Data Wrangling and Data Analysis
Clustering:

Daniel Oberski & Erik-Jan van Kesteren

Department of Methodology & Statistics

Utrecht University

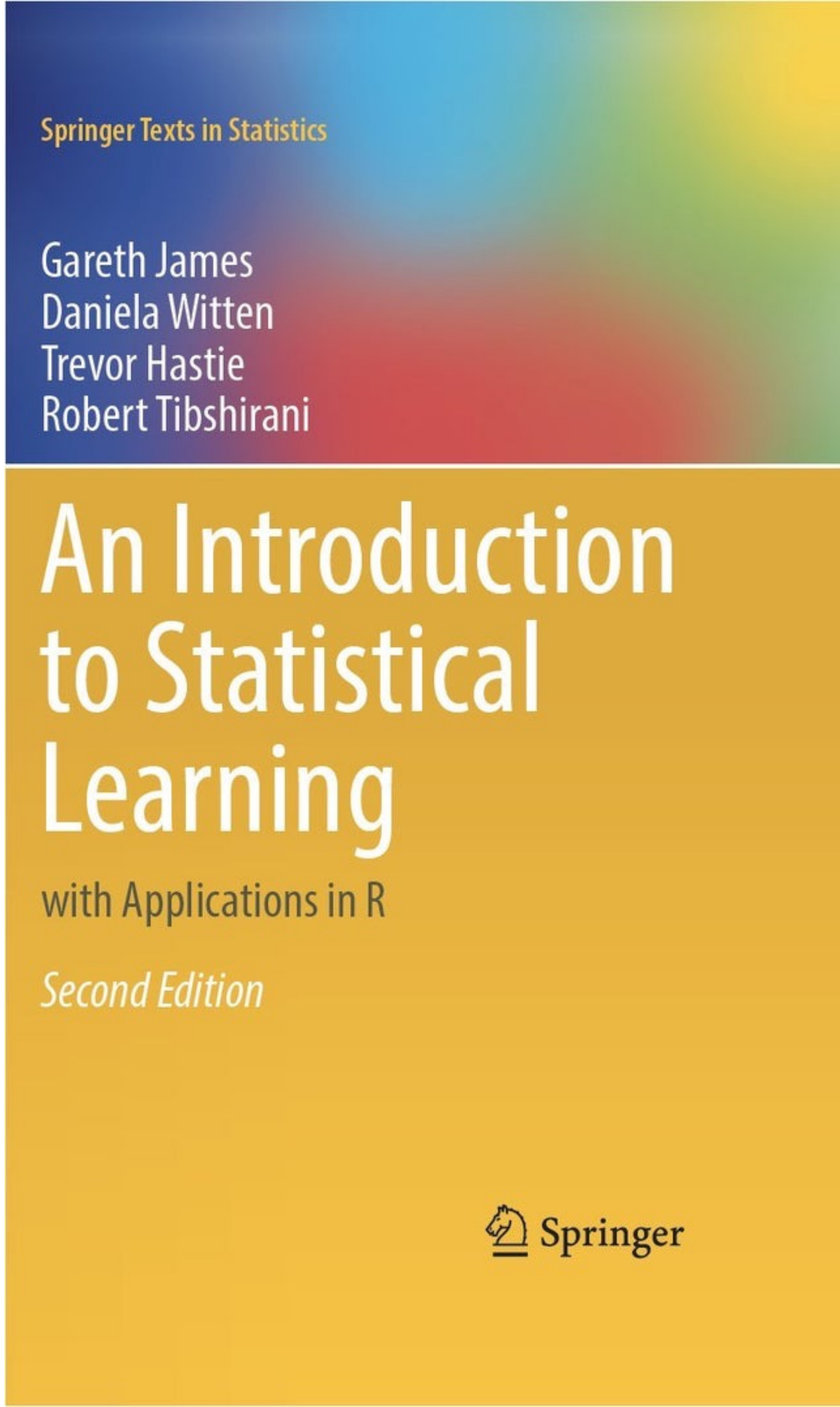
Utrecht University

This week

- Monday: Missing data
- Tuesday: What to do about missing data
- **Thursday: Clustering #1**

Reading materials about clustering (this week & next)

- Selected paragraphs from **Introduction to Statistical Learning (ISLR)** §12.1 and 12.4
- "Mixture models: latent profile and latent class analysis" [Oberski, 2016] §1, §2
- http://daob.nl/wp-content/papercite-data/pdf/oberski2016mixturemodels.pdf



Optional, much more in-depth material

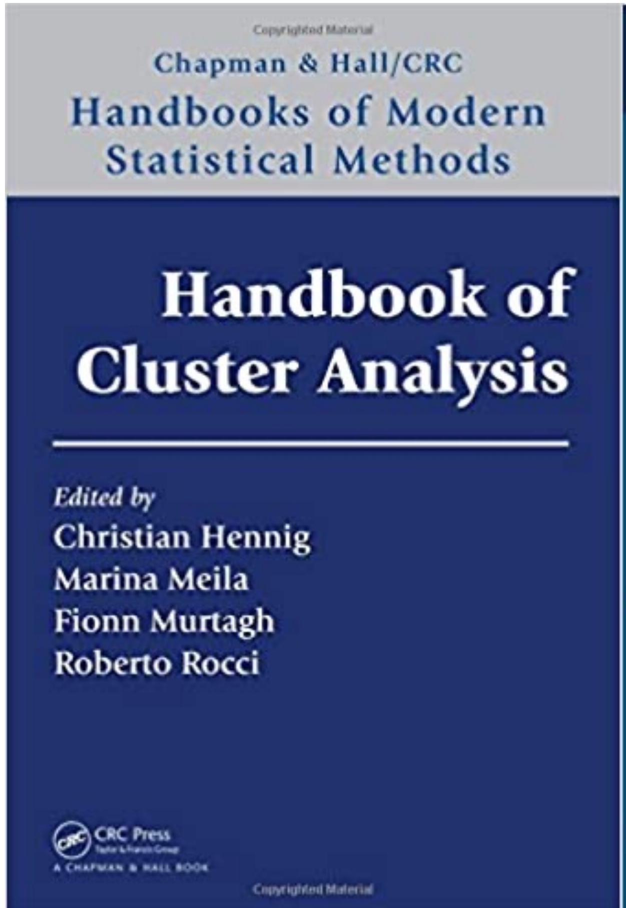
Clustering strategy and method selection (ch. 31),
[*https://arxiv.org/pdf/1503.02059.pdf*](https://arxiv.org/pdf/1503.02059.pdf)

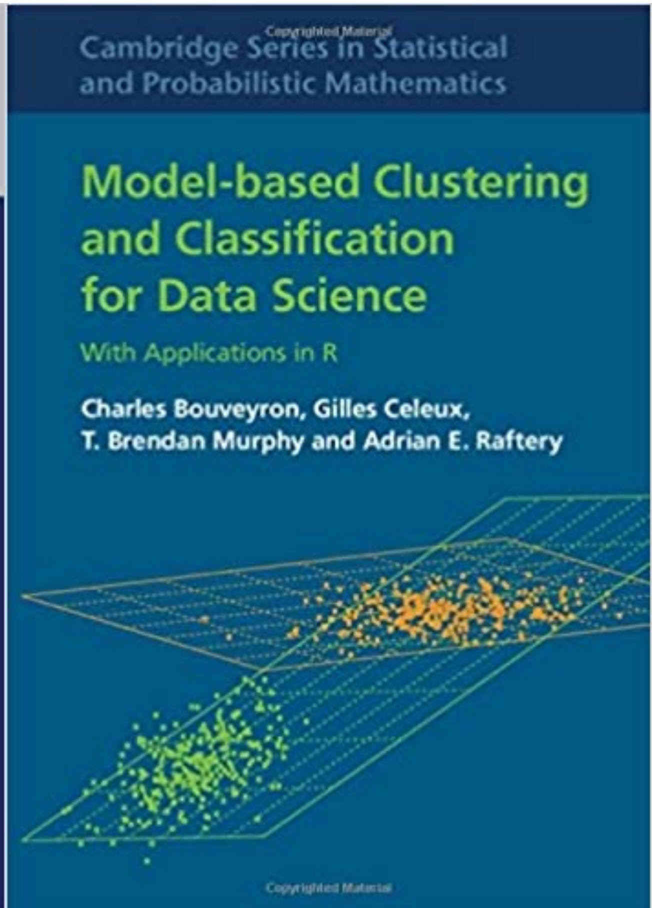
Handbook of Cluster Analysis

Hennig et al. (2016)

Model-based Clustering and Classification for Data Science

Bouveyron et al. (2018)





Clustering

Find subgroups (clusters) of similar examples in a database

Why clustering?

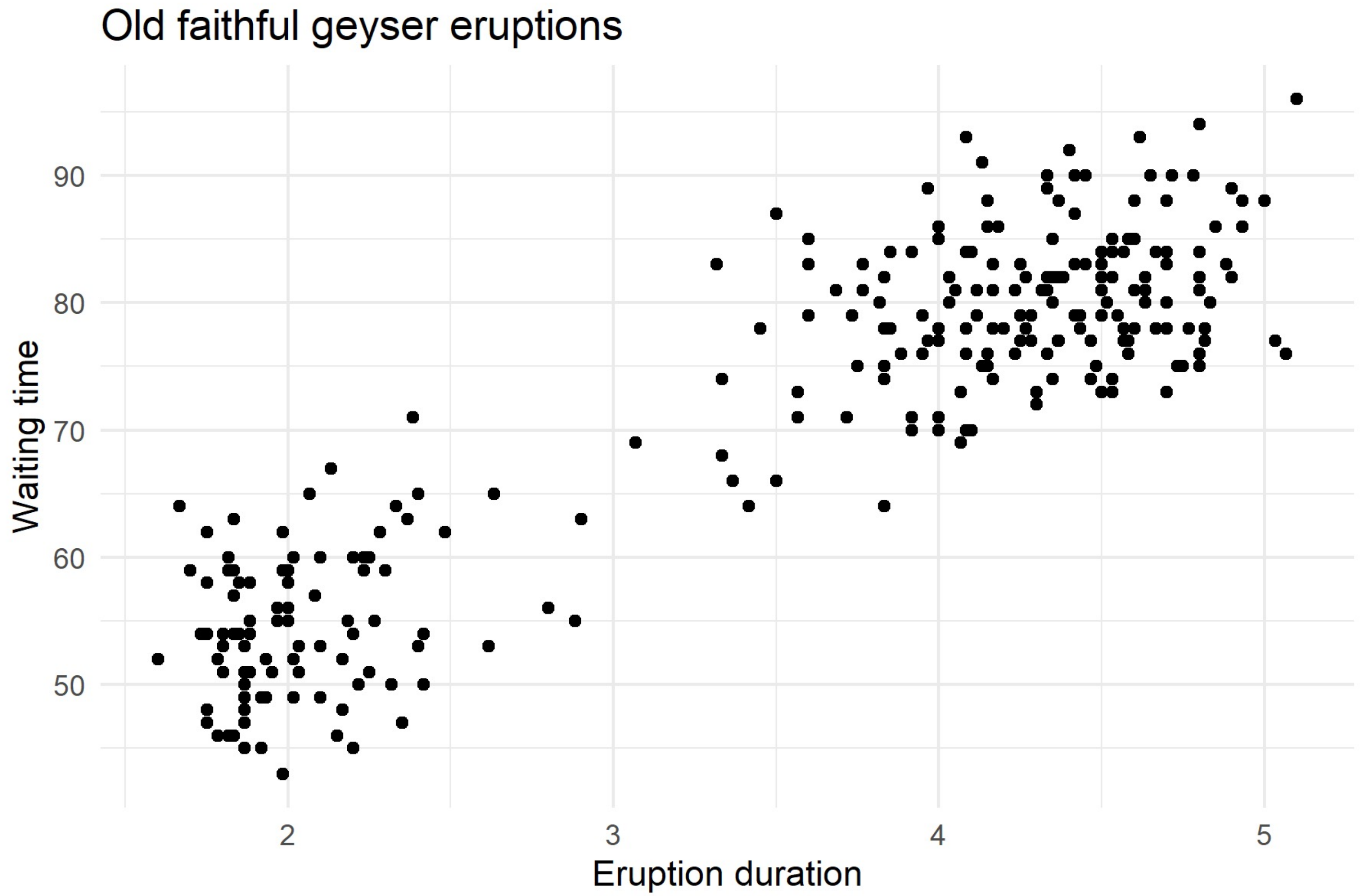
- Unsupervised: expect groups in our data, but were not able to measure them
  - potential new subtypes of cancer tissue
- We want to summarize features into a categorical feature to use in further decisions/analysis
  - subgrouping customers by their spending types

Some applications of clustering

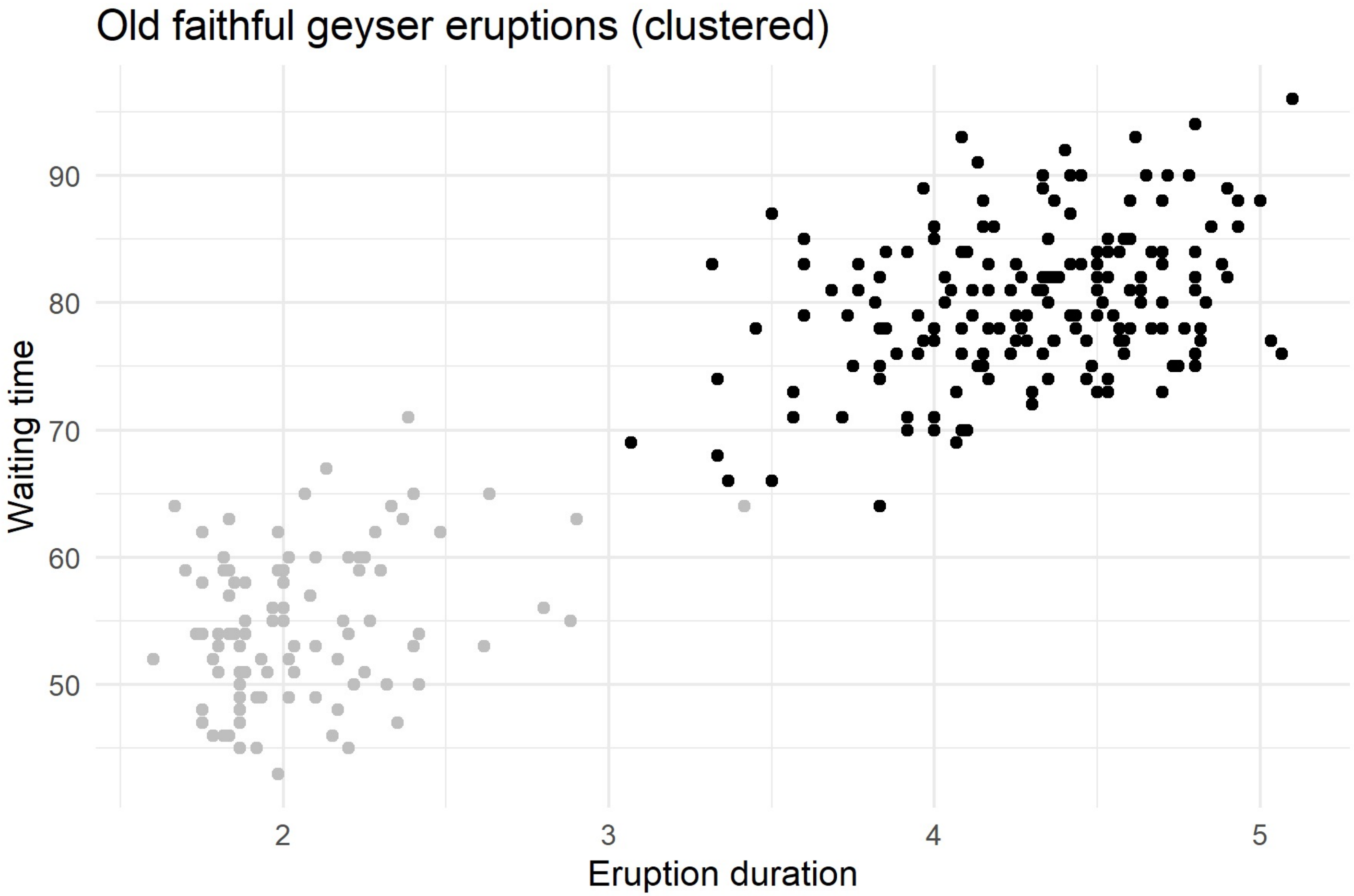
- Intermediate step for other fundamental data mining problems
- Collaborative filtering
- Customer segmentation
- Data summarization
- Dynamic trend detection
- Multimedia data analysis
- Biological data analysis
- Social network analysis

Aggarwal & Reddy (2019)

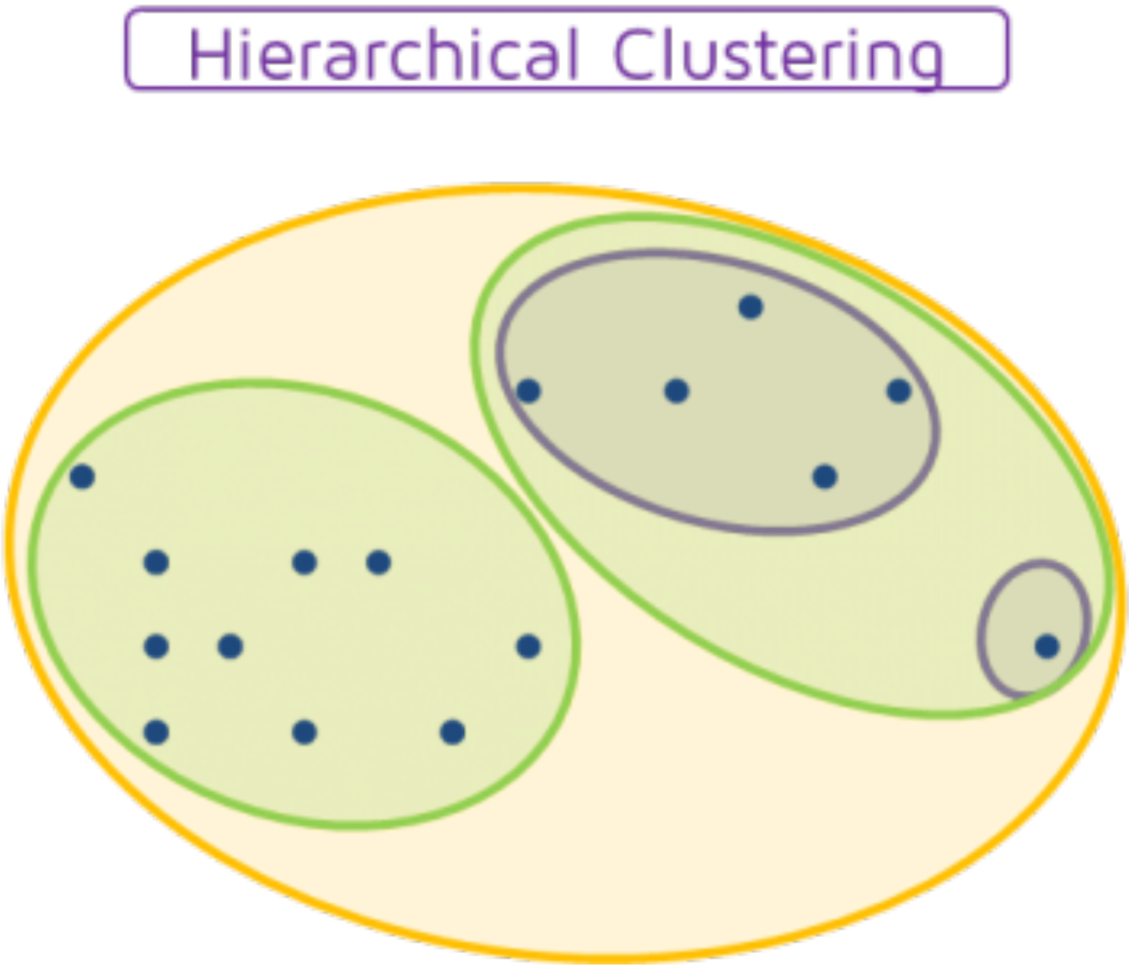
Old Faithful: two types of eruption?

