

Hierarchical clustering

One tree, every number of clusters

Hierarchical clustering

1 Trees
2 Build a tree
3 Linkage
4 Distance
5 Real data
6 Which method






Monday vs today

k-means

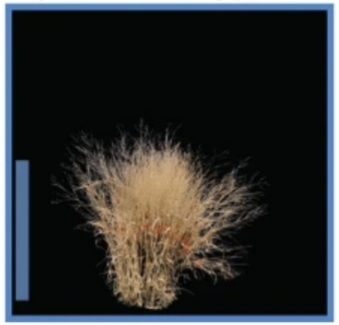
choose K first
→ K groups

hierarchical

no K needed
→ a tree: cut it for any K

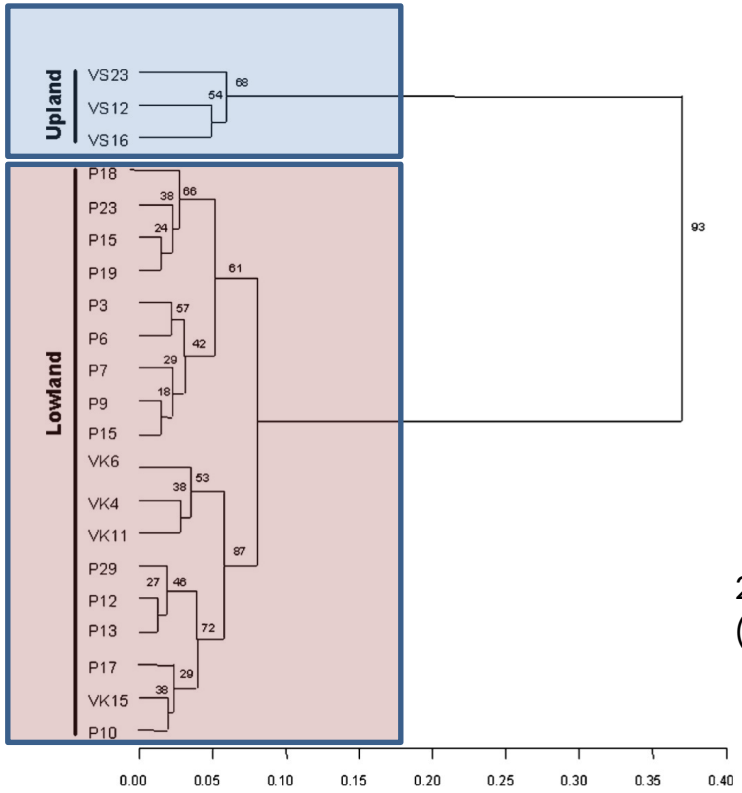
Same goal: **group samples that are close together**. Different route.

