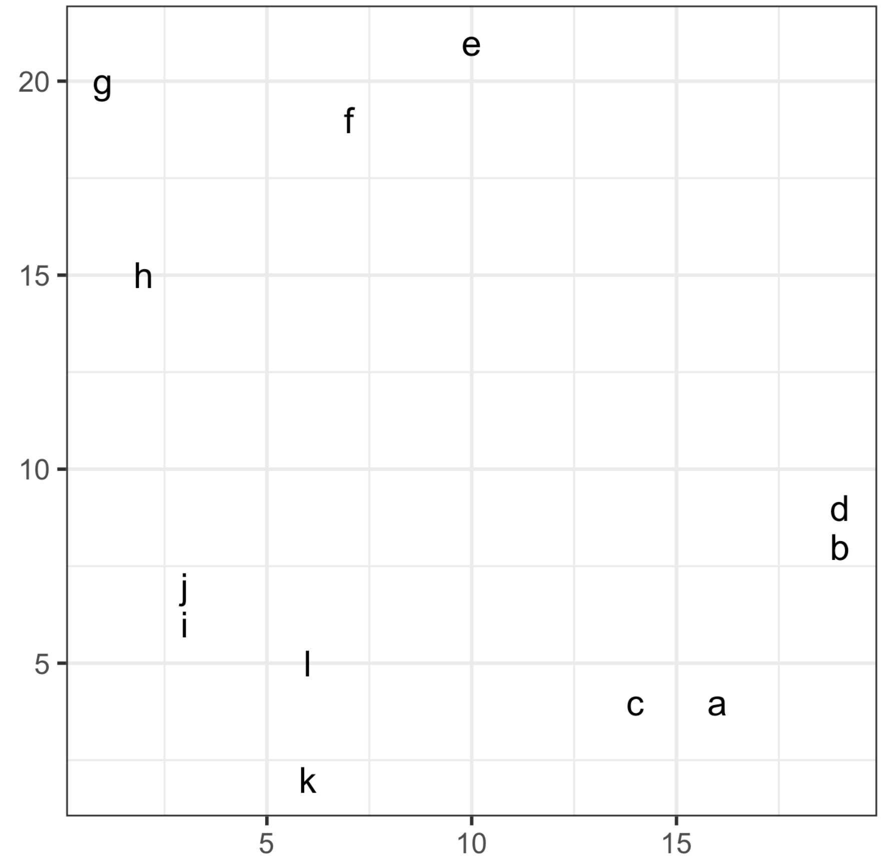
Hierarchical clustering 1/4

- Agglomeration: Begin with all observations in singleton clusters. Sequentially join points into clusters, until all are in one cluster.
- Divisive: Begin with all observtions in one cluster, adn sequentially divide until all observations are in singleton clusters.
- Produces a tree diagram illustrating the process, called a dendrogram.