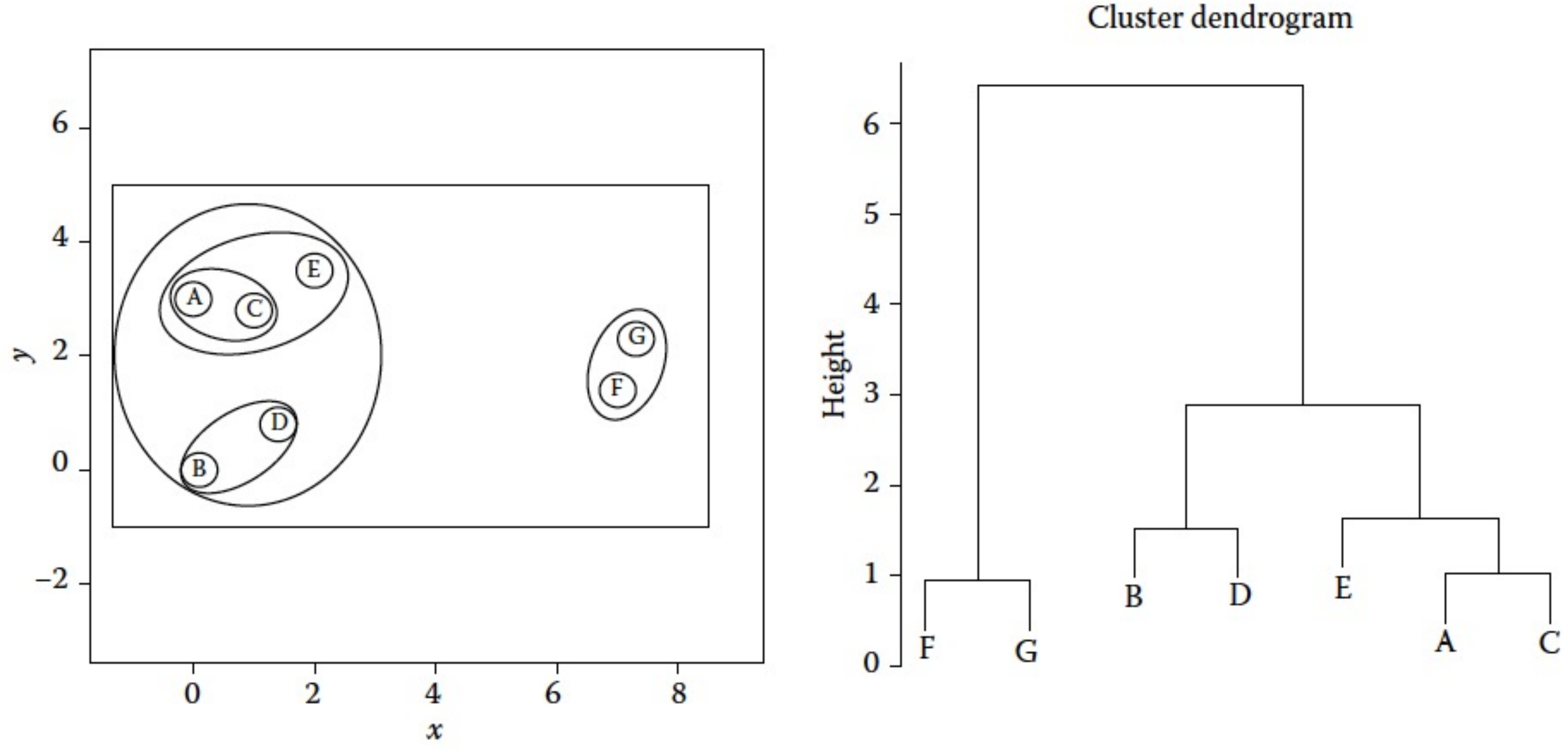


Old Faithful: two types of eruption?

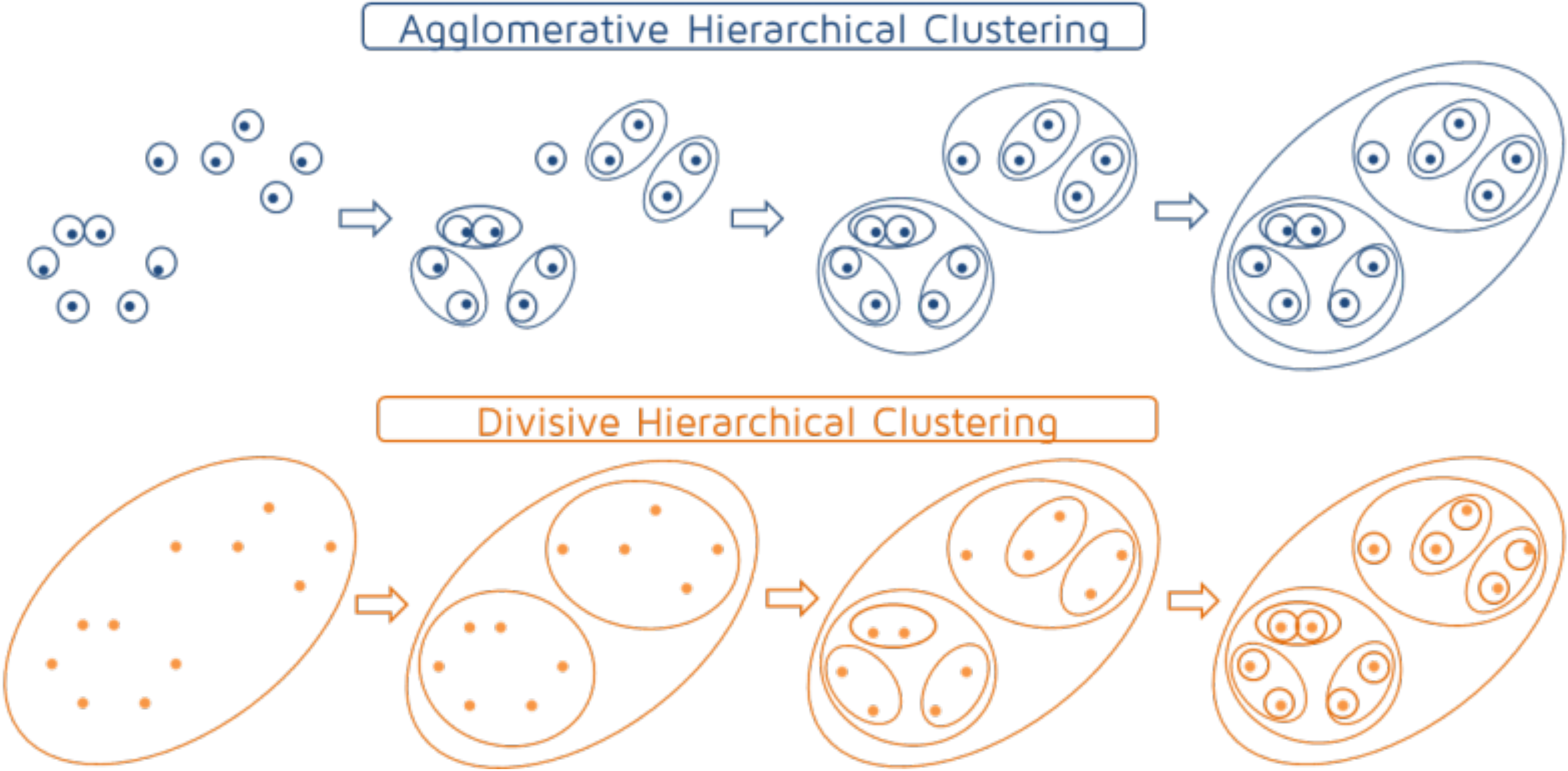


Clustering types





Source: Hennig et al. (2015) *Handbook of cluster analysis*.



Source: T. Fuertes
https://quantdare.com/hierarchical-clustering/

Hierarchical clustering

Bottom-up agglomerative clustering

- For each observation, compute the *distance* to all other observations
- Assign all examples to their individual cluster
- Combine *most similar* clusters
- Keep combining clusters until there is only one cluster left
- Select number of clusters for the final solution

(Divisive: start with ***one*** *cluster and keep splitting most* ***different****)*

Hierarchical clustering

In R:

```r
distances <- dist(faithful,    method = "euclidean")
result    <- hclust(distances, method = "average")
```

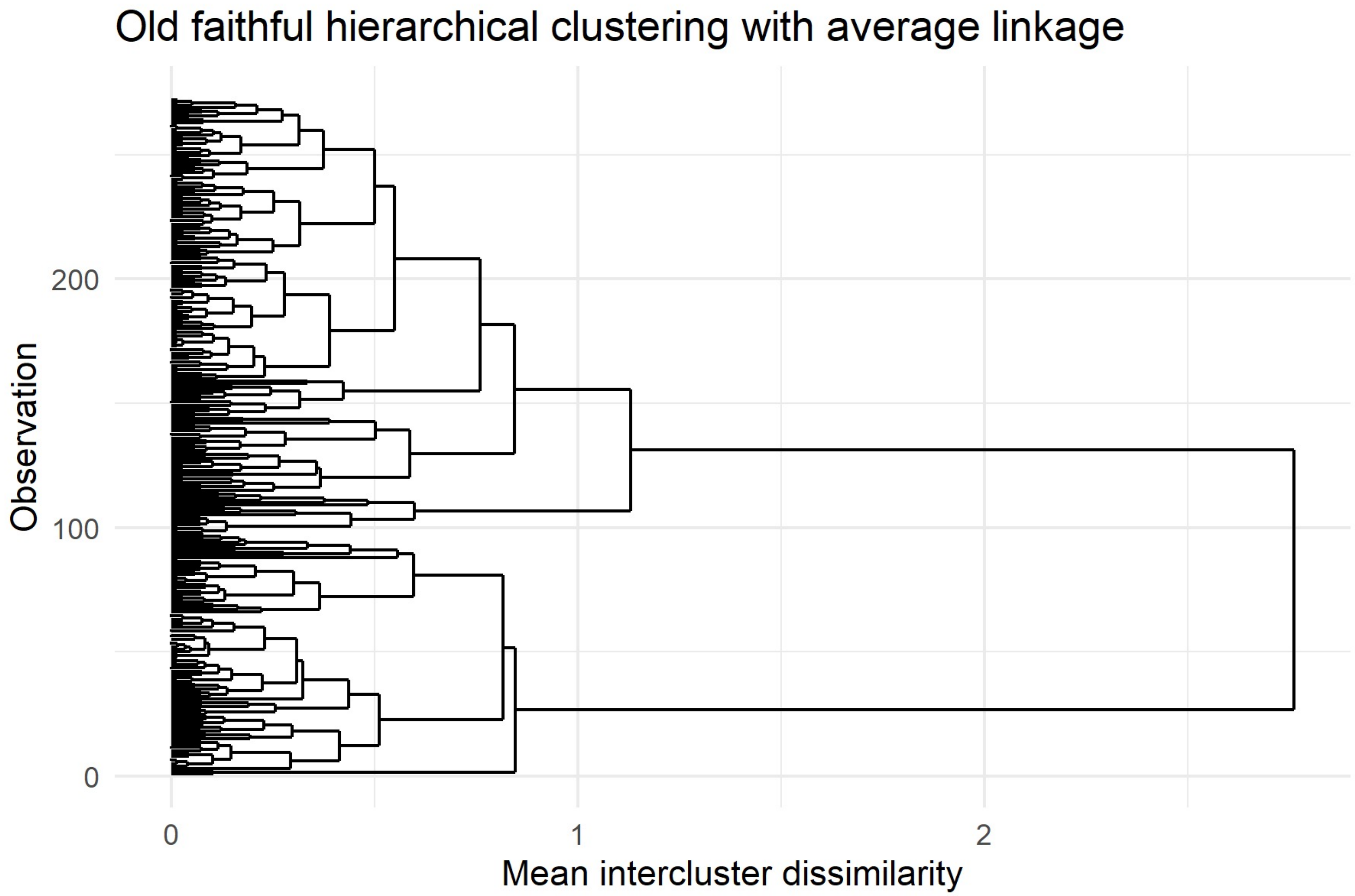
Then we can plot the dendrogram with `plot()` or `ggdendrogram`

```r
library(ggdendro)
ggdendrogram(result)
```

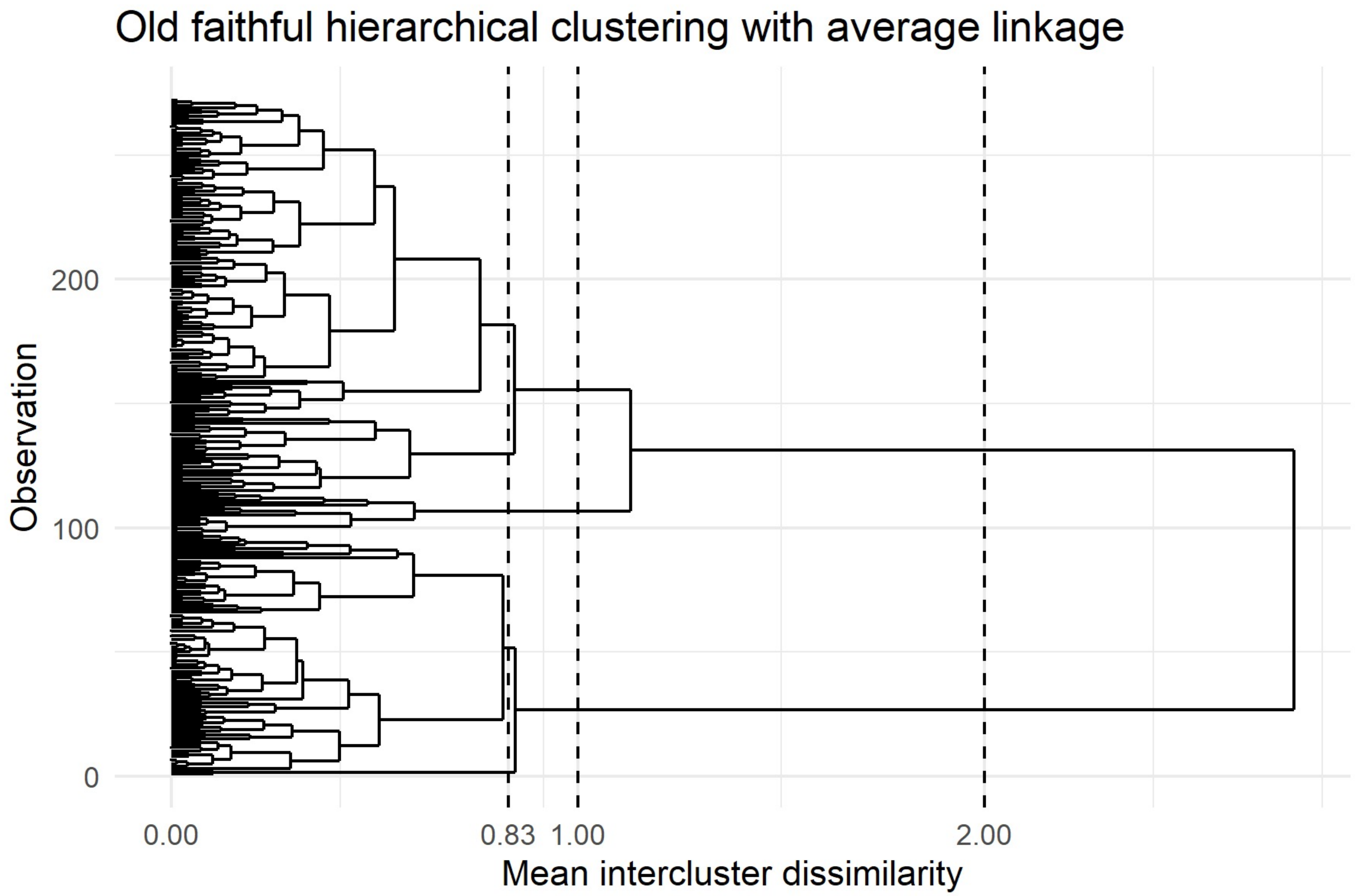
Then, select the number of clusters using a cutoff point

```r
cutree(result, h = 2)
```