Upland ecotype



Lowland ecotype



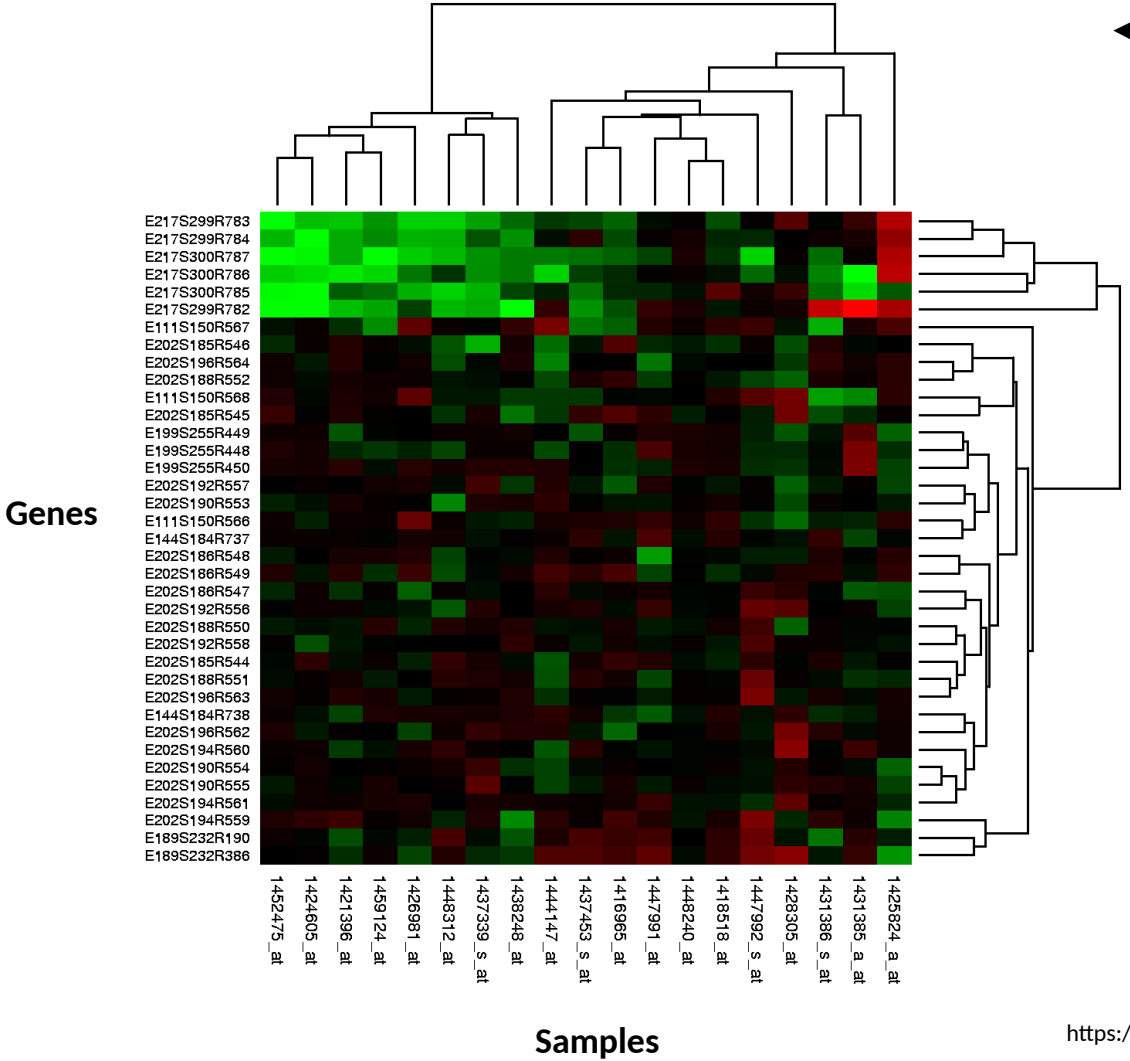


Hierarchical clustering

Data: 85 genetic markers

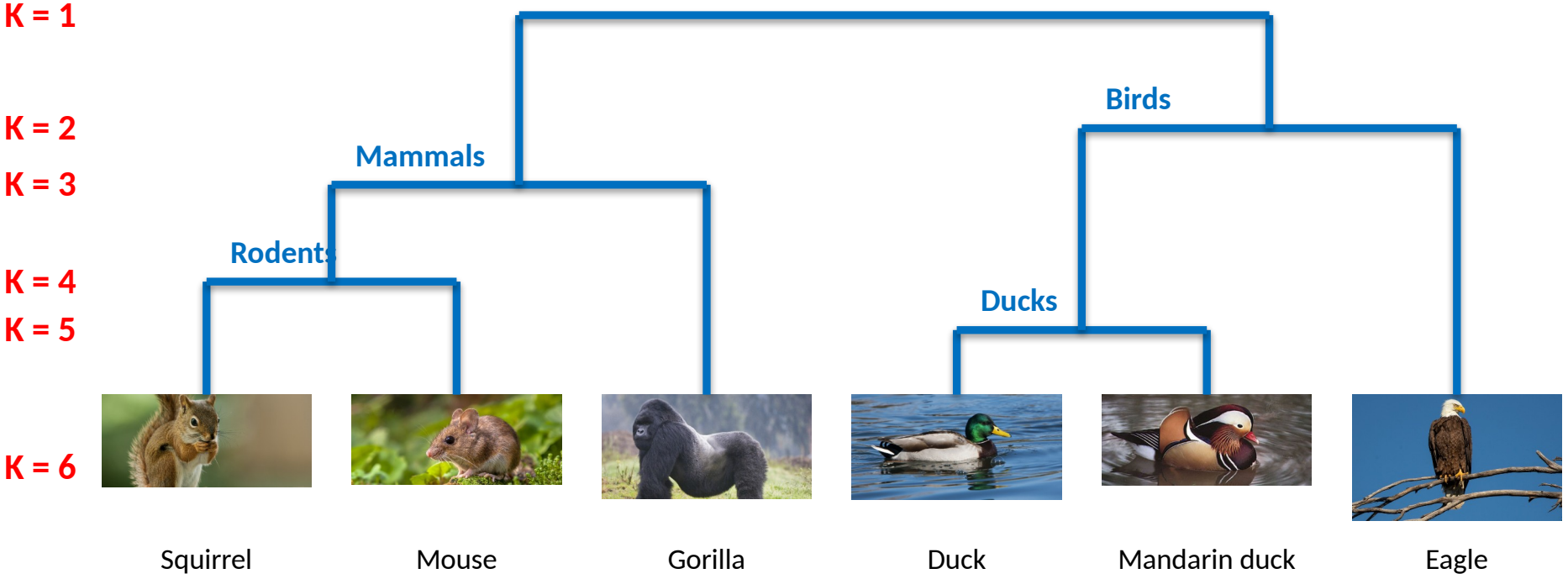
22 plants
(observations)