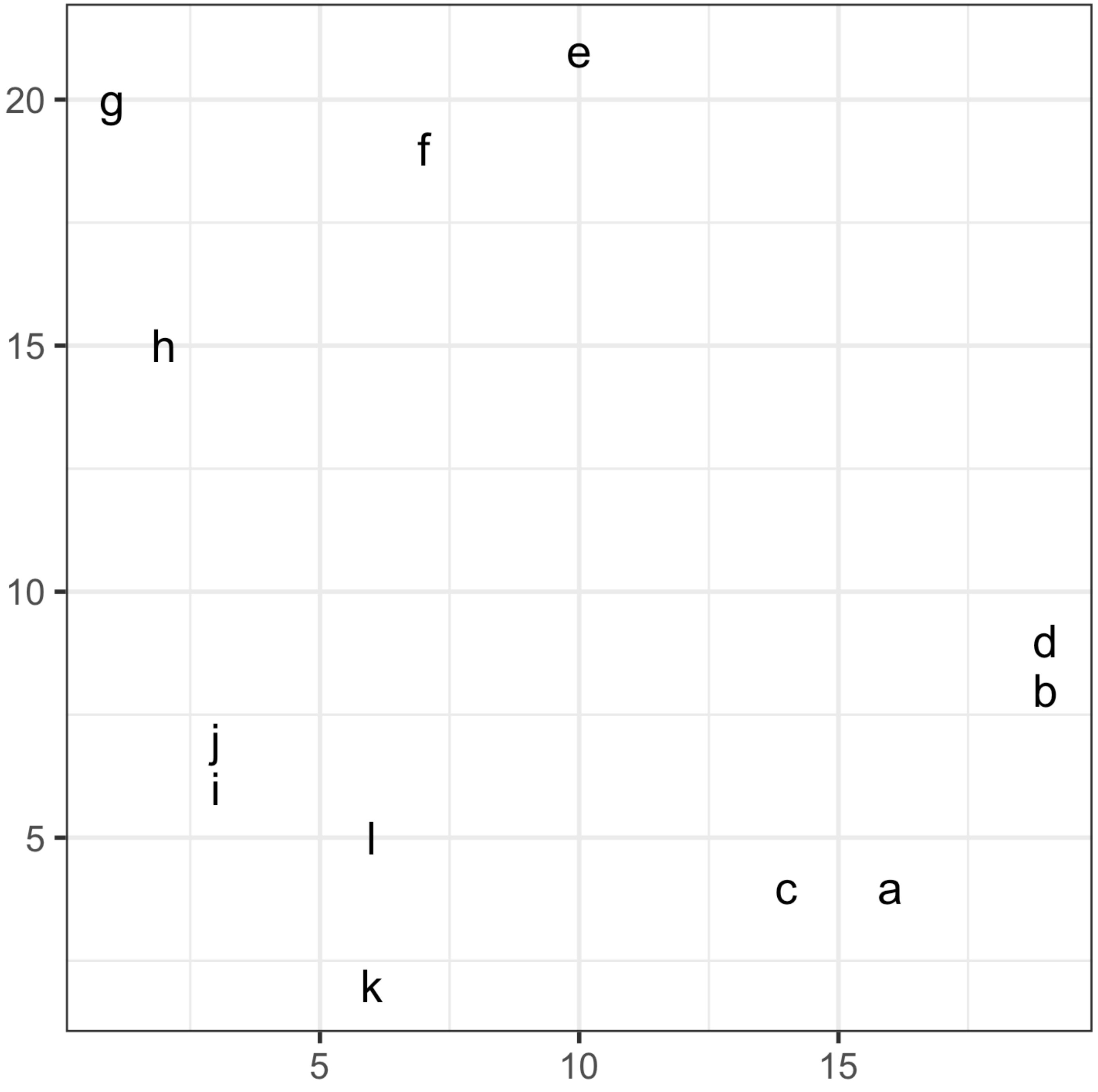
Hierarchical clustering 2/4

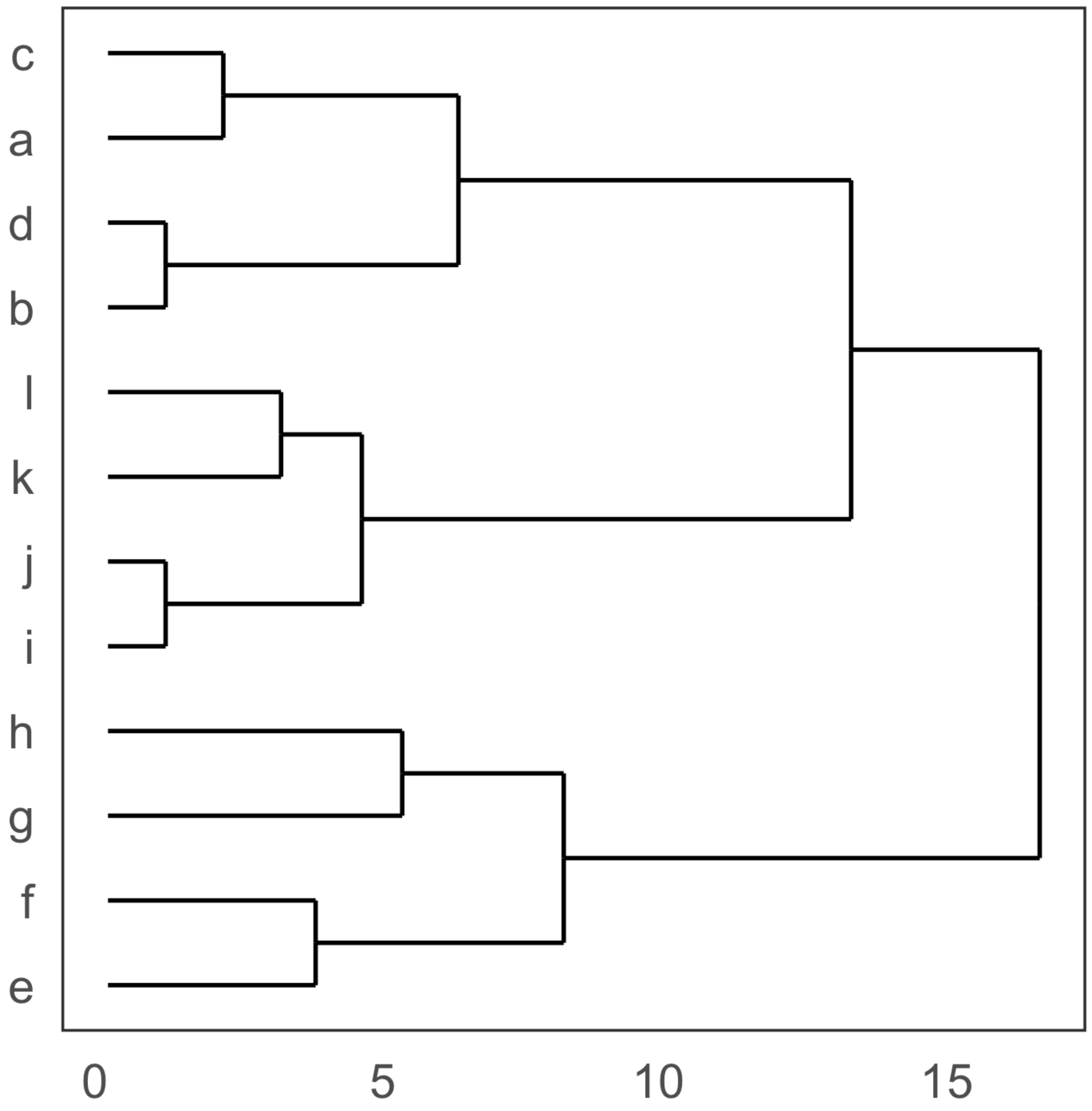
$n \times n$ distance matrix

	a	b	c	d	e	f	g	h	i	j	k	l
a	0.0	5.0	2.0	5.8	18.0	17.5	21.9	17.8	13.2	13.3	10.2	10.0
b	5.0	0.0	6.4	1.0	15.8	16.3	21.6	18.4	16.1	16.0	14.3	13.3
c	2.0	6.4	0.0	7.1	17.5	16.6	20.6	16.3	11.2	11.4	8.2	8.1
d	5.8	1.0	7.1	0.0	15.0	15.6	21.1	18.0	16.3	16.1	14.8	13.6
e	18.0	15.8	17.5	15.0	0.0	3.6	9.1	10.0	16.6	15.7	19.4	16.5
f	17.5	16.3	16.6	15.6	3.6	0.0	6.1	6.4	13.6	12.6	17.0	14.0
g	21.9	21.6	20.6	21.1	9.1	6.1	0.0	5.1	14.1	13.2	18.7	15.8
h	17.8	18.4	16.3	18.0	10.0	6.4	5.1	0.0	9.1	8.1	13.6	10.8
i	13.2	16.1	11.2	16.3	16.6	13.6	14.1	9.1	0.0	1.0	5.0	3.2
j	13.3	16.0	11.4	16.1	15.7	12.6	13.2	8.1	1.0	0.0	5.8	3.6
k	10.2	14.3	8.2	14.8	19.4	17.0	18.7	13.6	5.0	5.8	0.0	3.0
l	10.0	13.3	8.1	13.6	16.5	14.0	15.8	10.8	3.2	3.6	3.0	0.0