Hierarchical clustering

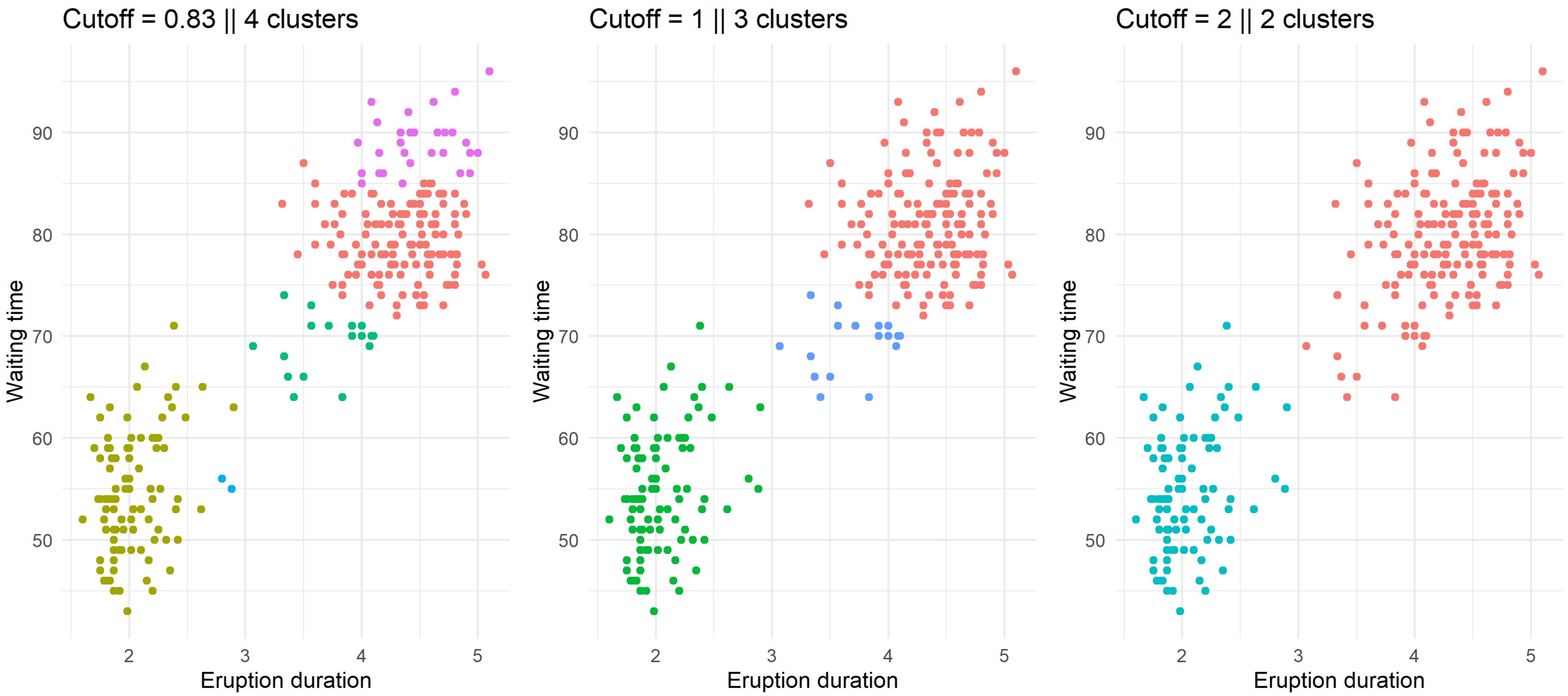


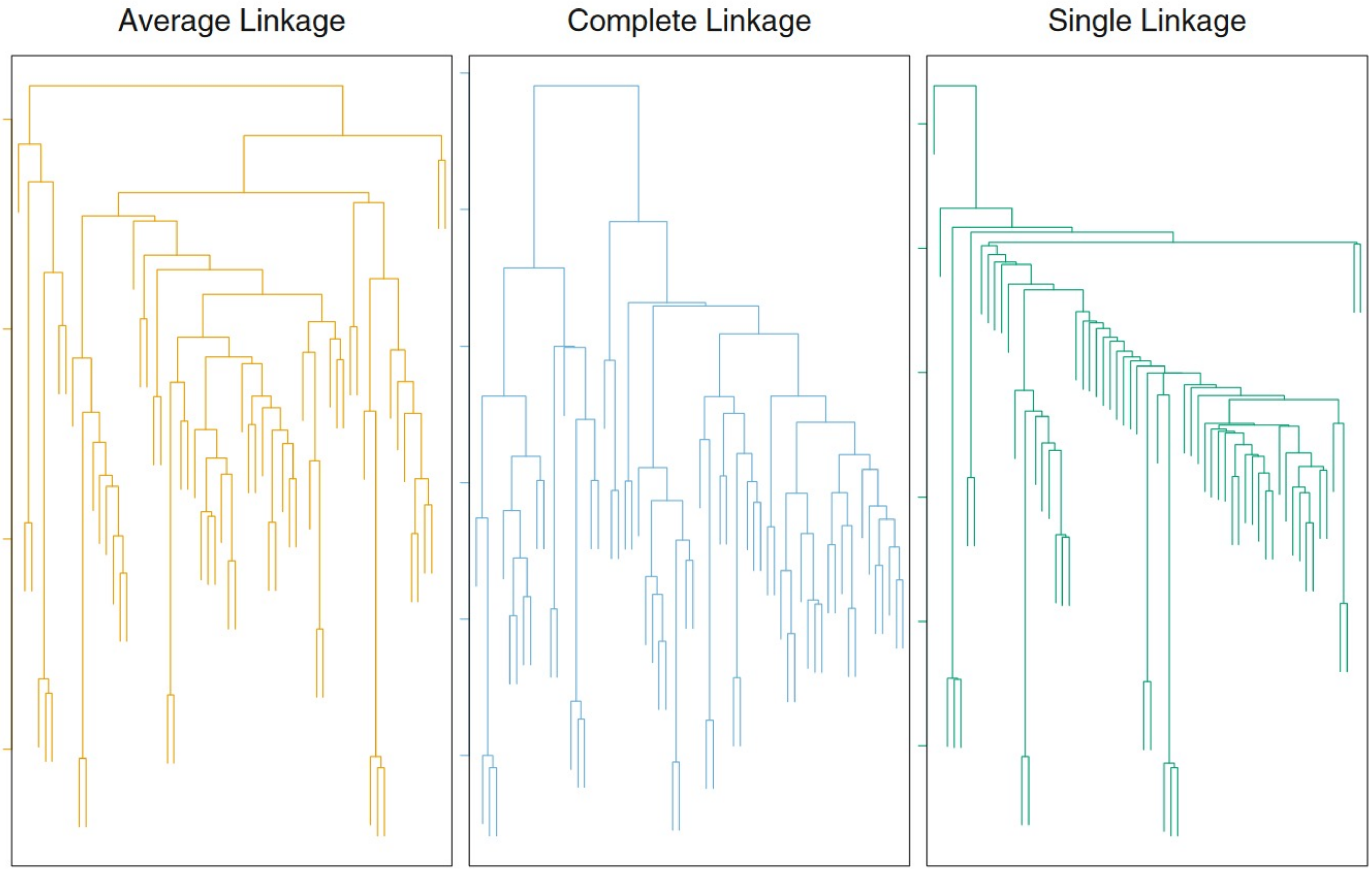
Hierarchical clustering



https://www.menti.com/uf9b1miv22

Hierarchical clustering





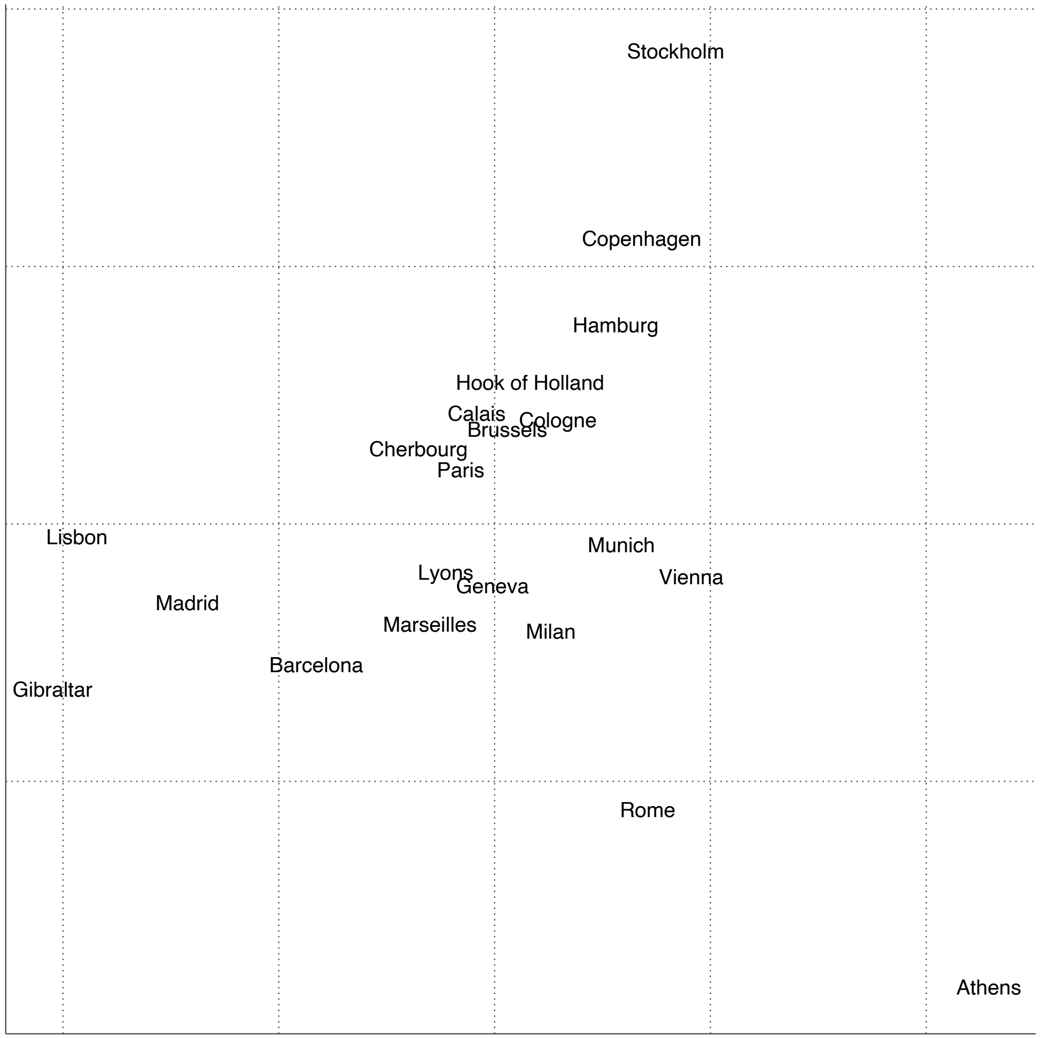
Utrecht University

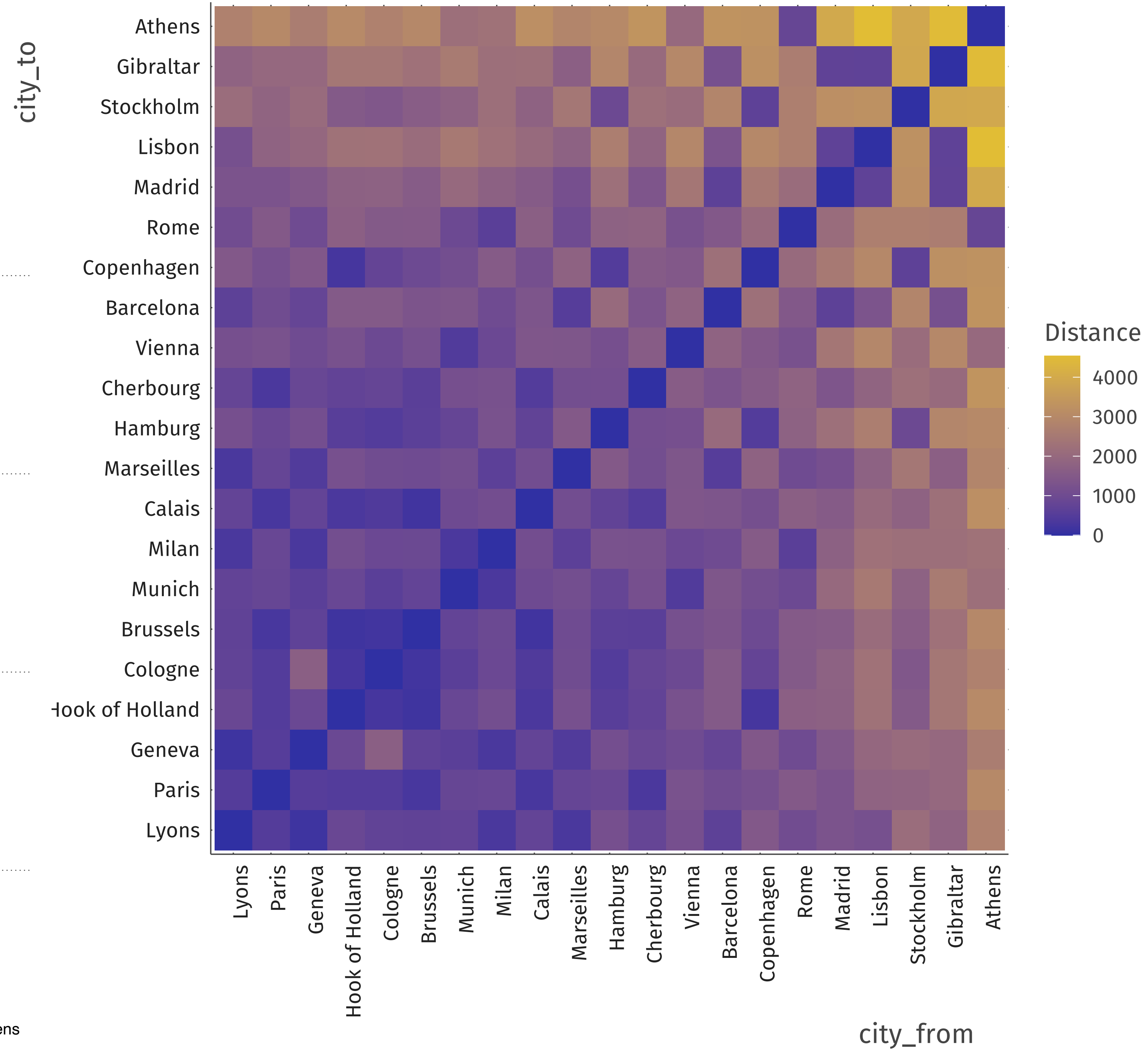
See ISLR, Table 10.2 for explanation of “linkage” options

Note: scaling

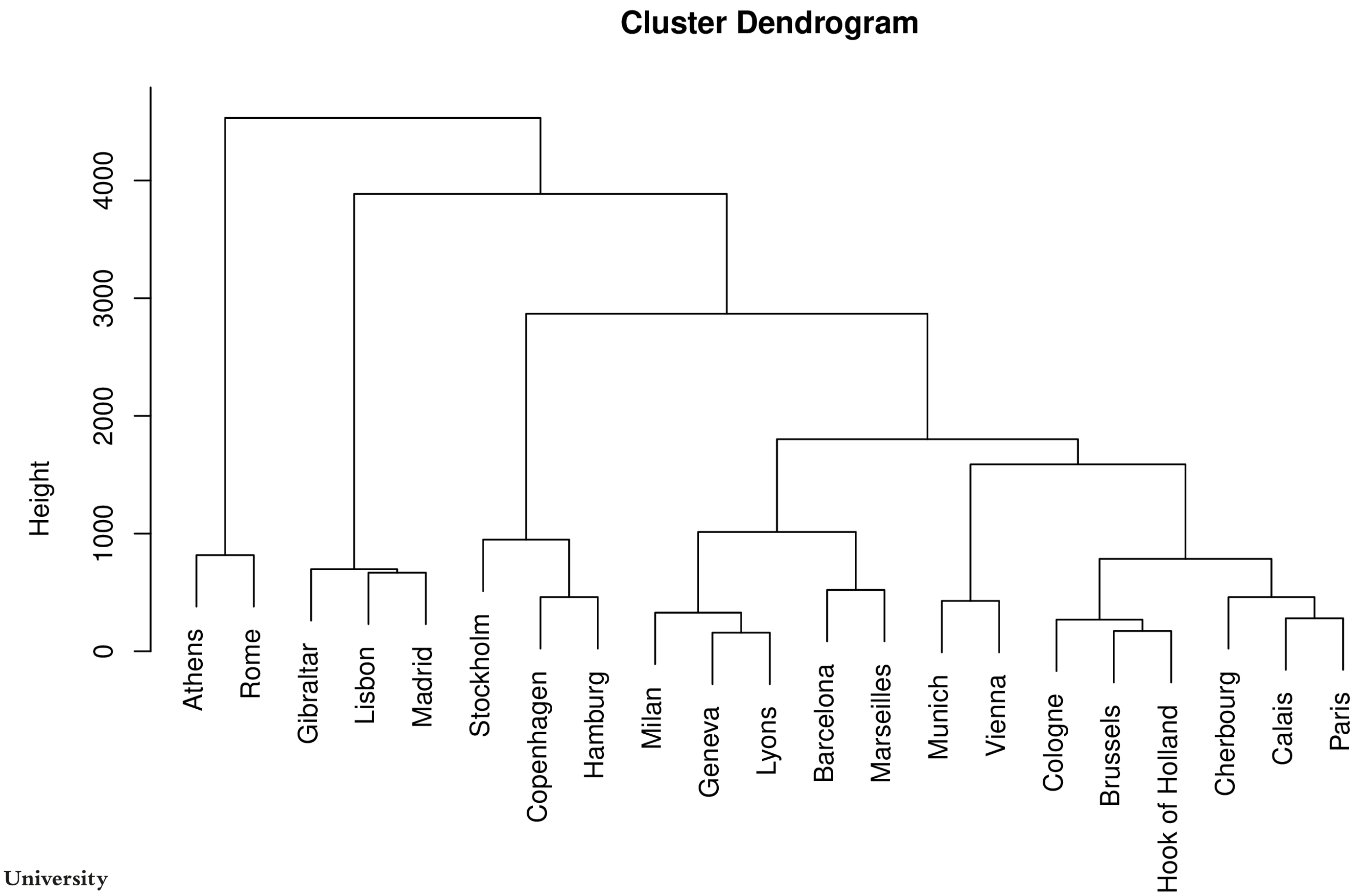
- It is generally a good idea to measure your features in the same scale before entering them into a clustering algorithm
- Otherwise, height in cm will be more important in the distance computation than width in m
- Generally, you want the features to have a similar scale
- This can be done by standardization, or z-transformation: subtract the mean from each feature and divide by its observed standard deviation
- Changes the interpretation of the values, but not their association