https://link.springer.com/article/10.1007/s10722-005-3878-9



https://en.wikipedia.org/wiki/Heat_map#/media/File:Heatmap.png

A tree of groups within groups



1 Trees

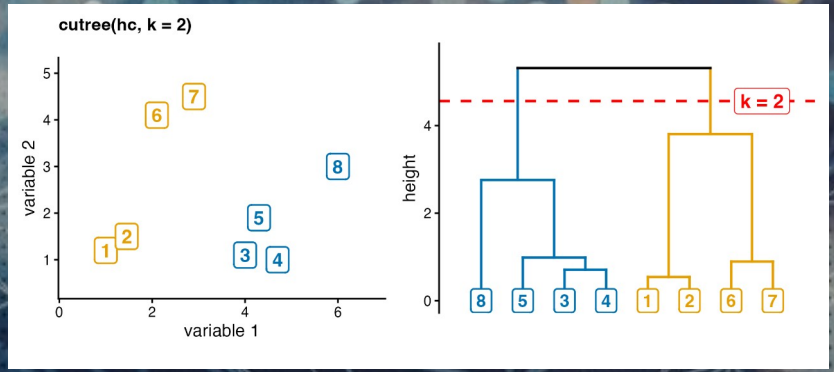
2 **Build a tree**

3 Linkage

4 Distance

5 Real data

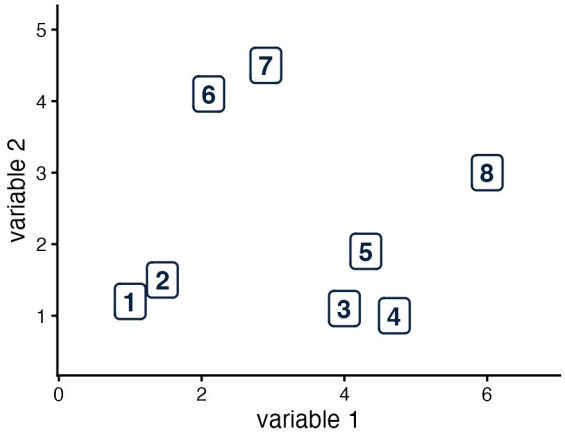
6 Which method



Live poll

Multiple choice

8 points. Which two merge first?



A 1 and 2

B 3 and 4

C 6 and 7

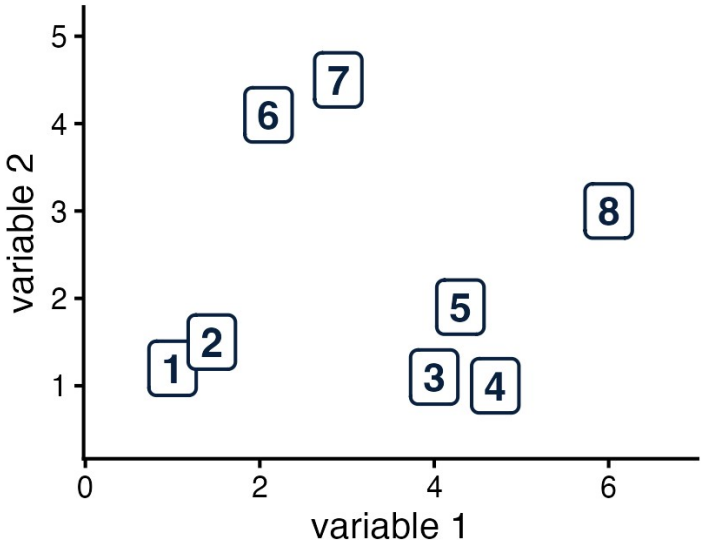
D 5 and 8

Scan with your phone

Join on your phone or laptop: greymonroe.github.io/live-poll-test/submit.html?poll=pls206-w8w-merge-first