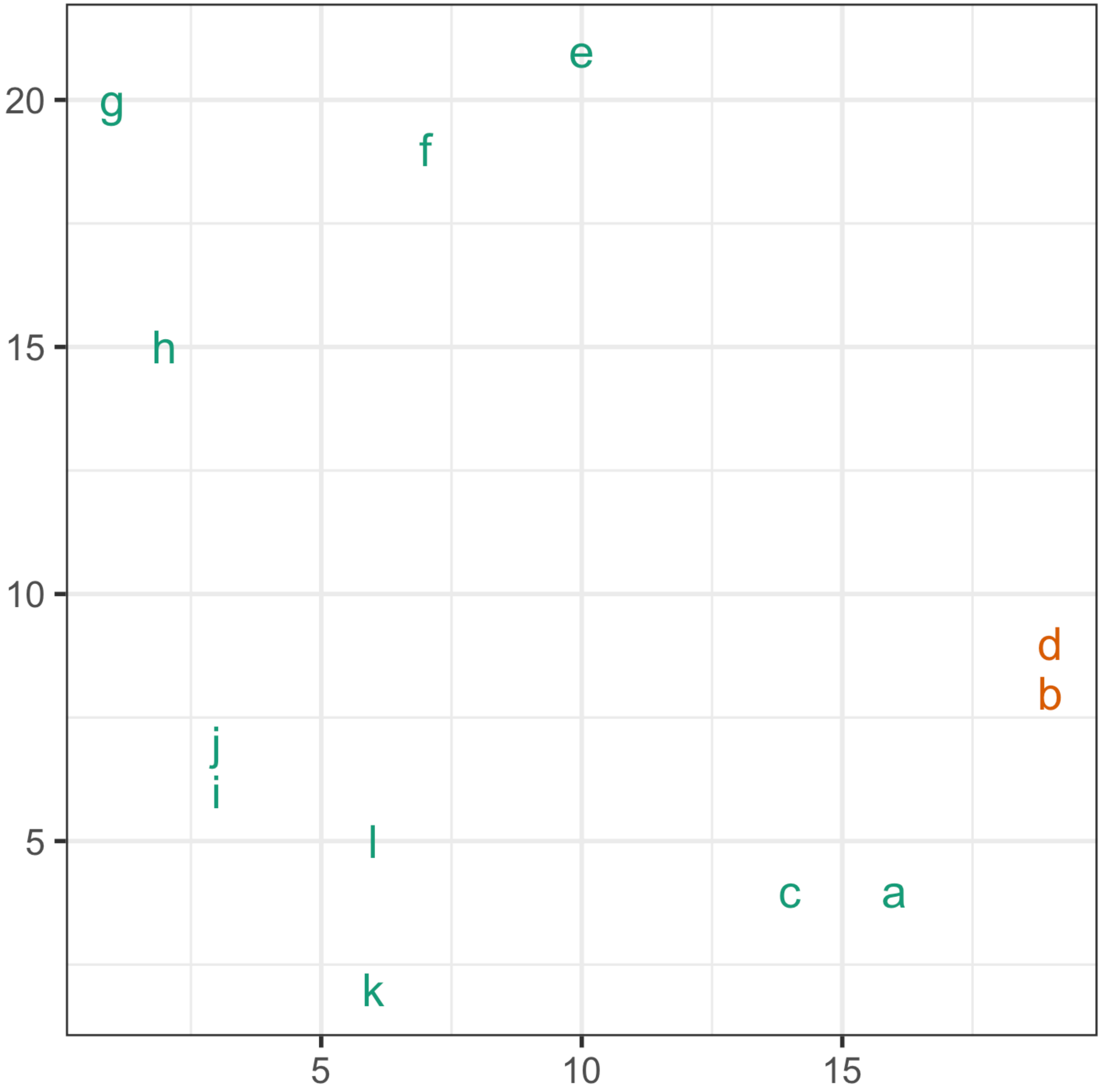


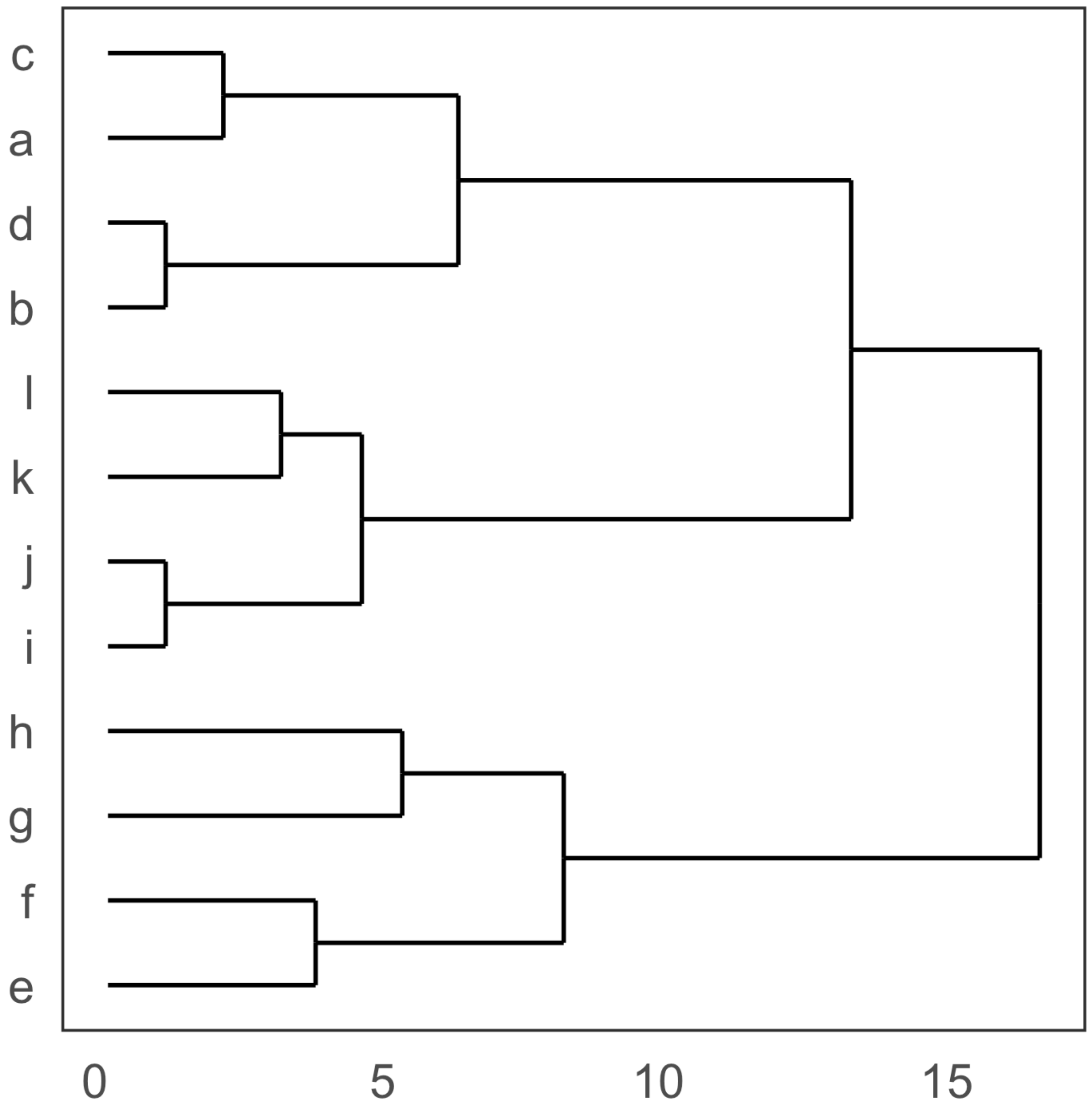
Hierarchical clustering 3/4





Hierarchical clustering 4/4