“**Distance**” matrix



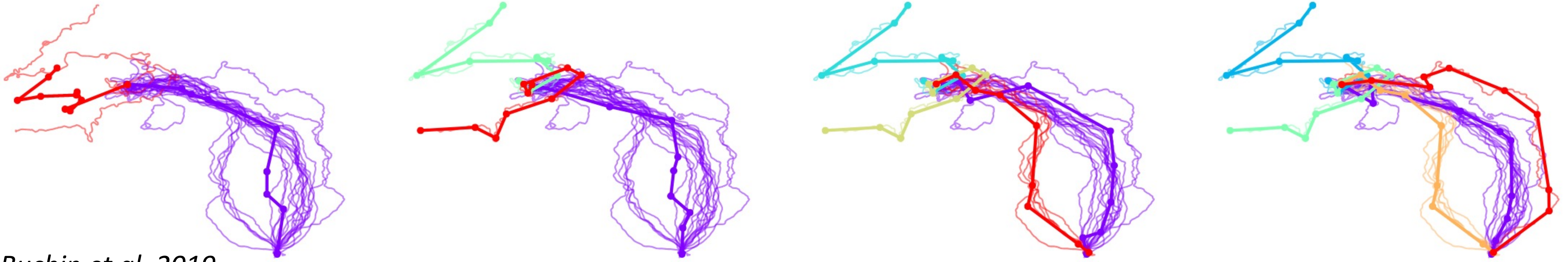


```r
plot(hclust(eurodist))
```



“**Distance**” can mean lots of things

- **Continuous:** Euclidean, maximum, Manhattan, Minkowski, …
- Time series: ”Dynamic time warping”, Fréchet, cross-corr., wavelets, …
- Networks: Modularity, Shortest path, …
- Text/DNA: Edit distance, Hamming distance, TF-IDF distance, ..
- Images: “Structural similarity”, GAN loss…

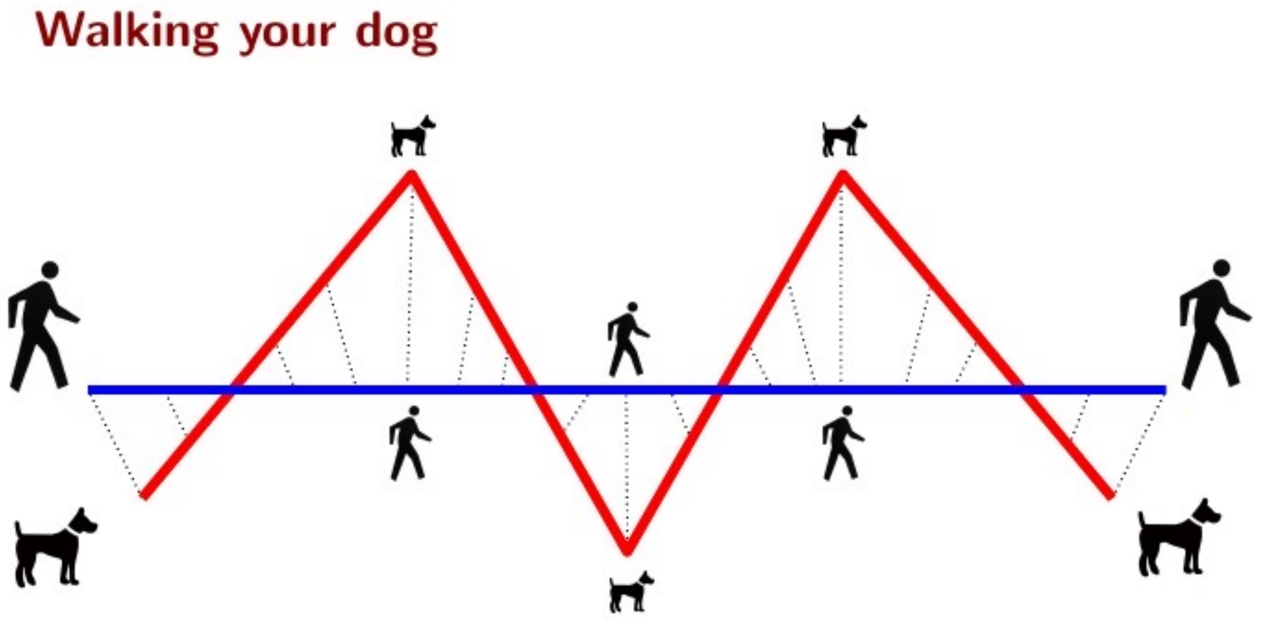


Buchin et al. 2019

Figure 1: Example of a (k, ℓ)-clustering for the flight paths of a pigeon with the number of clusters k increasing from 2 (left) until 5 (right) and the complexity of the clusters being $\ell = 10$. Trajectories belonging to the same cluster are shown in the same color. For each cluster, a center trajectory generated by the algorithm is shown using thick lines of the same color.

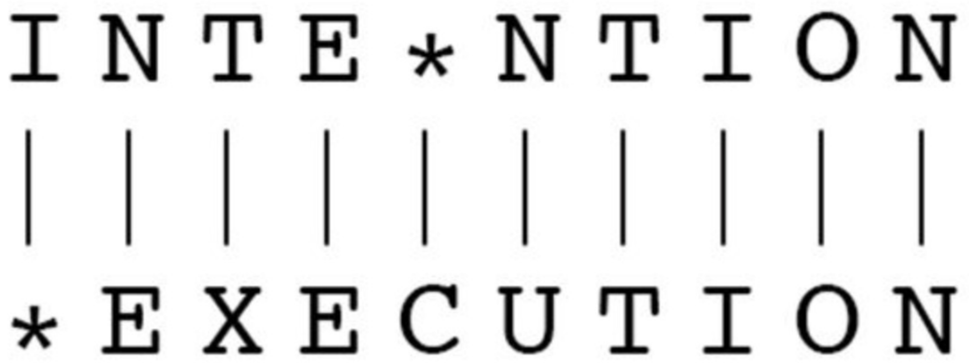
Some example distances that aren't Euclidean but useful for specific data types

Fréchet distance for time series/curves



The **Fréchet distance** between the curves is the minimum leash length that permits such a walk

Minimum edit distance for text/DNA



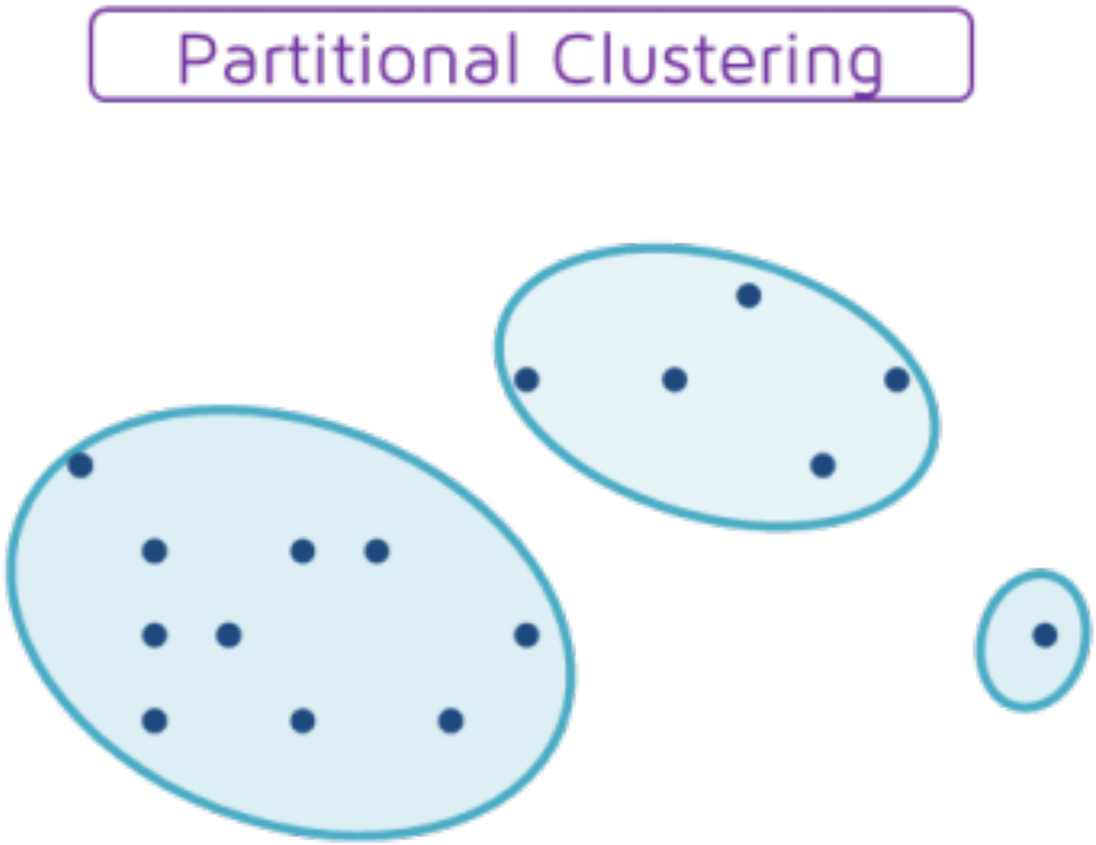
(edit distance = 5)

https://www.menti.com/uf9b1miv22

Hierarchical clustering: conclusion

- Tree-based representation - dendrogram
- Determine number of clusters afterwards
- Different distance metrics possible
- Different agglomeration methods (“linkage” rules) possible
- Taking distance matrix as input → very flexible technique
- Distance matrix N×N → can be tricky when N is large (esp. divisive)

Clustering types



Source: T. Fuertes
https://quantdare.com/hierarchical-clustering/

K-means clustering algorithm

1. Randomly assign examples to K clusters

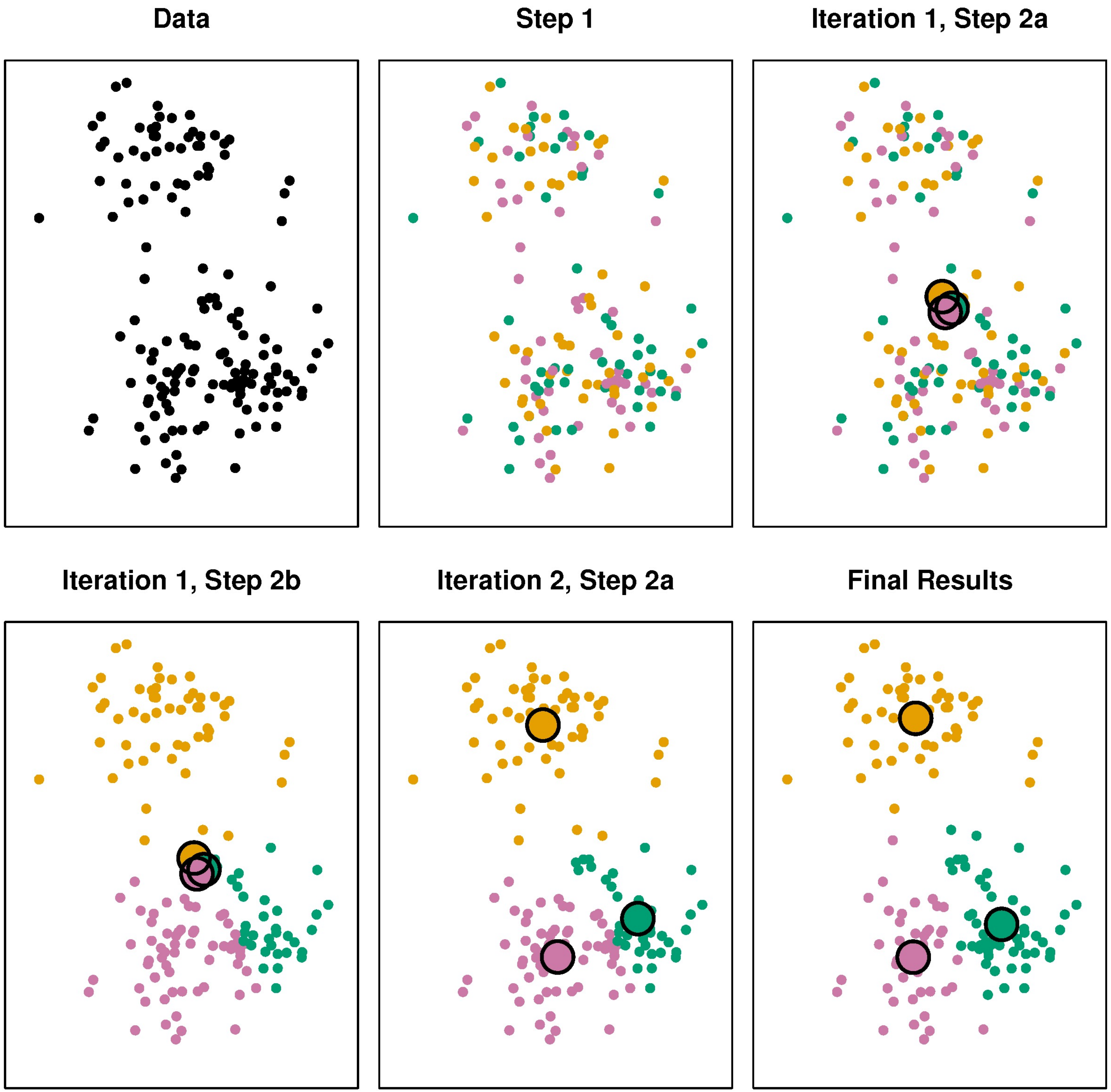
2a. Calculate the centroid (per-feature mean) for each cluster

```
faithful %>% group_by(cluster) %>% summarize_all(mean)

#>   cluster eruptions waiting
#>     <int>     <dbl>   <dbl>
#> 1       1      3.69    73.4
#> 2       2      3.30    68.6
```

2b. Assign each example to the cluster belonging to its closest centroid

3. If the assignments changed, go to step 2a, else stop

K-means clustering



ISLR, page 389

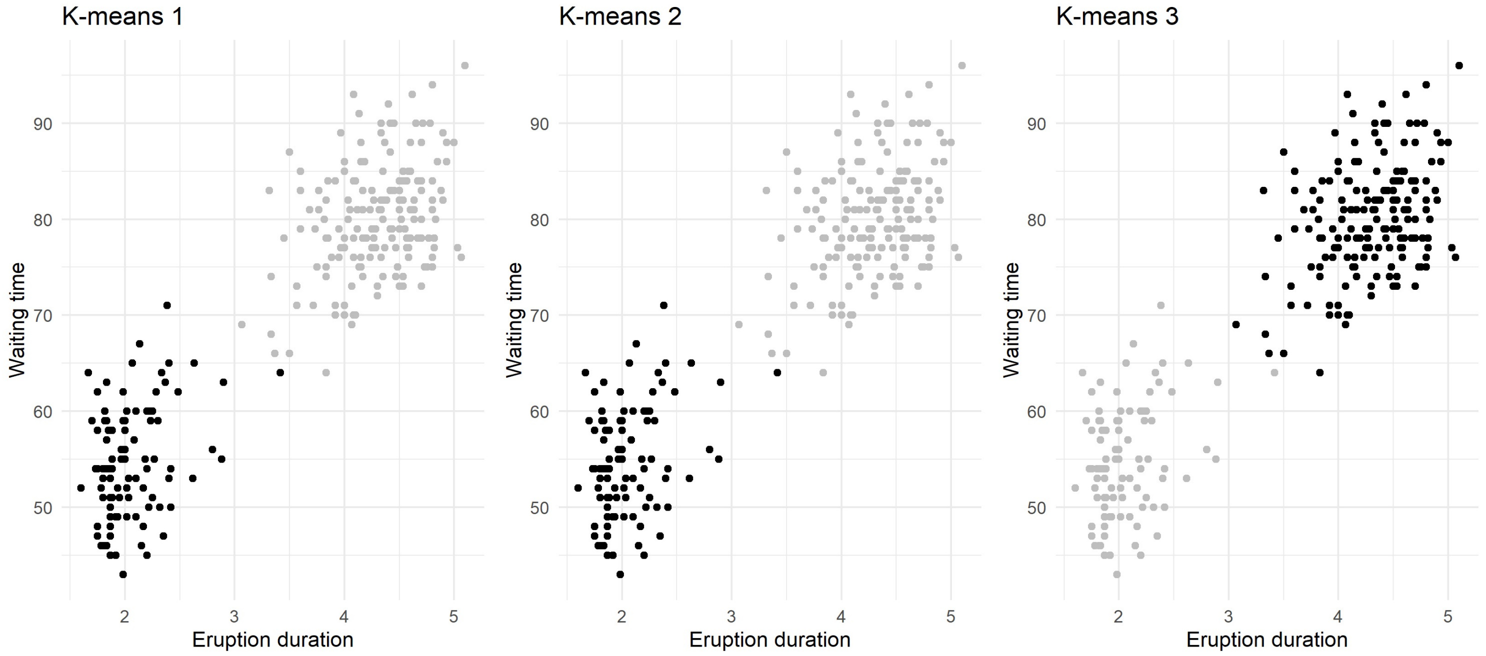
K-means clustering

- K (and, in theory, the distance metric) are hyperparameters (ISLR Sec 10.3)

  (in practice, distance is almost always Euclidean (sum of squares criterion))
- K is determined in advance by the analyst
- Could be based on knowledge about the data or the goal of the analysis
  - Perhaps there are generally 2 types of geyser eruption because of physics
  - We may have resources to approach customers in at most 3 different ways
- Other criteria discussed later.