host screen

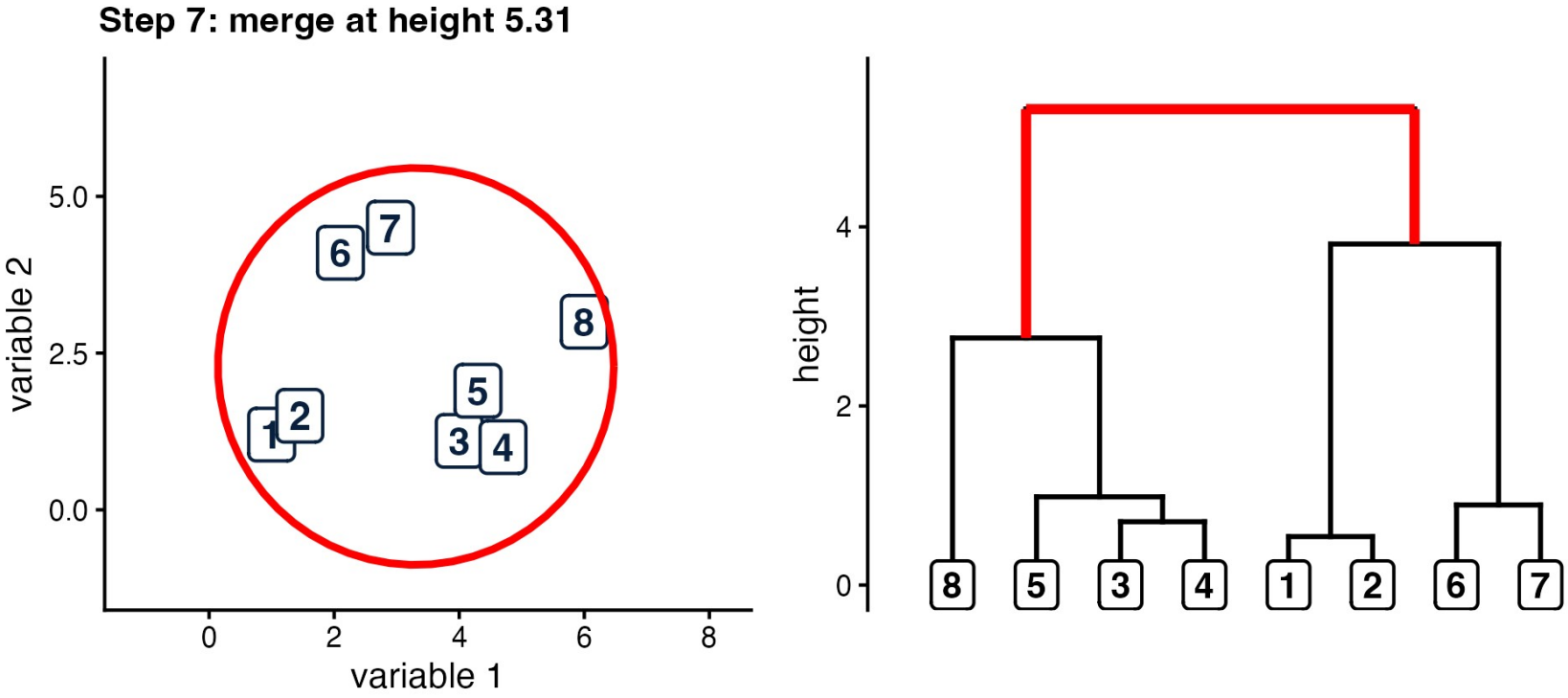
Step 1: distance between every pair



	1	2	3	4	5	6	7
2	0.54						
3	3.00	2.58					
4	3.71	3.29	0.71				
5	3.37	2.88	0.85	0.98			
6	3.10	2.68	3.55	4.05	3.11		
7	3.81	3.33	3.57	3.94	2.95	0.89	
8	5.31	4.79	2.76	2.39	2.02	4.05	3.44

dist() gives every pair. **Smallest: 1 and 2, 0.54** → they merge first.

Merge the closest two, repeat

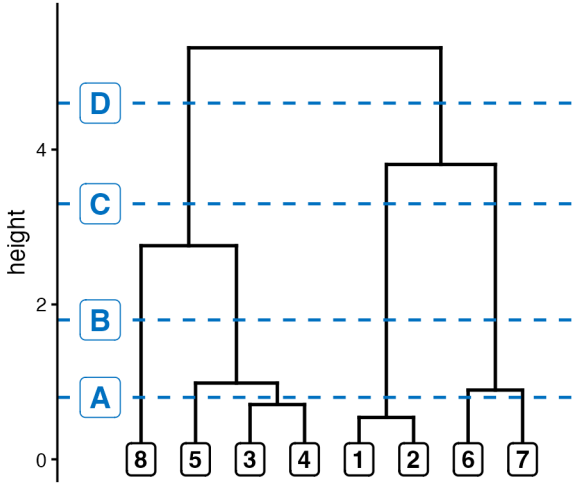


Height of each join = **the distance when they merged**

Live poll

Multiple choice

Where would you cut this tree to get 3 clusters?