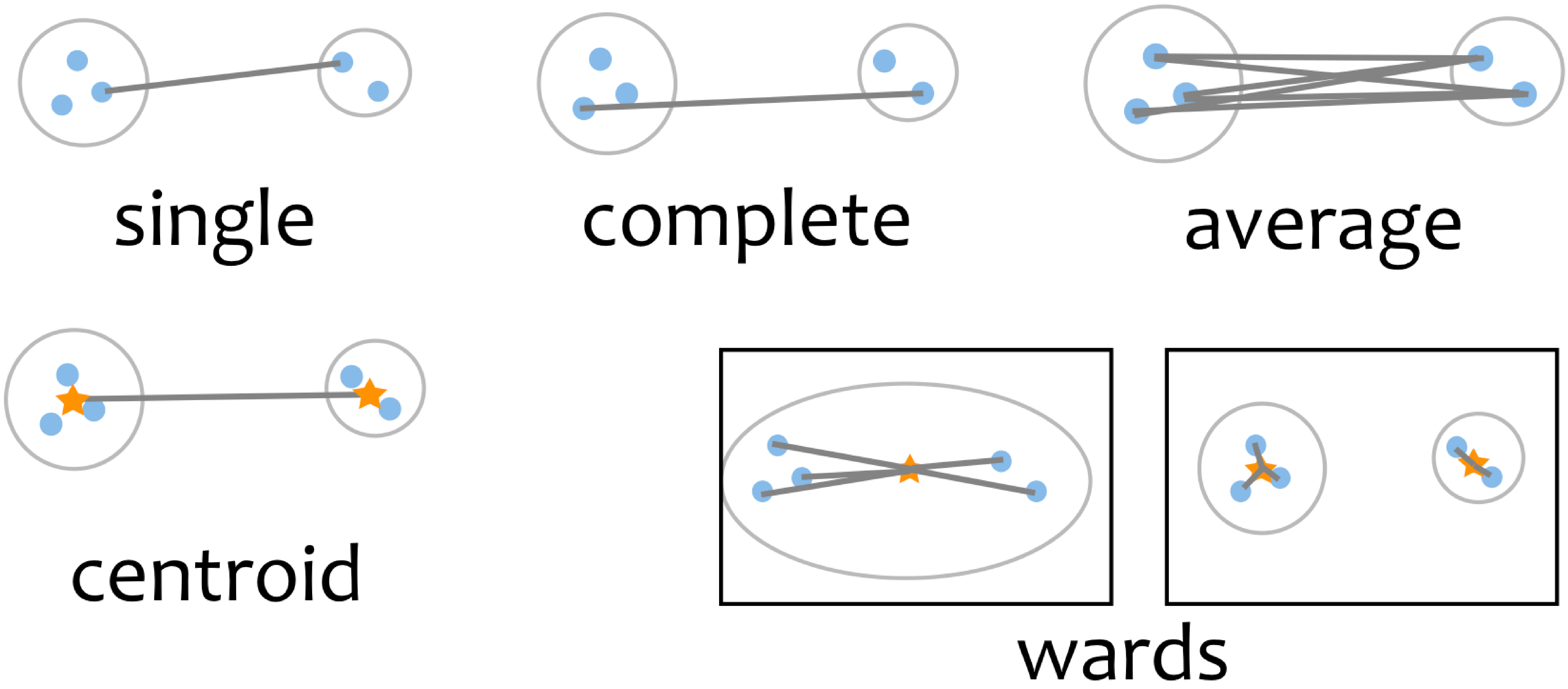
Linkage

What is the distance between the new cluster (d,b) and all of the other observations?