K-means clustering

- Because the initialization is random, the result is random
  - Label switching: cluster 1, 2, 3 may end up in each other's locations
  - Some examples at the boundary may end up in different clusters altogether
  - Use multiple starts to obtain the best solution

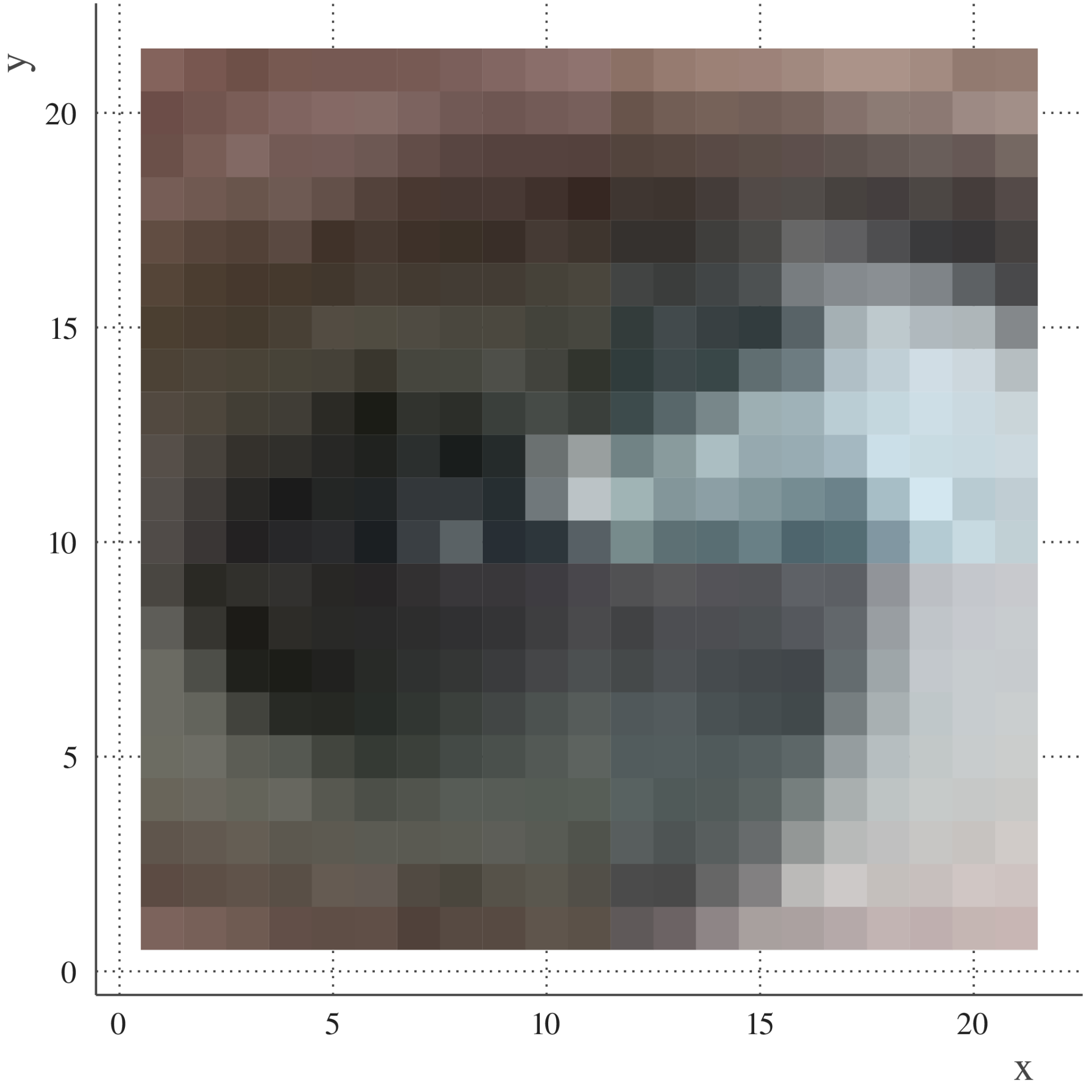


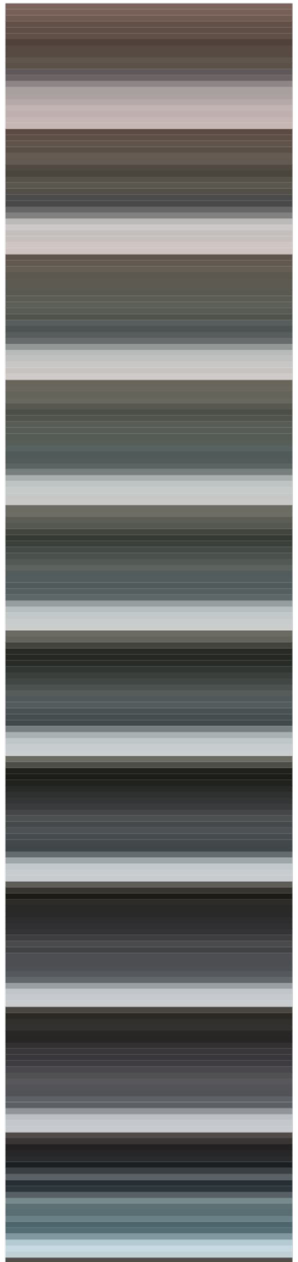
K-means clustering

K-means clustering applied to images is called *"vector quantization"*

Goal: image compression → less storage!

Cluster pixels, then replace them by their cluster centroid





r,	g,	b
0.49,	0.39,	0.36
0.34,	0.29,	0.26
0.66,	0.63,	0.62
0.36,	0.29,	0.26
0.29,	0.27,	0.24
0.51,	0.50,	0.51
0.37,	0.33,	0.30
0.36,	0.36,	0.33
0.40,	0.42,	0.42
0.41,	0.40,	0.35
0.34,	0.36,	0.34
0.36,	0.39,	0.39
0.42,	0.42,	0.38
0.27,	0.29,	0.27
0.33,	0.37,	0.38
0.42,	0.42,	0.39
0.23,	0.25,	0.24
0.27,	0.30,	0.31
0.42,	0.42,	0.39
0.20,	0.21,	0.21
0.26,	0.28,	0.29
0.37,	0.36,	0.35
0.19,	0.19,	0.20
0.30,	0.32,	0.33
0.29,	0.27,	0.25
0.22,	0.22,	0.23
0.32,	0.33,	0.34
0.31,	0.29,	0.28
0.35,	0.38,	0.40
0.41,	0.50,	0.53
0.33,	0.31,	0.29