A

B

C

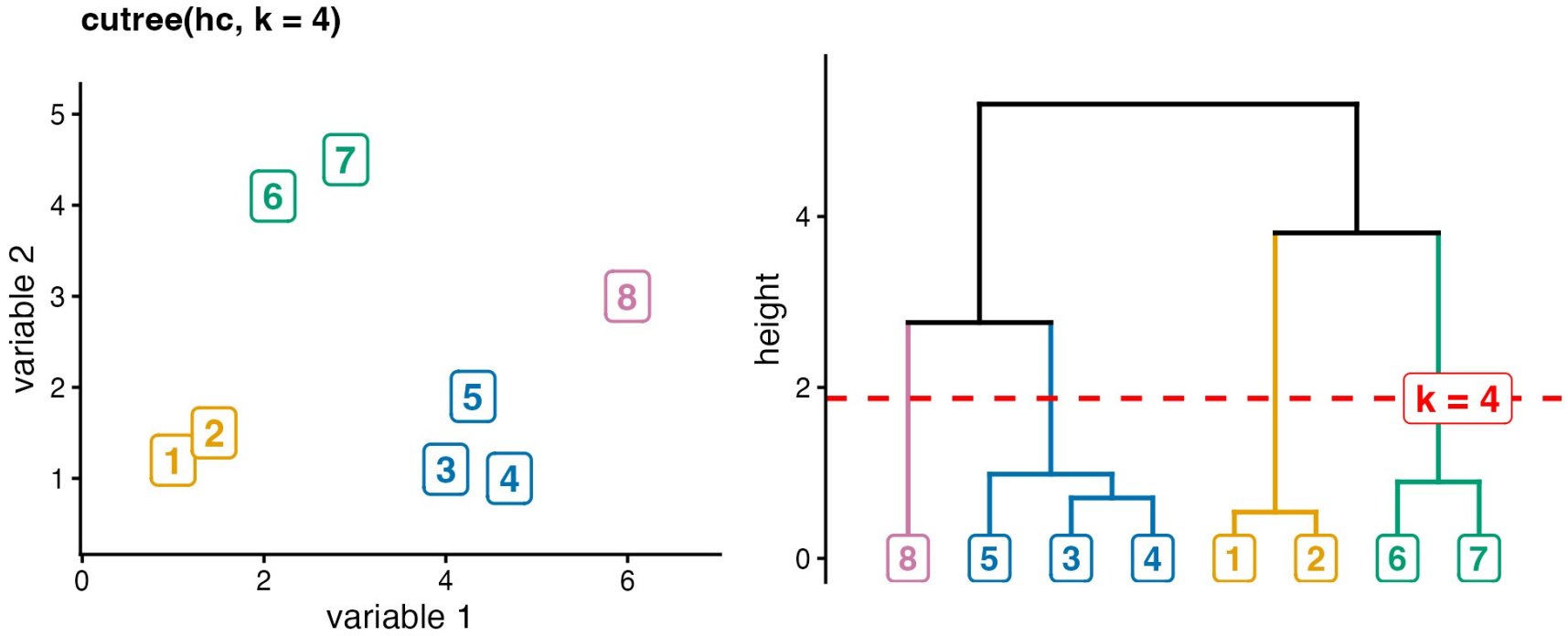
D

Scan with your phone

[Join on your phone or laptop: greymonroe.github.io/live-poll-test/submit.html?poll=pls206-w8w-cut3](greymonroe.github.io/live-poll-test/submit.html?poll=pls206-w8w-cut3)

[host screen](greymonroe.github.io/live-poll-test/host.html?poll=pls206-w8w-cut3)

cutree(): cut at any K



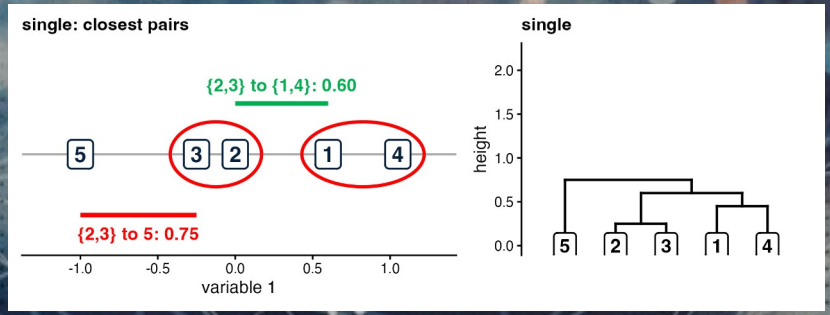
One tree, **every K**. No re-fitting, unlike kmeans().

In R: three lines

```
> d <- dist(toy)
> hc <- hclust(d)
> round(hc$height, 2)
[1] 0.54 0.71 0.89 0.98 2.76 3.81 5.31  ← first and last merge
> cutree(hc, k = 3)
1 2 3 4 5 6 7 8
1 1 2 2 2 3 3 2  ← cluster of each point
```

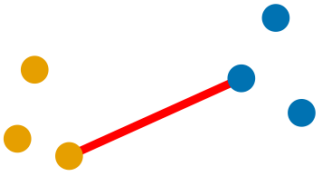
dist() → **hclust()** → plot() and cutree()

1 Trees
2 Build a tree
3 Linkage
4 Distance
5 Real data
6 Which method

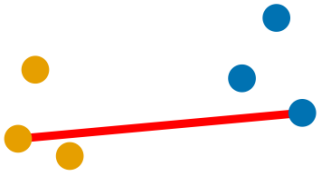


How far apart are two clusters?