Between points in the cluster to points not in the cluster.

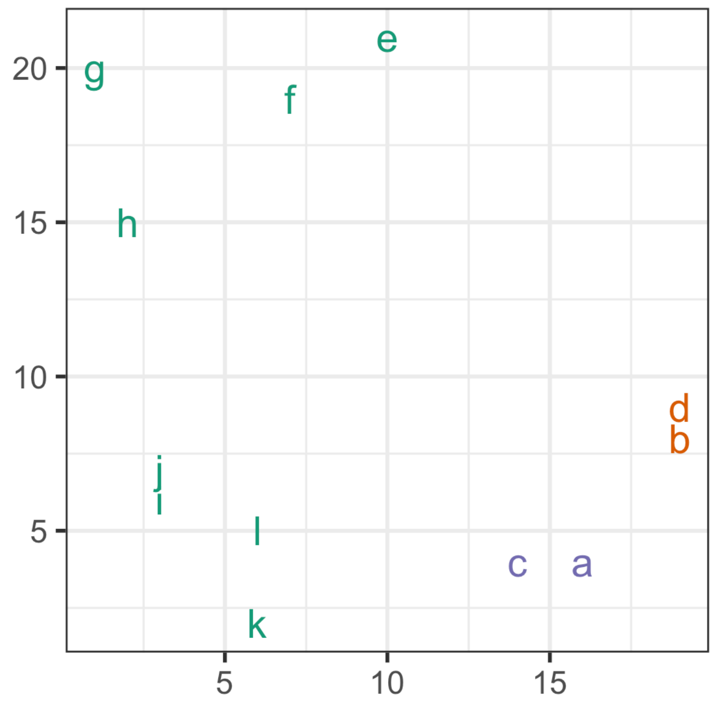
- Single: minimum distance between points in the different clusters
- Complete: maximum distance between points in the different clusters
- Average: average of distances between points in the different clusters
- Centroid: distances between the average of the different clusters
- Wards: minimizes the total within-cluster variance

Linkage



Calculations with different linkage choices

	a	b	c	d	e	f	g	h	i	j	k	l
a	0.0	5.0	2.0	5.8	18.0	17.5	21.9	17.8	13.2	13.3	10.2	10.0
b	5.0	0.0	6.4	1.0	15.8	16.3	21.6	18.4	16.1	16.0	14.3	13.3
c	2.0	6.4	0.0	7.1	17.5	16.6	20.6	16.3	11.2	11.4	8.2	8.1
d	5.8	1.0	7.1	0.0	15.0	15.6	21.1	18.0	16.3	16.1	14.8	13.6
e	18.0	15.8	17.5	15.0	0.0	3.6	9.1	10.0	16.6	15.7	19.4	16.5
f	17.5	16.3	16.6	15.6	3.6	0.0	6.1	6.4	13.6	12.6	17.0	14.0
g	21.9	21.6	20.6	21.1	9.1	6.1	0.0	5.1	14.1	13.2	18.7	15.8
h	17.8	18.4	16.3	18.0	10.0	6.4	5.1	0.0	9.1	8.1	13.6	10.8
i	13.2	16.1	11.2	16.3	16.6	13.6	14.1	9.1	0.0	1.0	5.0	3.2
j	13.3	16.0	11.4	16.1	15.7	12.6	13.2	8.1	1.0	0.0	5.8	3.6
k	10.2	14.3	8.2	14.8	19.4	17.0	18.7	13.6	5.0	5.8	0.0	3.0
l	10.0	13.3	8.1	13.6	16.5	14.0	15.8	10.8	3.2	3.6	3.0	0.0



Distance (b,d):

Distance (a,c):

Linkage between (b,d) and (a,c)

Single:

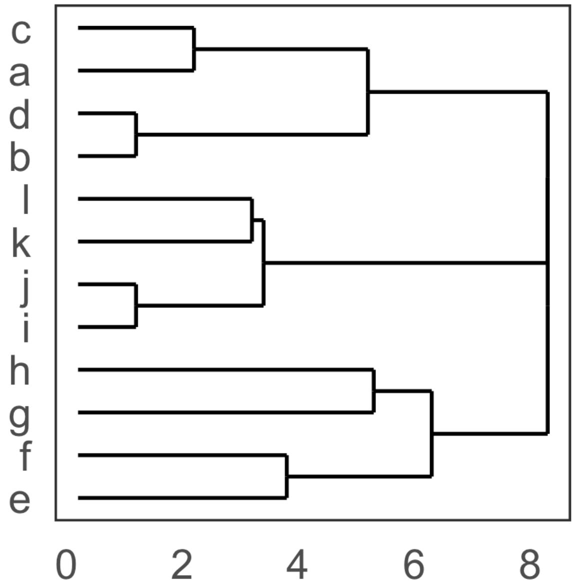
Complete:

Average:

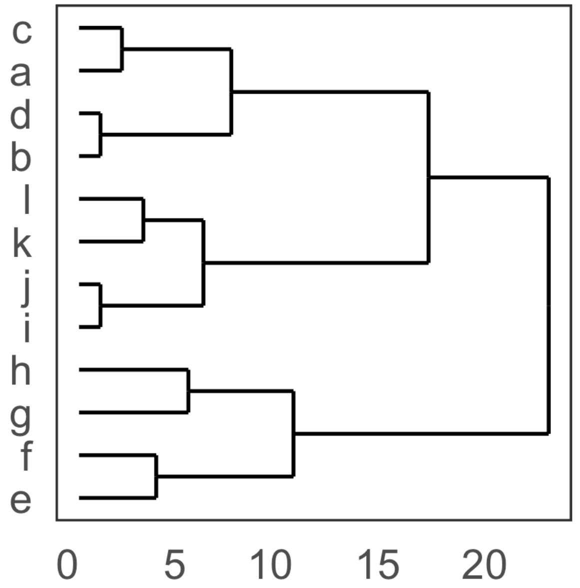
Centroid:

Results from different linkage choices

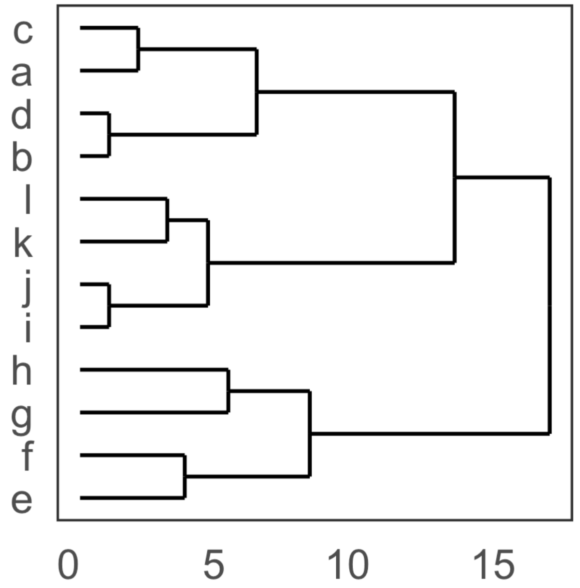
single



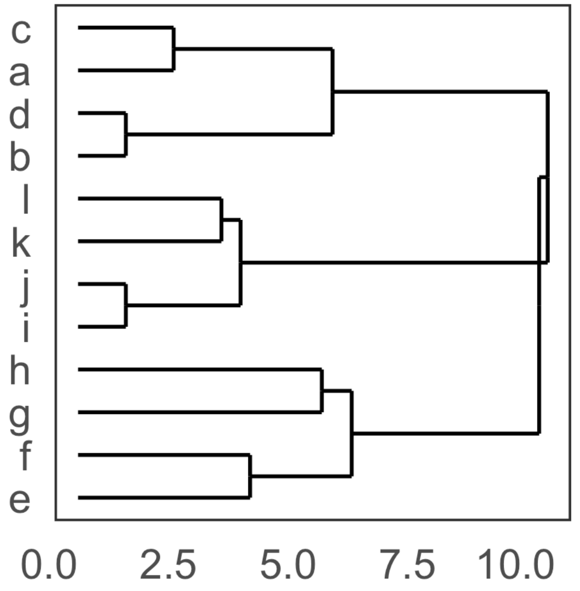
complete



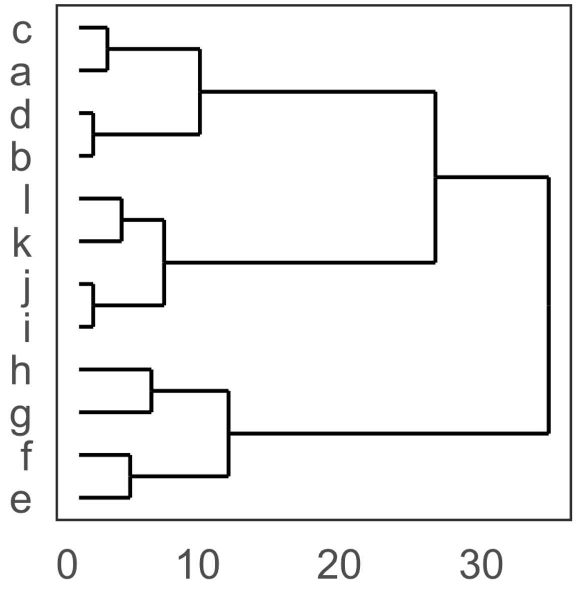
average

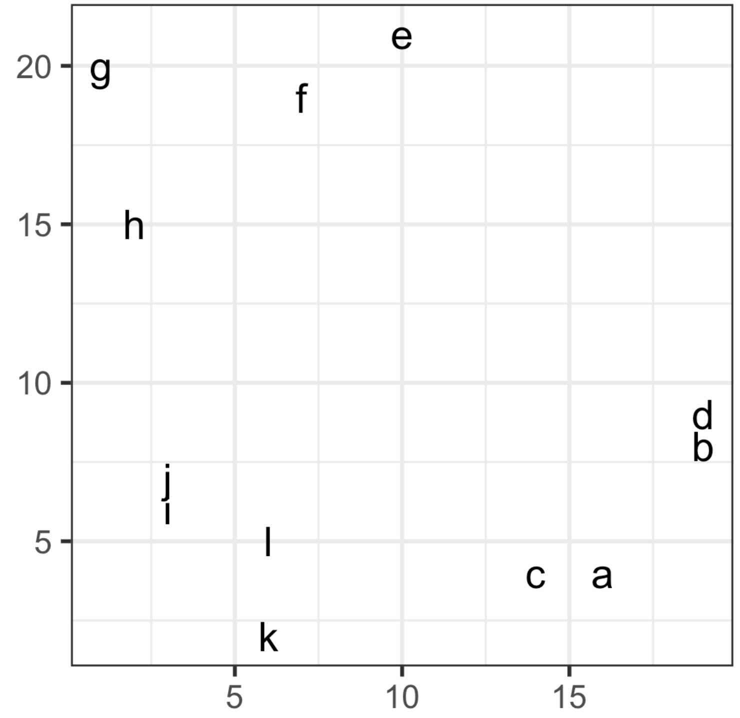


centroid



wards





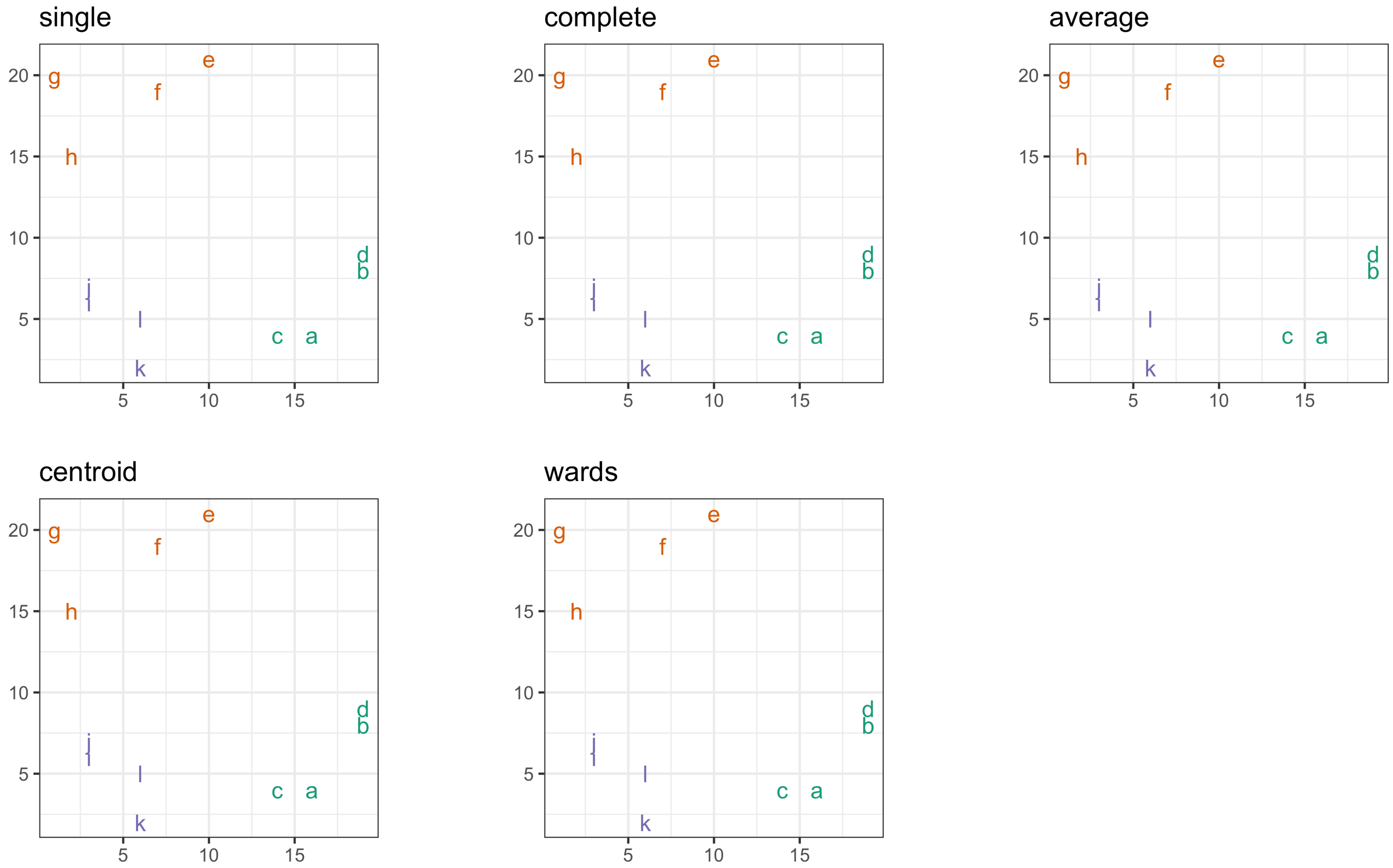
Dendrogram

- Each leaf of the dendrogram represents one observation
- Leaves fuse into branches and branches fuse, either with leaves or other branches.
- Fusions lower in the tree mean the groups of observations are more similar to each other.

Cut the tree to partition the data into k clusters.