Original

`671 kB`

K = 100

`419 kB`

Original

671 kB

$K = 10$

126 kB

Original

671 kB

$K = 9$

117 kB

Original

`671 kB`

$K = 8$

`101 kB`

Original	$K = 7$
`671 kB`	`93 kB`

Original

671 kB

$K = 6$

79 kB

Original

671 kB

$K = 5$

65 kB

Original

671 kB

$K = 4$

57 kB

Original

671 kB

$K = 3$

36 kB

Original	$K = 2$
`671 kB`	`18 kB`

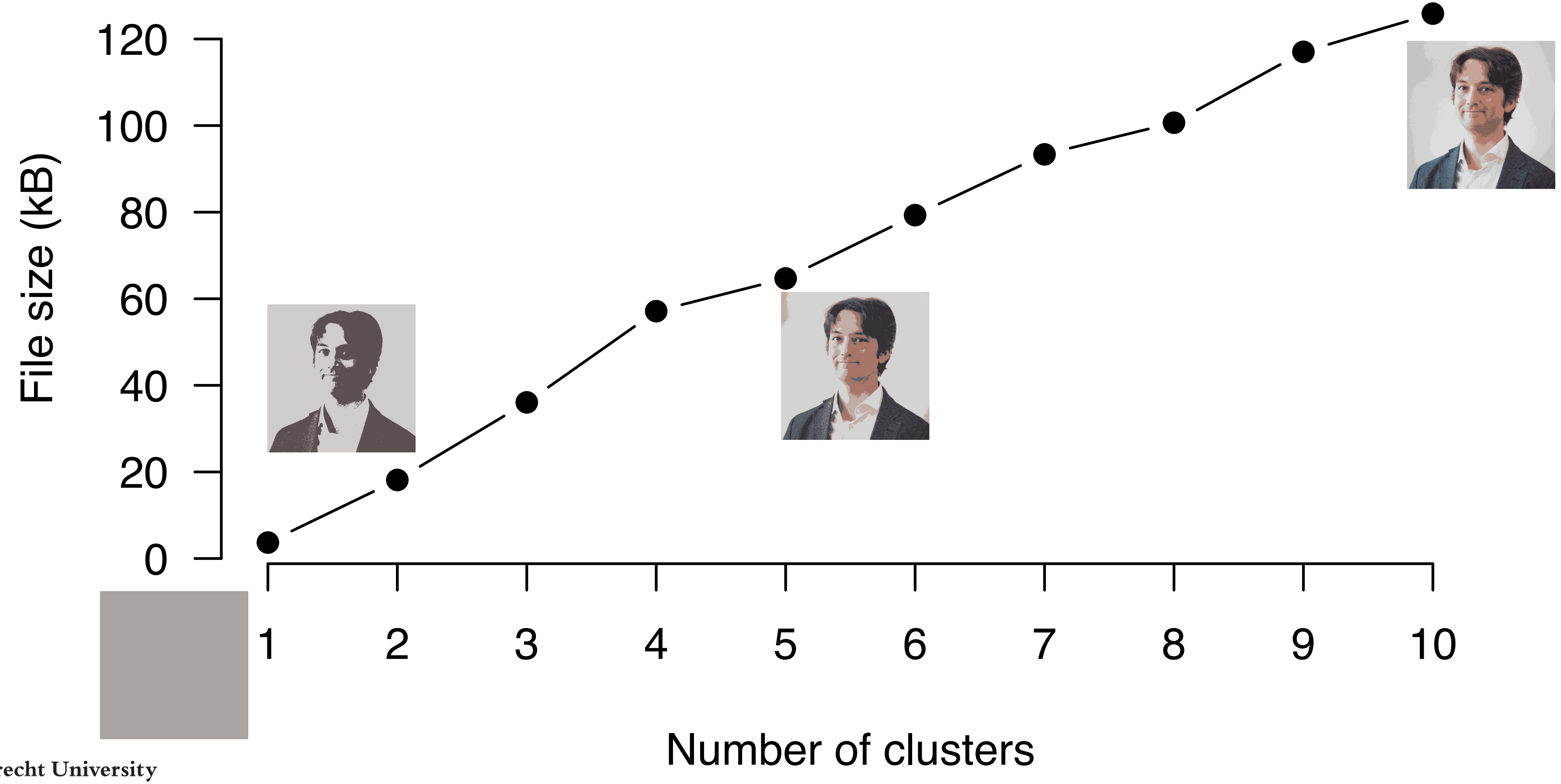
Original

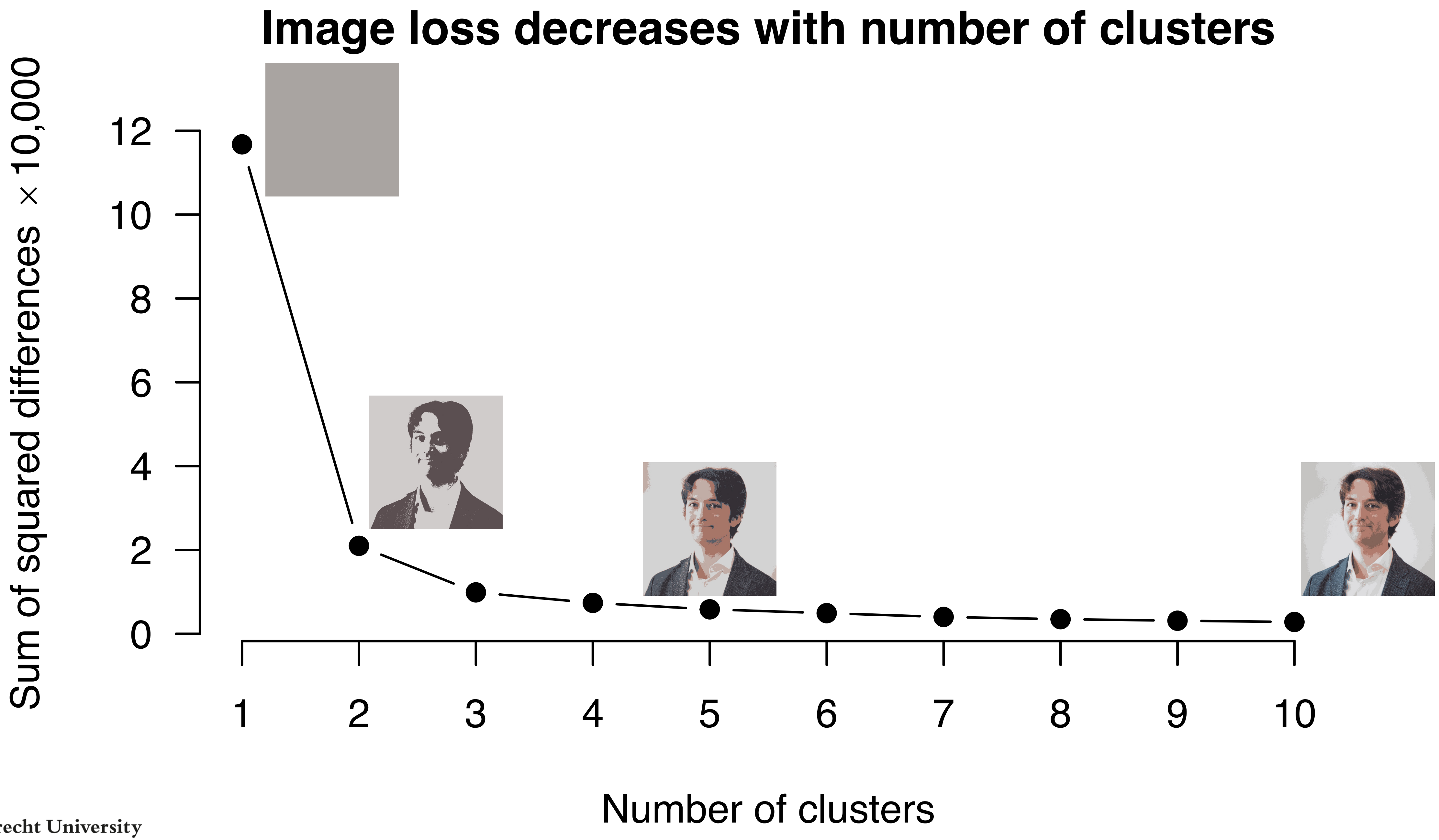
`671 kB`

$K = 1$

`4 kB`

File size increases with number clusters





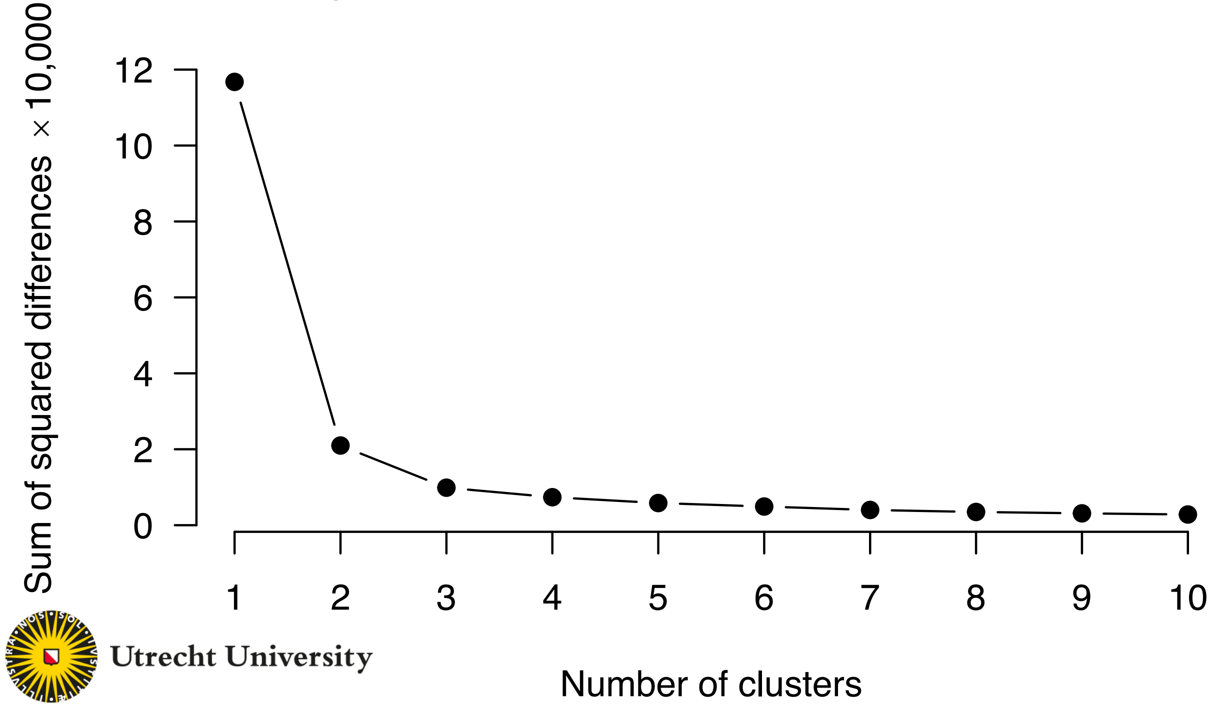
Utrecht University

https://www.menti.com/uf9b1miv22

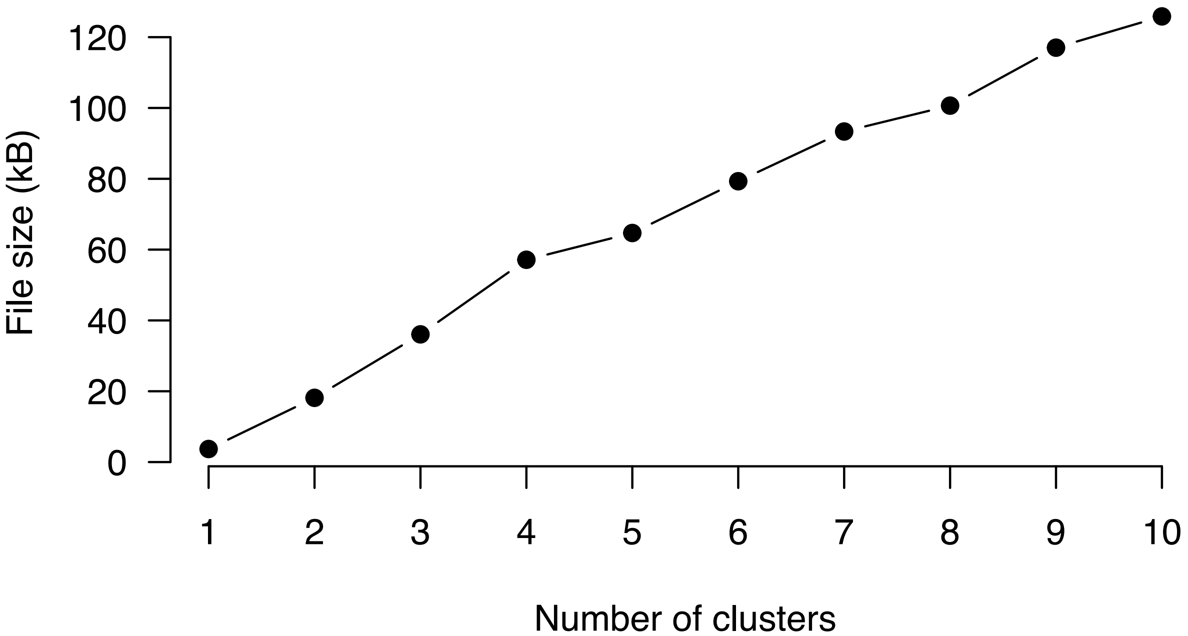
Which of these is the best and why?

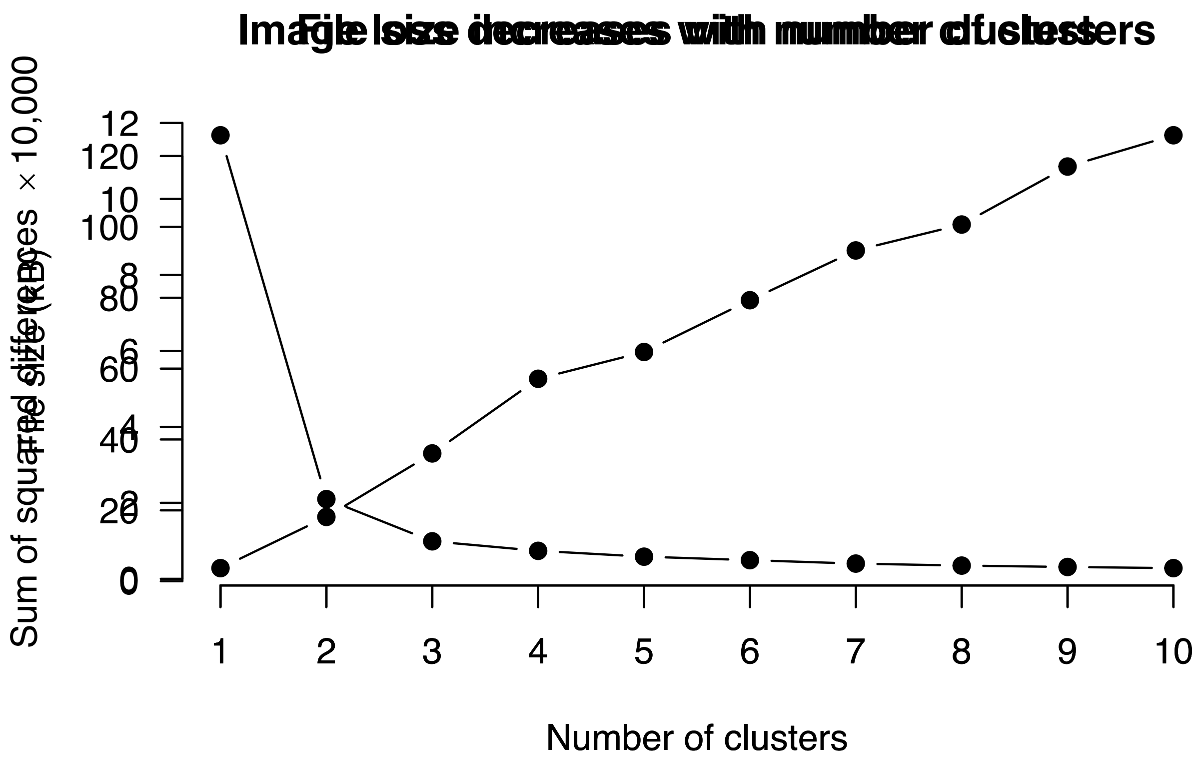
K= 1 5 10 100 640,000

Image loss decreases with number of clusters



File size increases with number clusters





- More clusters gives **better "fit"** in terms of reconstruction of the image
  (compression is less "lossy")
- More clusters gives **bigger file size**
  (solution is more complex, takes more bytes to store)
- So the **model loss and model complexity trade off against each other**
- This is a common theme in (unsupervised) machine learning and you should remember this for model-based clustering lecture

Utrecht University

How to **evaluate** clustering results

1. Use of external information
2. Visual exploration
3. Stability assessment / sensitivity analysis
4. Internal validation indexes
5. (Testing for clustering structure)

Much more info & helpful advice: Clustering strategy & method selection (ch 31 of Handbook of clustering), https://arxiv.org/pdf/1503.02059.pdf

1. External validation

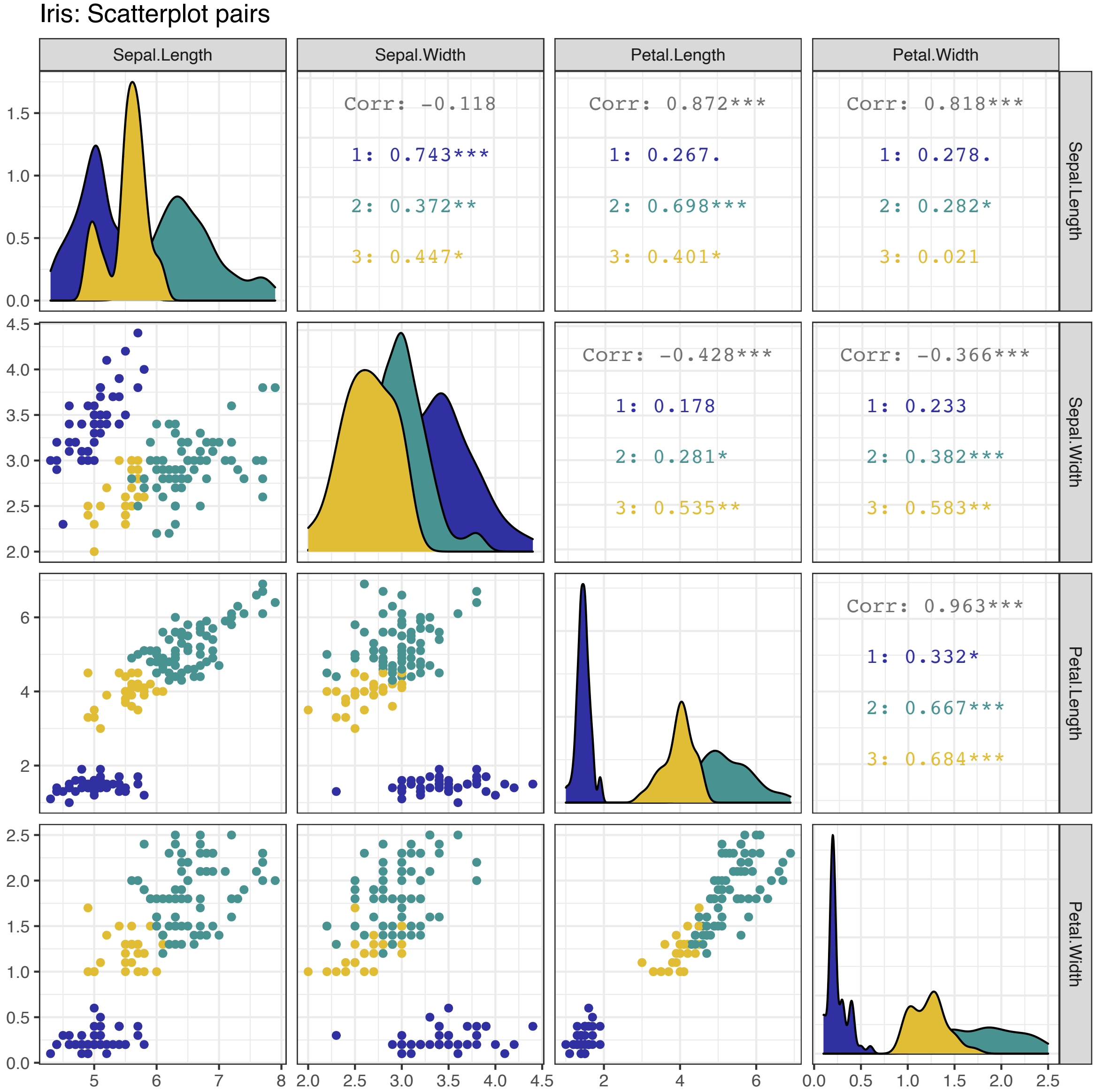
Are the clusters associated with *external* feature Y?

“Making unsupervised supervised”

Examples:

- Are my customer segments based on spending associated with the demographics of the customers?
- Are the geyser eruption types strongly correlated with water pressure or temperature?
- Can I recognize the person in the vector quantized picture?

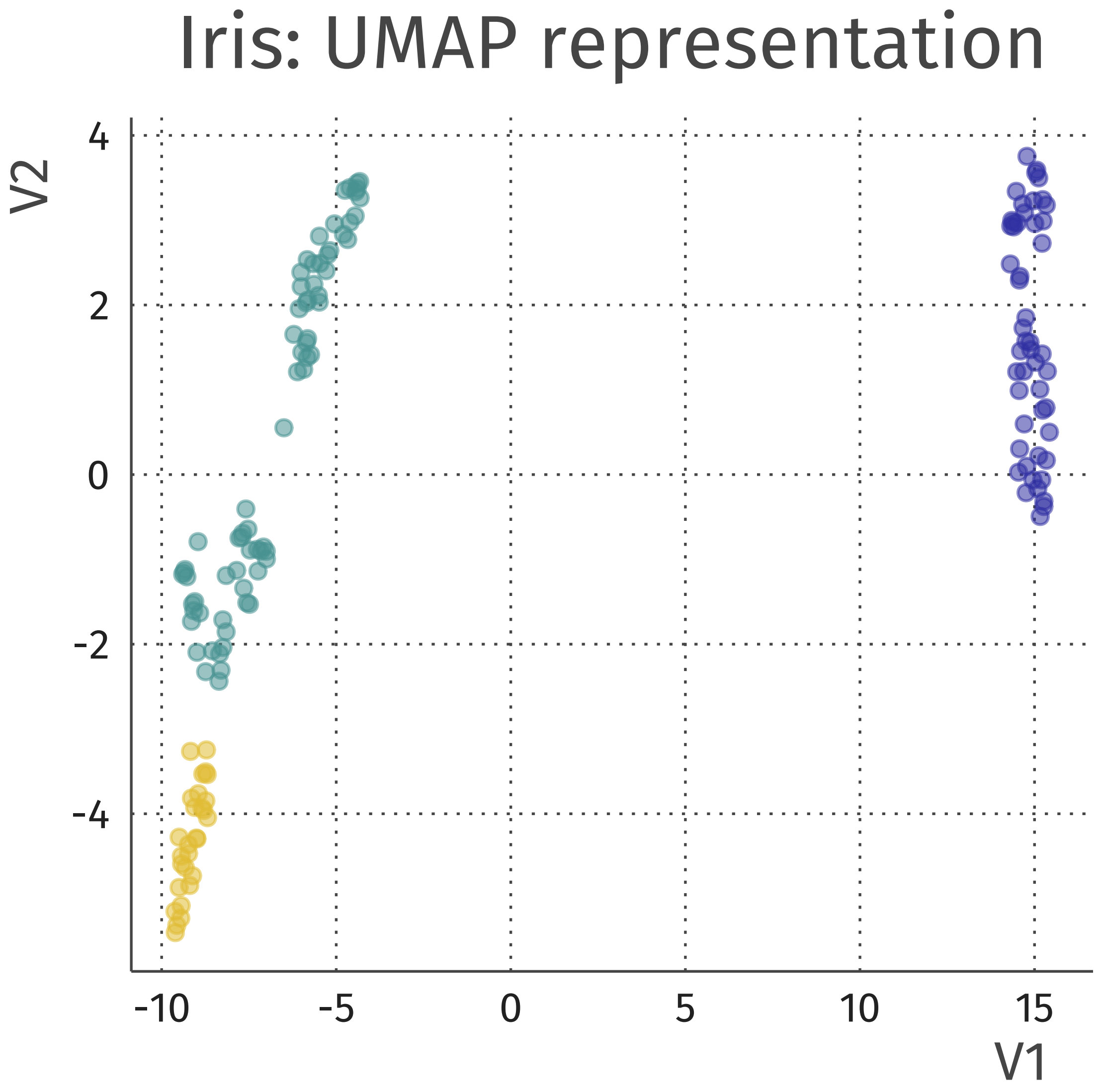
2. Visual exploration



- **Problem**: Kind of hard to see already…
- Wait till you get 1000 variables!