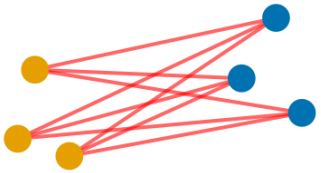
single: closest
2.19



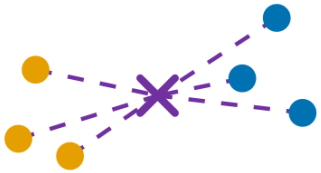
complete: farthest
3.31



average: all 9 pairs
2.84

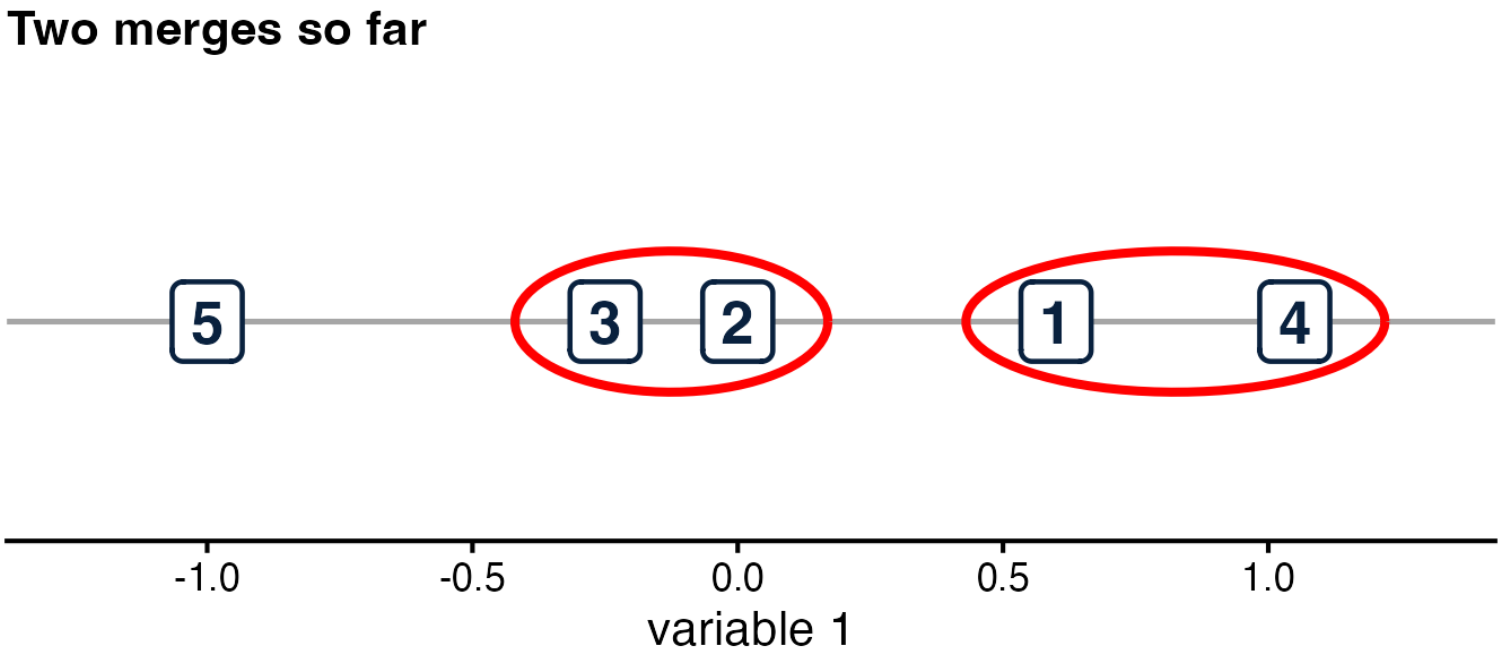


Ward: added
spread



single closest pair · **complete** farthest pair (R's default) · **average** · **Ward**

Five points on a line

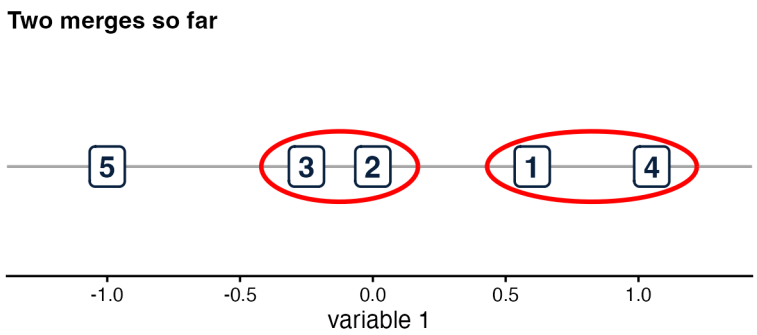


{2,3} and {1,4} have merged. **Complete linkage: what joins next?**

Live poll

Multiple choice

Complete linkage. What merges next?