Results from different linkage choices

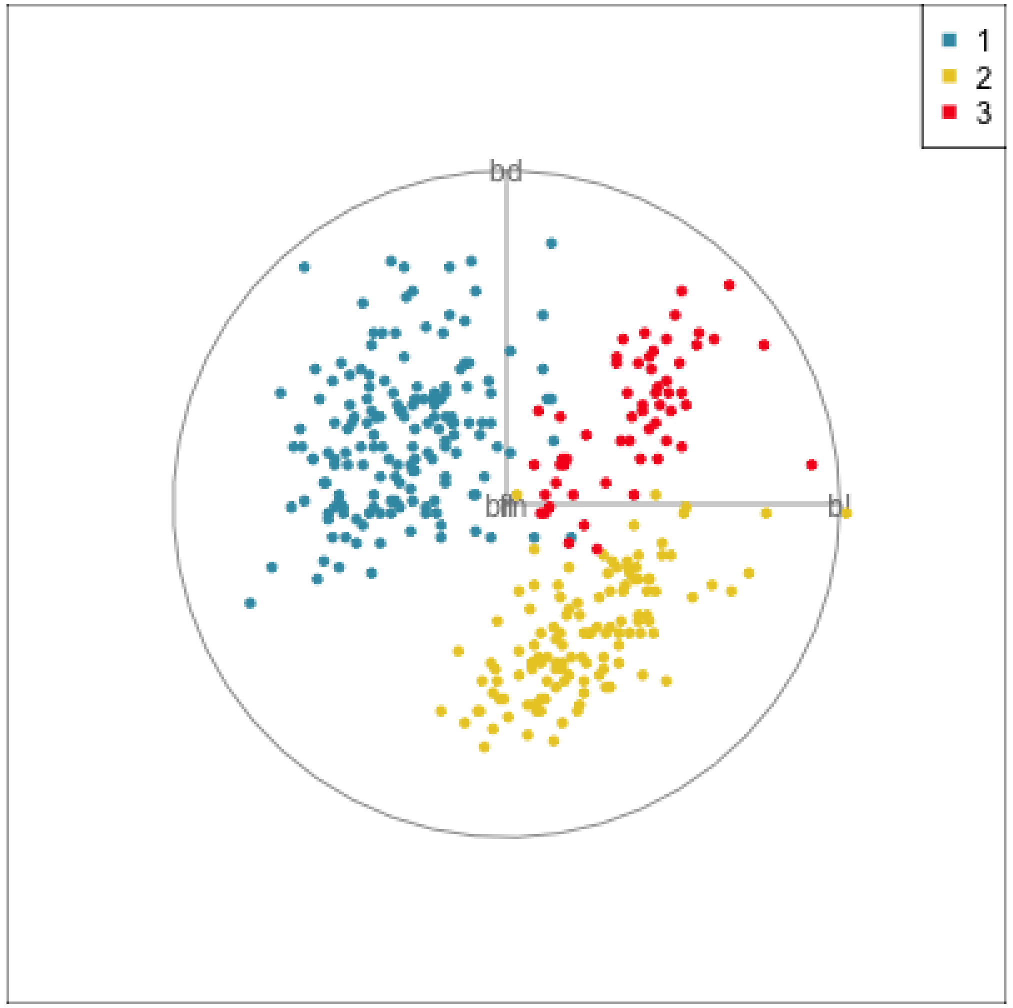
Model-in-the-data-space



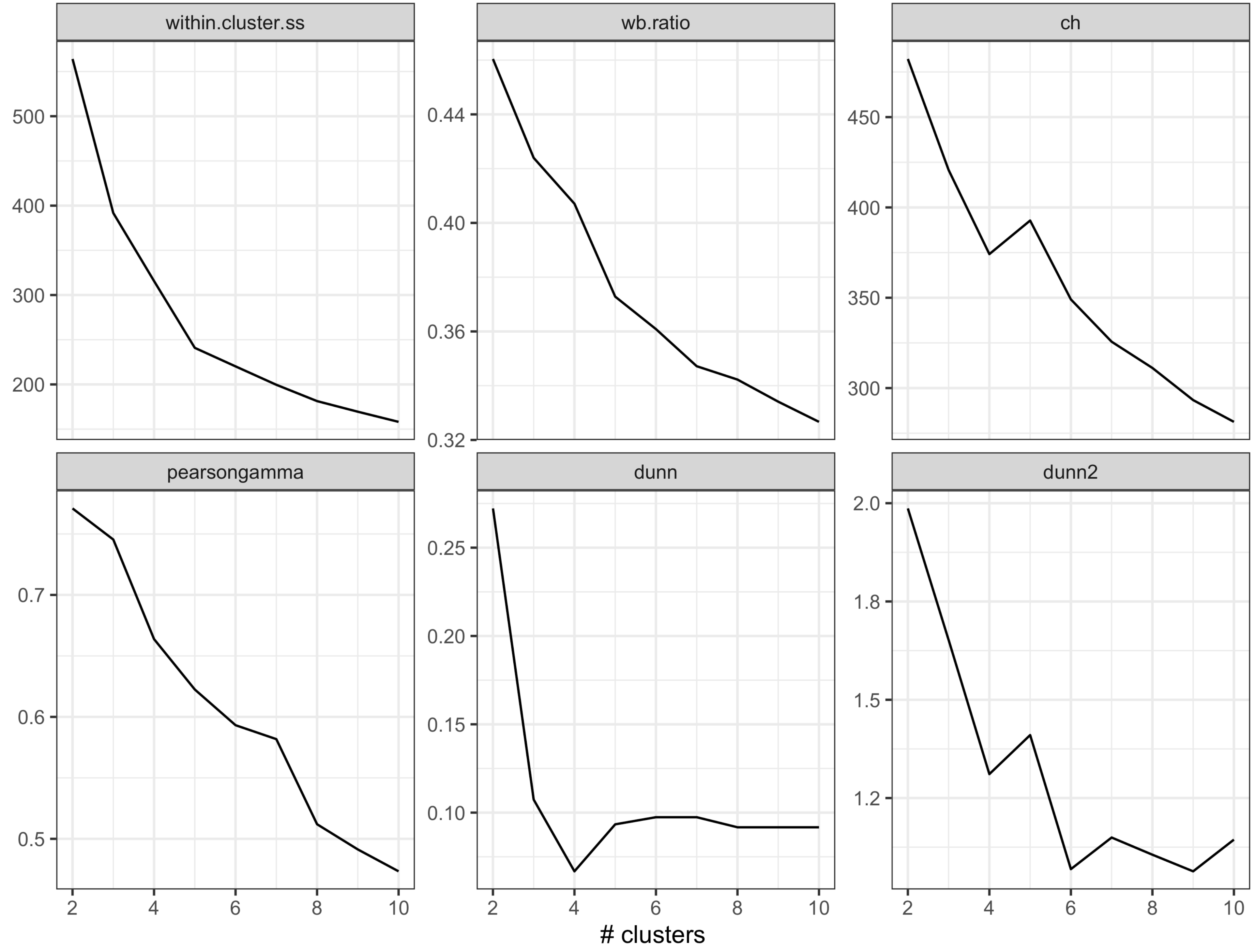
Example: penguins

- We know there are three clusters, but generally we don't know this.
- Will $k = 3$-means clustering see three?
- Fit for various values of k. Add cluster label to data.
- Examine solution in plots of the data.
- Compute cluster metrics.
- NOTE: No need for `set.seed()` because results are deterministic.

```r
p_hc_w3 <- hclust(dist(p_std[,2:5]), method="ward.D2")
p_std_hc_w3 <- p_std |>
  mutate(cl = factor(cutree(p_hc_w3, 3)))
```



Choosing k with cluster statistics



- `within.cluster.ss` and `wb.ratio` suggest 3, and 5
- `pearsongamma` (Hubert) suggests 2-3
- `dunn`, `dunn2`, `ch` all 2?

Next: Model-based clustering and self-organising maps