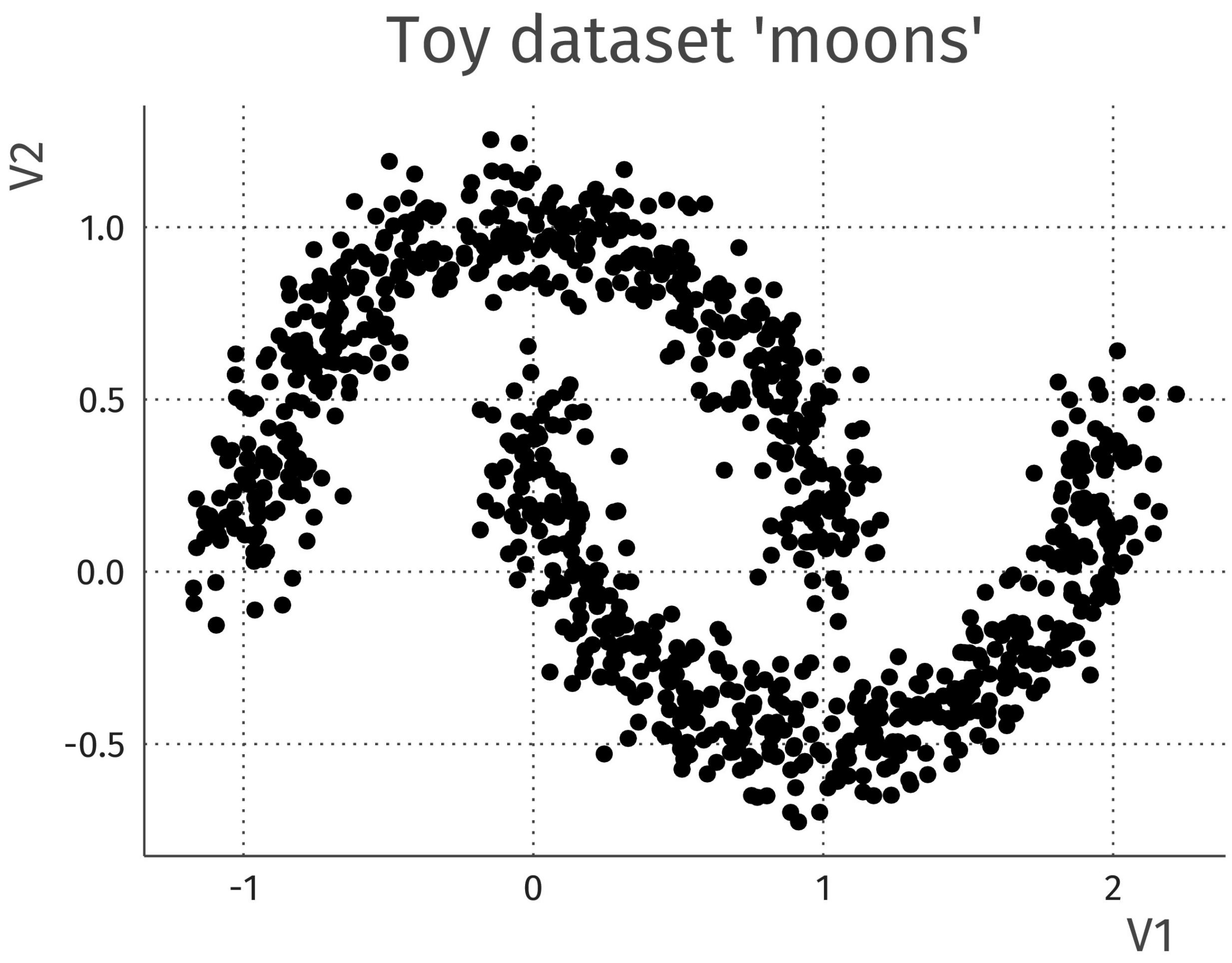
- **New idea**: Reduce variables into 2D "manifold" for visualization
- Popular techniques: UMAP, t-SNE, MDS, Discriminant Coordinates, (PCA)

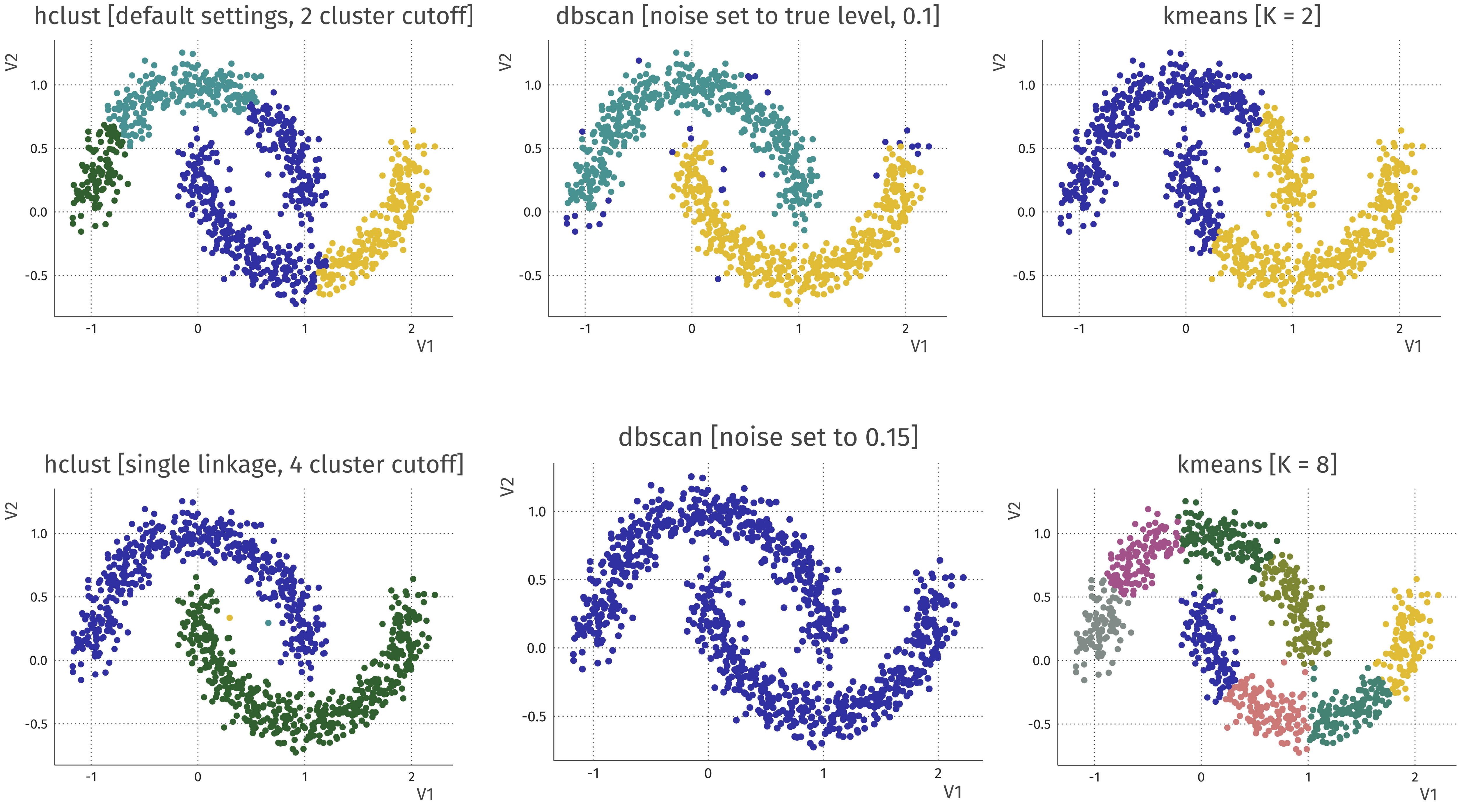
2. Visual exploration (using "manifold")



3. Stability assessment

A.k.a.: Clustering can be fiddly




hclust [default settings, 2 cluster cutoff]
dbscan [noise set to true level, 0.1]
kmeans [K = 2]
hclust [single linkage, 4 cluster cutoff]
dbscan [noise set to 0.15]
kmeans [K = 8]
V1
V2
-1
0
1
2
-0.5
0.0
0.5
1.0

3. Cluster “stability”

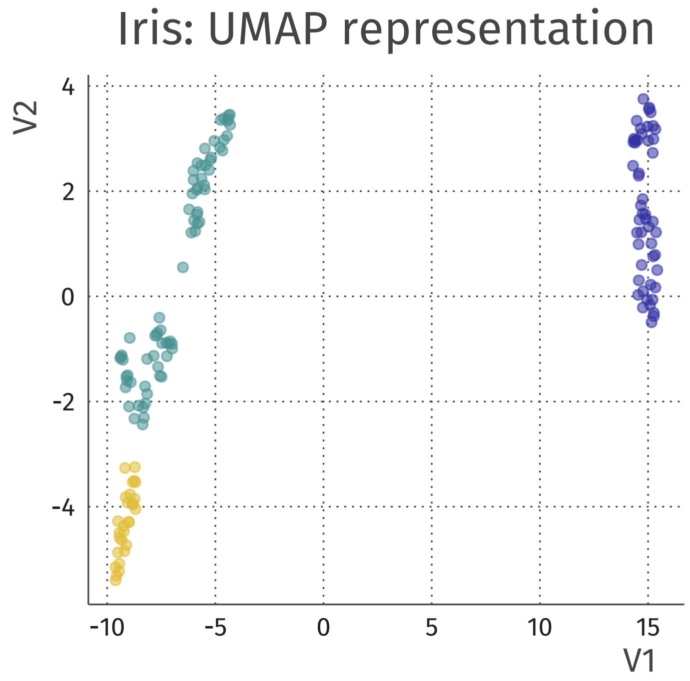
Three “stabilities”. How much does clustering change when:

1. Changing some **hyperparameters** (distance metric, linkage, K, ...)
2. Changing some **observations** (bootstrapping, Hennig, 2007)
3. Changing some **features**

Check if observations are classified into same cluster across choices

e.g. using Jaccard index, Rand index

3. Clusterwise stability in R




```
library(fpc)
data(iris)
clusterboot(iris[, 1:4],
            clustermethod = hclustCBI,
            method = "complete", k = 3)
```

```
Clusterwise Jaccard bootstrap (omitting
multiple points) mean:
[1] 0.891 0.459 0.719
```

4. Internal validation indices

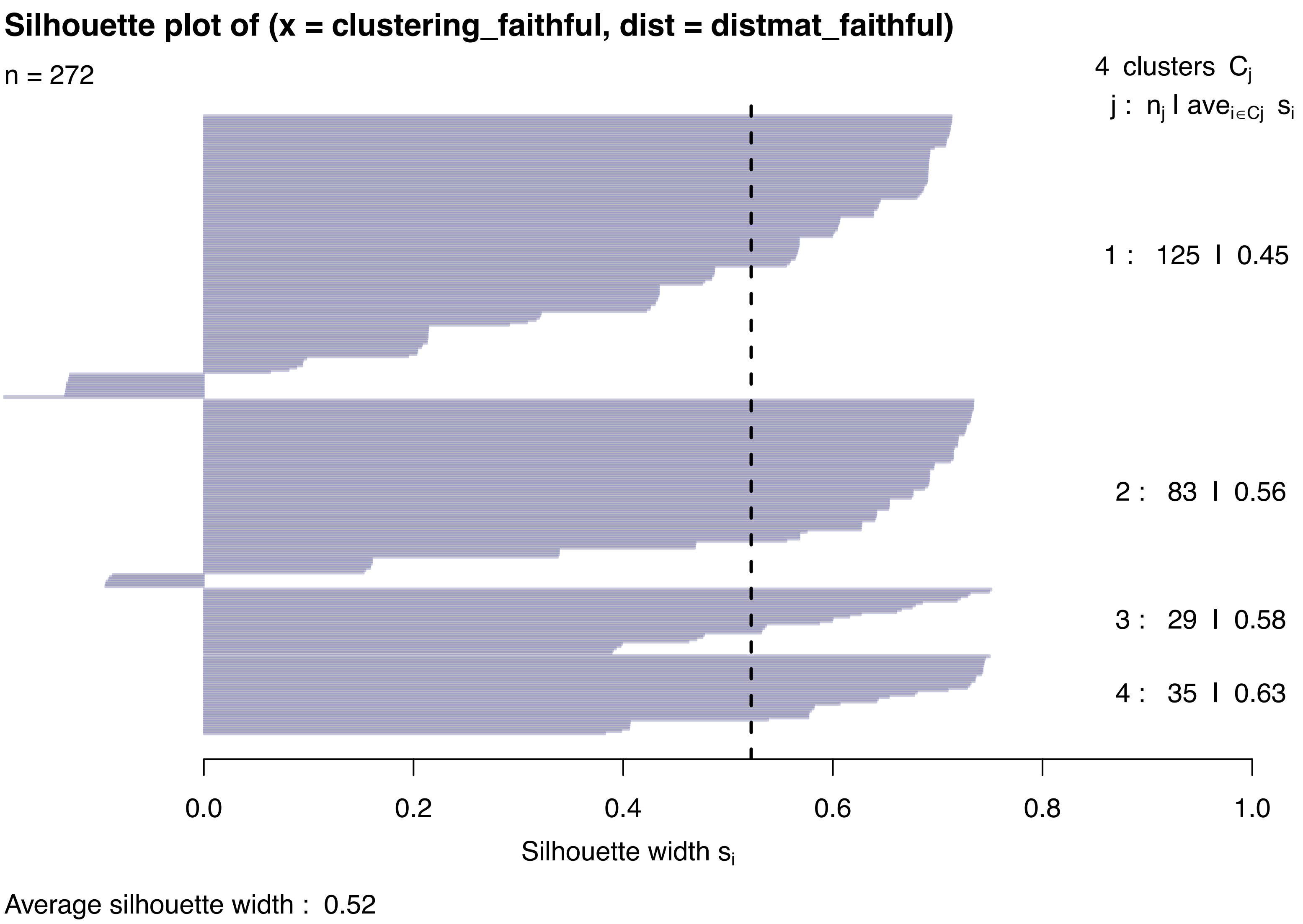
- Only look at "unsupervised" bit: data and clustering
- Quantify how "successful" the clustering is in some sense
- Popular **measures**:
  - Average sihouette width (**ASW**) (*how close are points to other clusters*)
  - "Gap statistic" *(Tibshirani et al. 2001)*
  - (measures from model-based clustering → tomorrow)

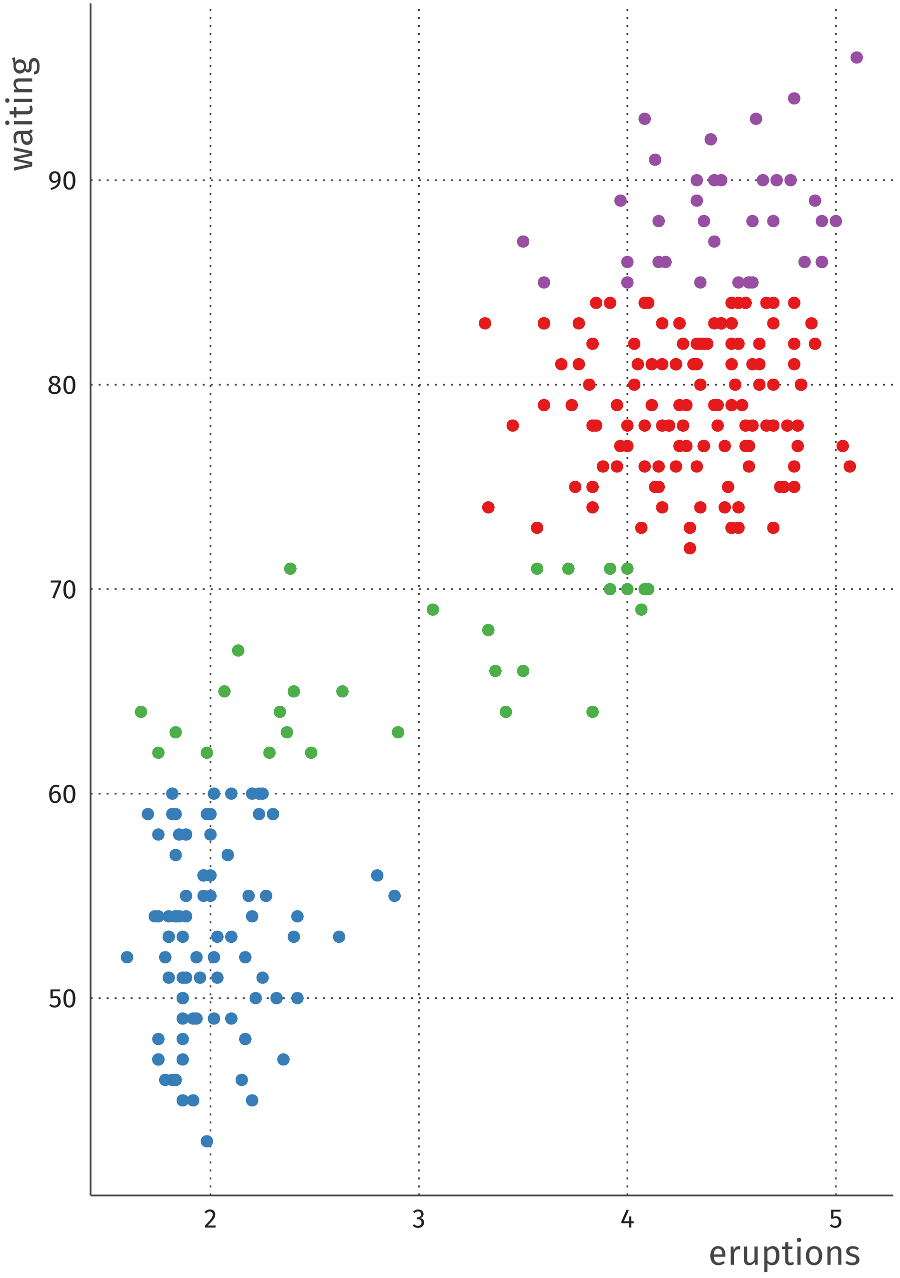
Disadvantage: don't take account of the clustering **aim**!