A {2,3} + {1,4}

B {2,3} + 5

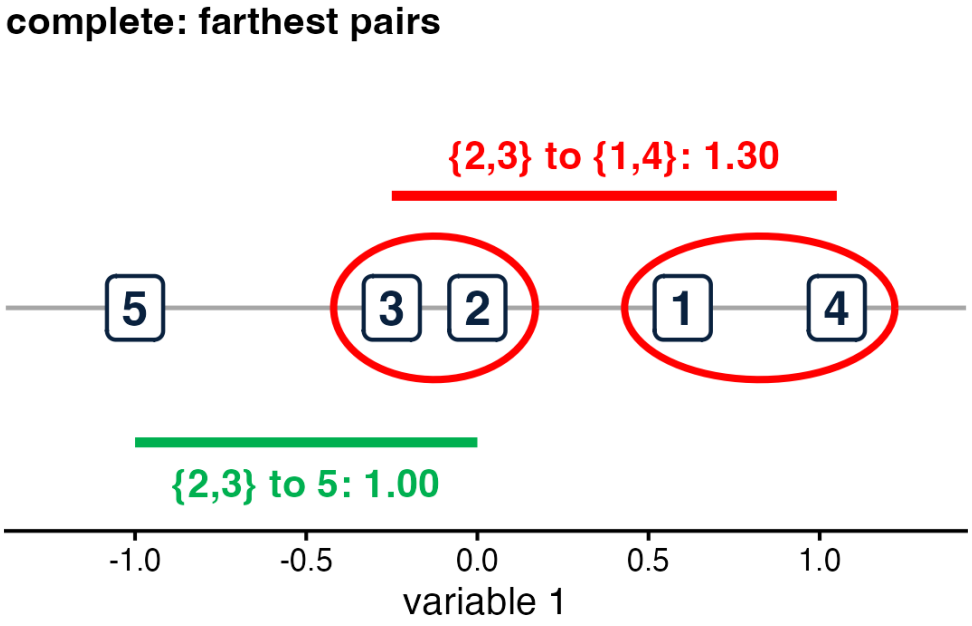
C {1,4} + 5

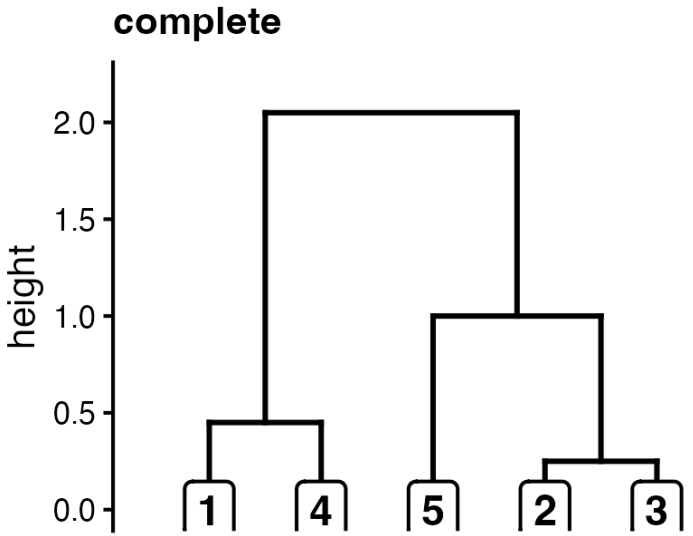
Scan with your phone

[Join on your phone or laptop: greymonroe.github.io/live-poll-test/submit.html?poll=pls206-w8w-complete-next](greymonroe.github.io/live-poll-test/submit.html?poll=pls206-w8w-complete-next)

[host screen]

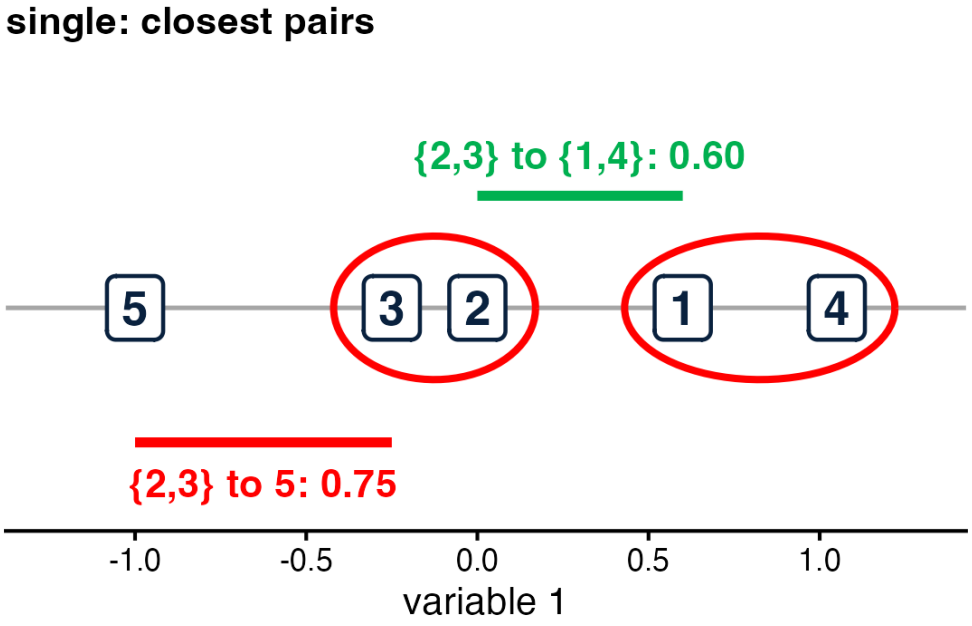
Complete: compares the farthest points

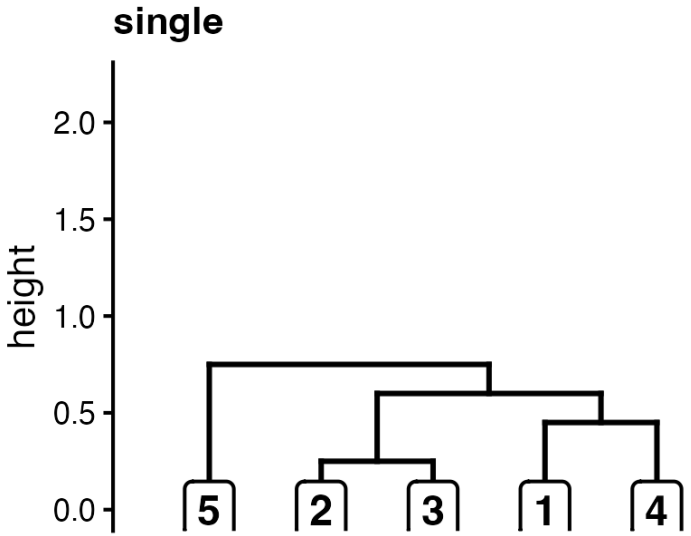




Farthest pairs: **1.00 < 1.30**, so **5 joins {2,3}**, even though 2 is near 1

Single: compares the closest points





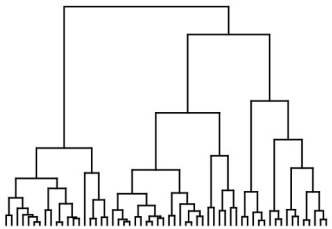
Closest pairs: **0.60 < 0.75**, so **{2,3} joins {1,4}**. Same data, a different tree.

Live poll

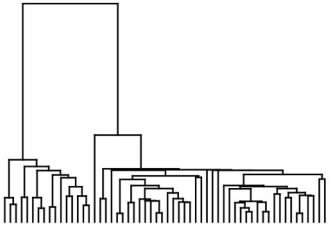
Multiple choice

Same 58 points, two linkages. Which tree is single linkage?