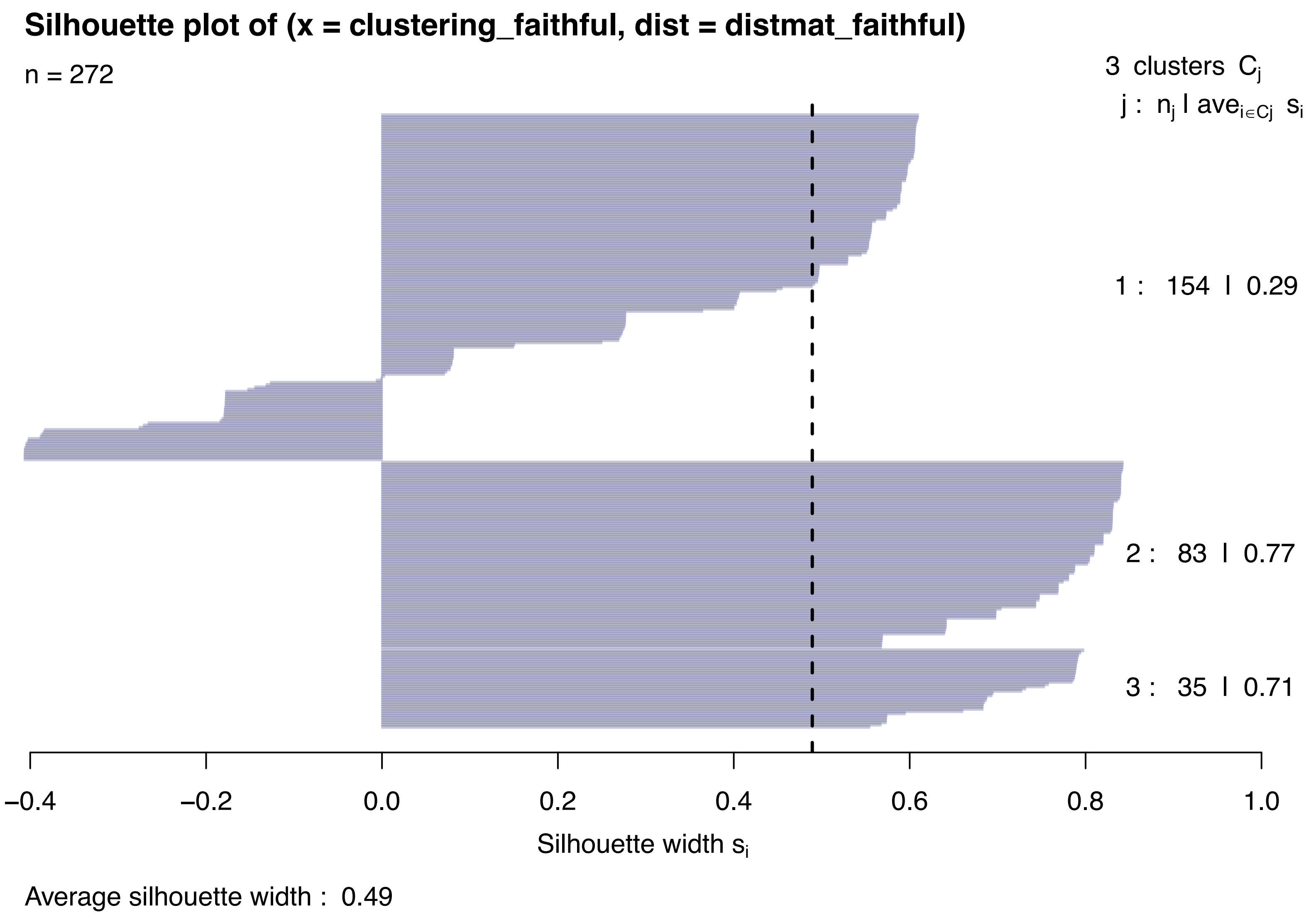
Silhouette analysis in R

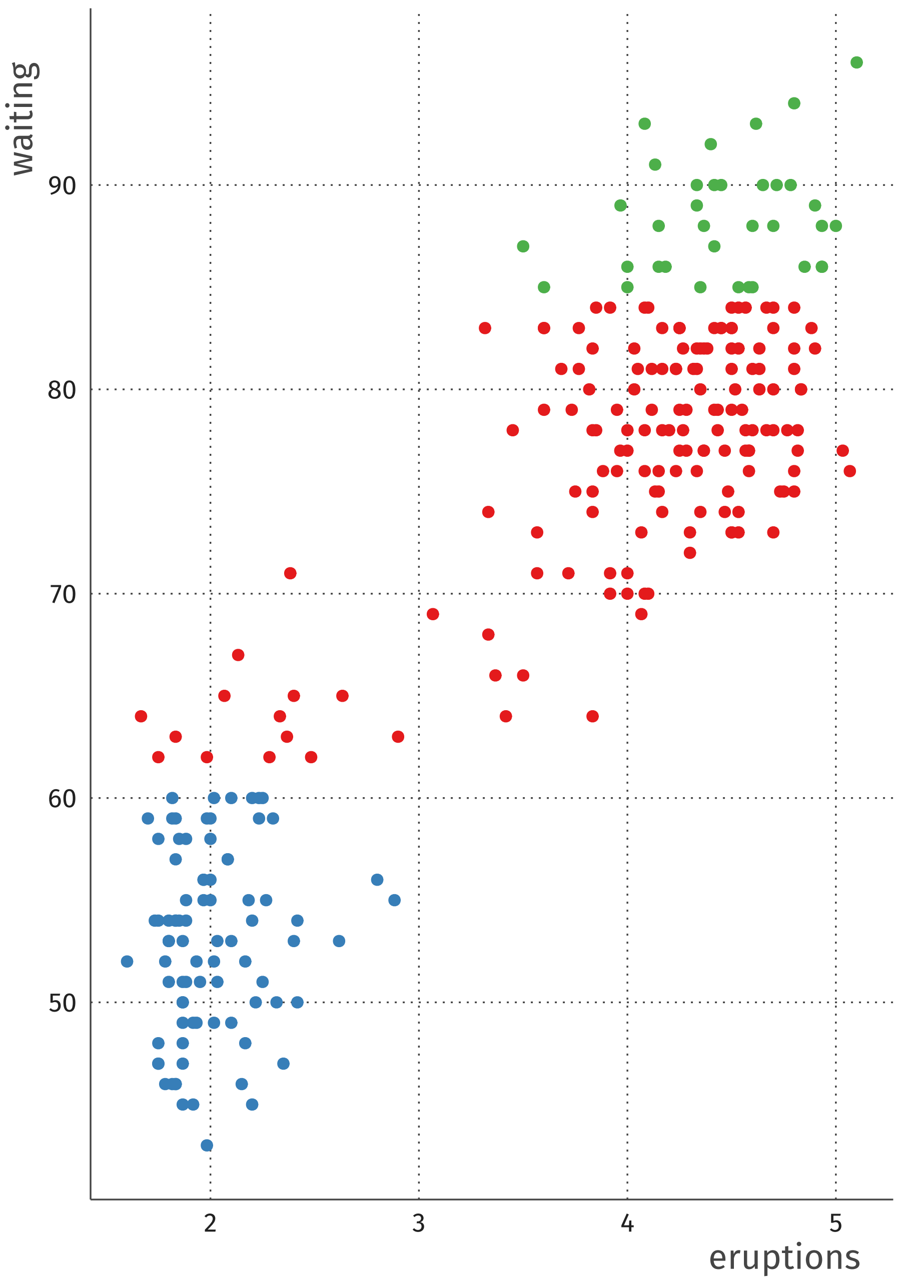
```
distmat_faithful <- dist(faithful)
hclust_faithful <- hclust(distmat_faithful)

clustering_faithful <- cutree(hclust_faithful, 2)
silhouette_scores <- silhouette(clustering_faithful, distmat_faithful)

plot(silhouette_scores)
```

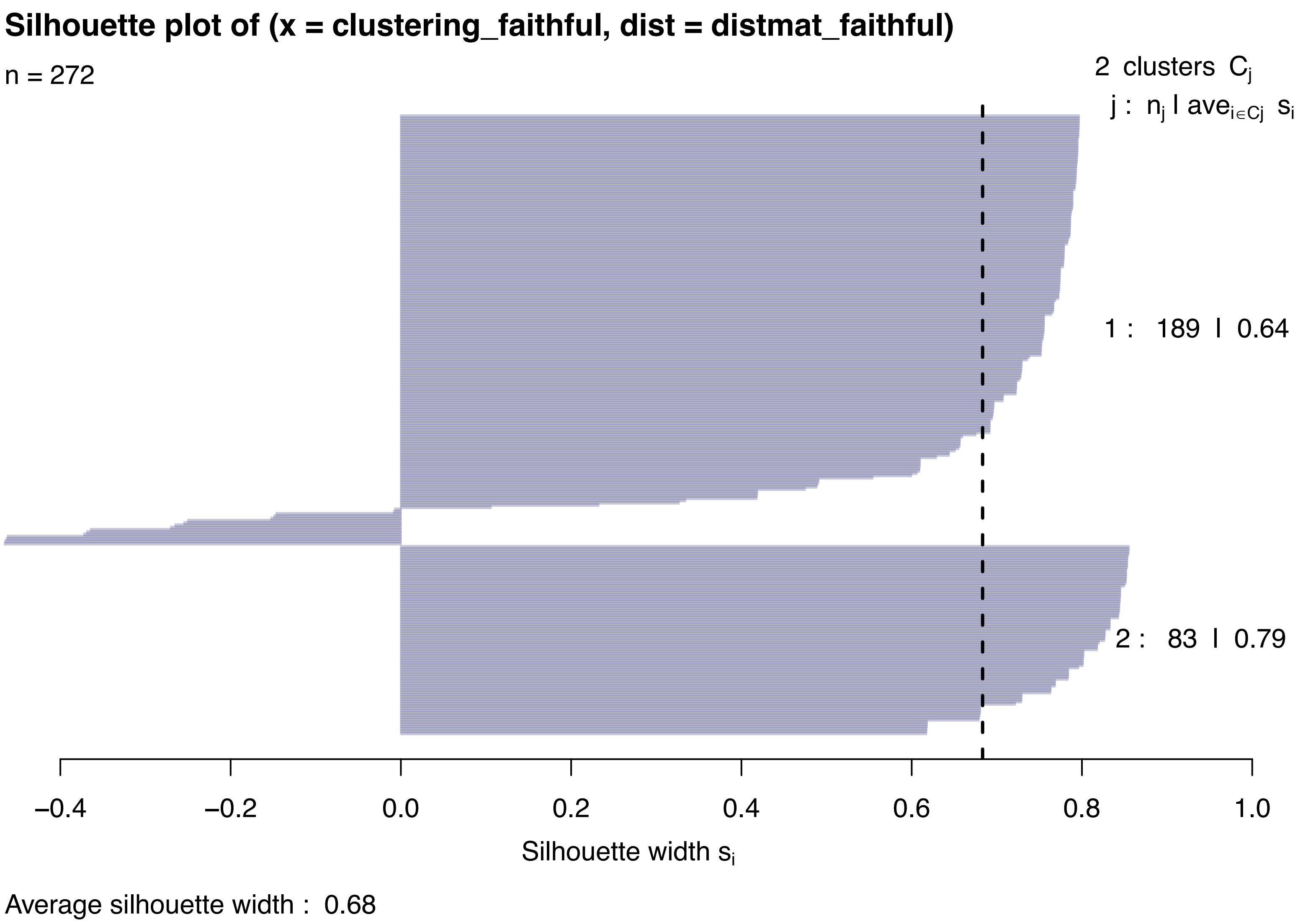
Note that this works for any type of clustering!


Silhouette plot of (x = clustering_faithful, dist = distmat_faithful)
n = 272
4 clusters C_j
j : n_j | $ave_{i \in Cj}$ s_i
1 : 125 | 0.45
2 : 83 | 0.56
3 : 29 | 0.58
4 : 35 | 0.63
0.0
0.2
0.4
0.6
0.8
1.0
Silhouette width s_i
Average silhouette width : 0.52

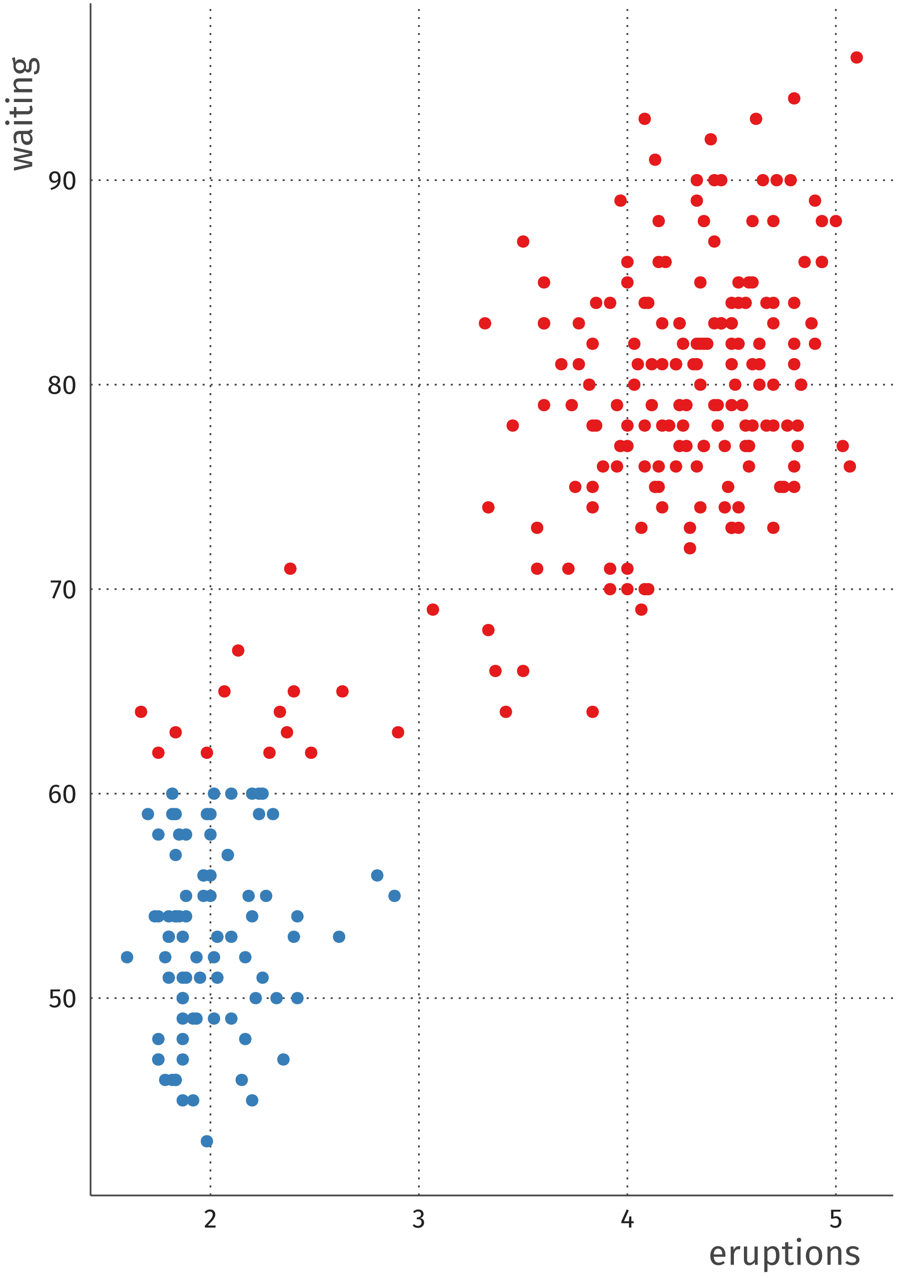

waiting
90
80
70
60
50
2
3
4
5
eruptions


Silhouette plot of (x = clustering_faithful, dist = distmat_faithful)
n = 272
3 clusters C_j
j : n_j | $ave_{i \in Cj}$ s_i
1 : 154 | 0.29
2 : 83 | 0.77
3 : 35 | 0.71
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
1.0
Silhouette width s_i
Average silhouette width : 0.49


waiting
90
80
70
60
50
2
3
4
5
eruptions



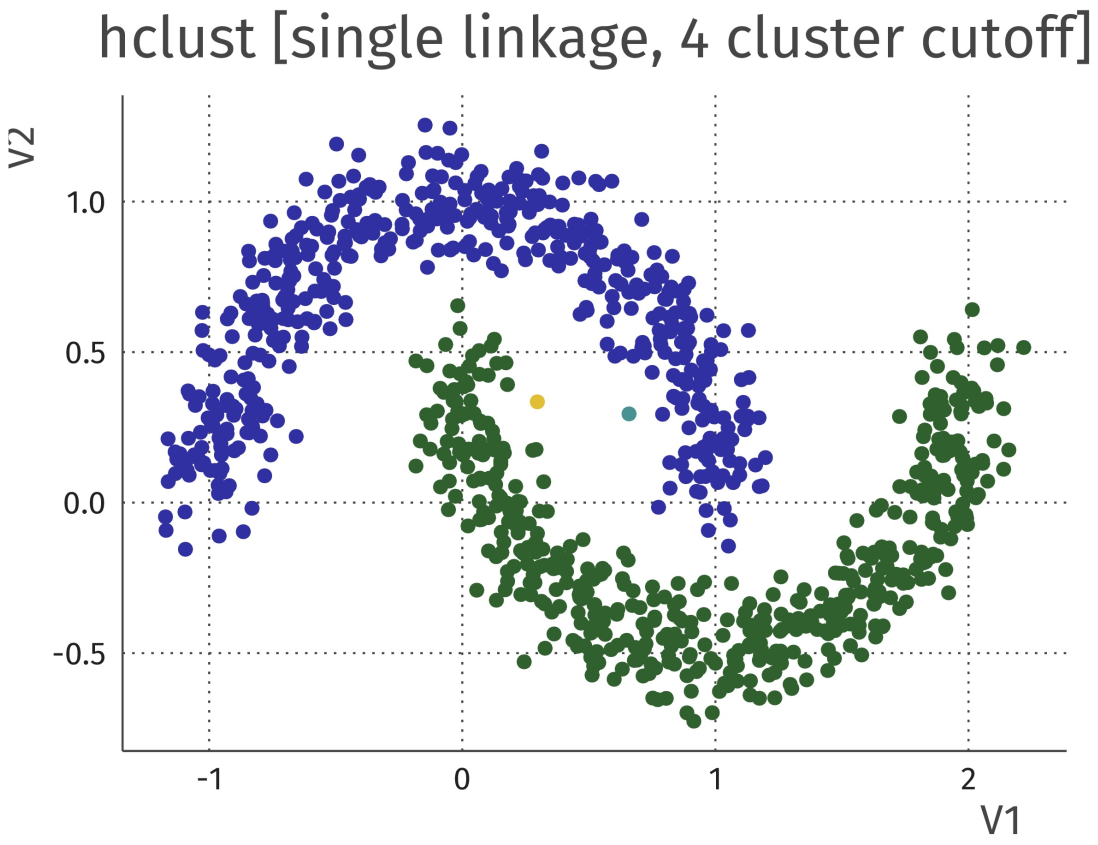
Silhouette plot of (x = clustering_faithful, dist = distmat_faithful)
n = 272
2 clusters C_j
j : n_j | $\mathrm{ave}_{i \in C_j}$ s_i
1 : 189 | 0.64
2 : 83 | 0.79
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
1.0
Silhouette width s_i
Average silhouette width : 0.68



waiting
90
80
70
60
50
2
3
4
5
eruptions

What do you think the average silhouette width (ASW) of this solution will approximately be?

<hidden solution>



https://www.menti.com/uf9b1miv22

Conclusion: clustering

- Clustering looks for “similar” groups of observations
- T*wo* basic clustering methods:
  1. **Hierarchical** clustering (e.g. bottom-up agglomerative, top-down divisive, …)
  2. **Partitional** clustering (e.g. k-means, DBSCAN, …).
- Cluster evaluation is an important and subtle topic;
- No way to get rid of critical thought.