A



B



A left

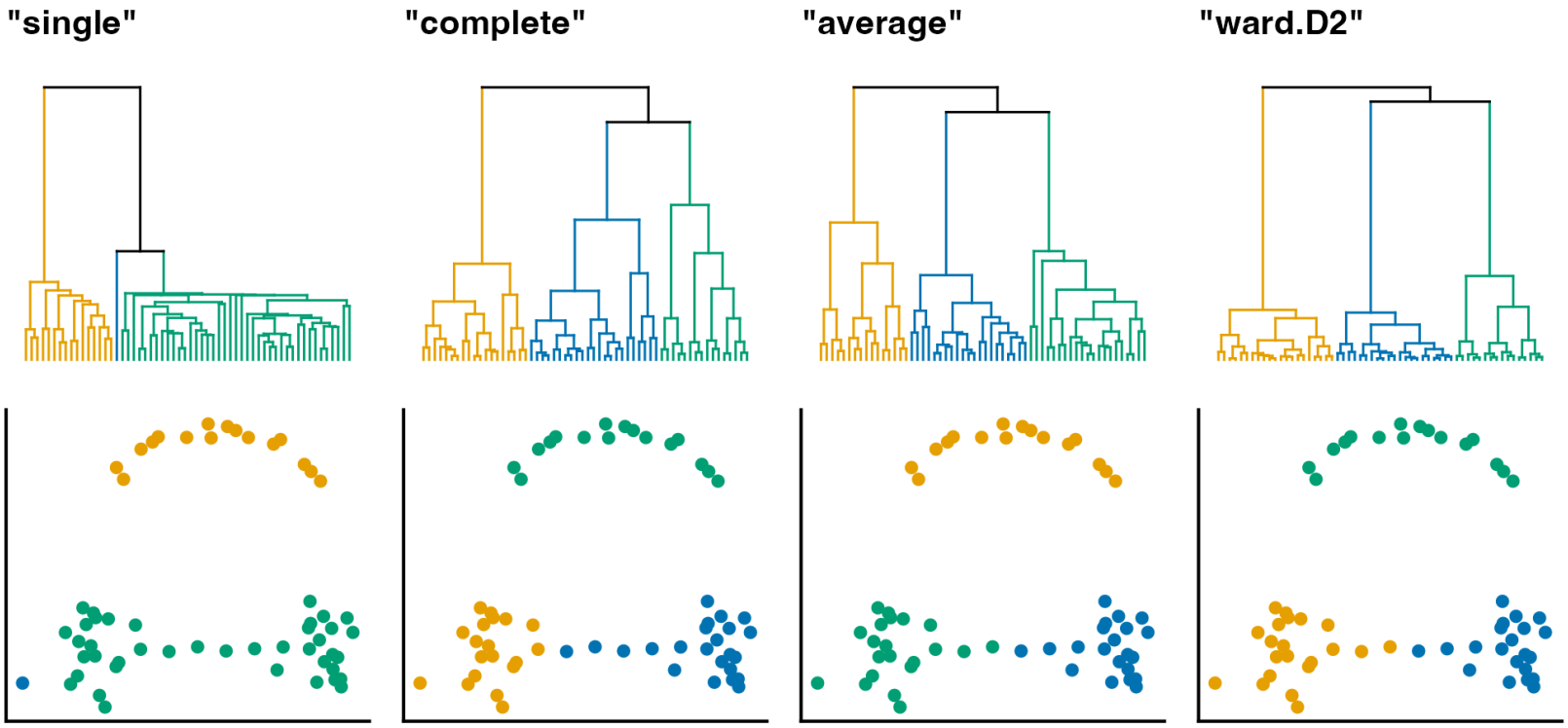
B right

Scan with your phone

[Join on your phone or laptop: greymonroe.github.io/live-poll-test/submit.html?poll=pls206-w8w-single-tree](greymonroe.github.io/live-poll-test/submit.html?poll=pls206-w8w-single-tree)

host screen

Same 58 points, four linkages



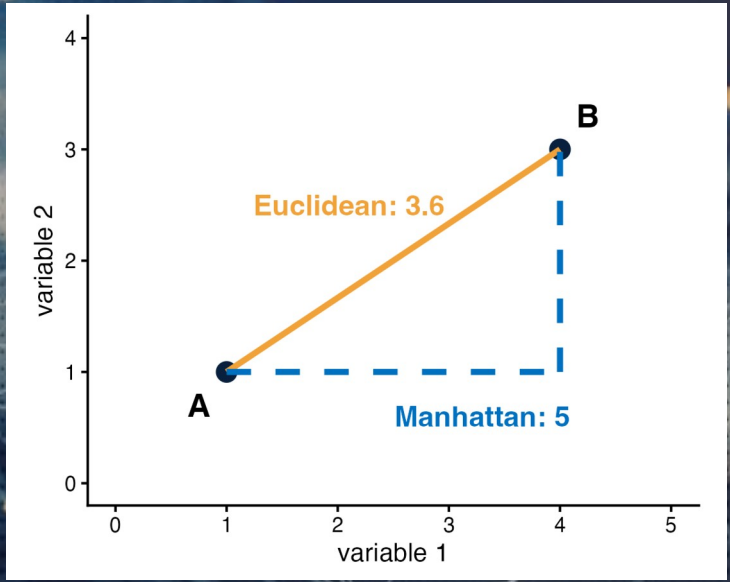
single follows chains (and isolates outliers); **complete, average, Ward** make compact groups

Which linkage?

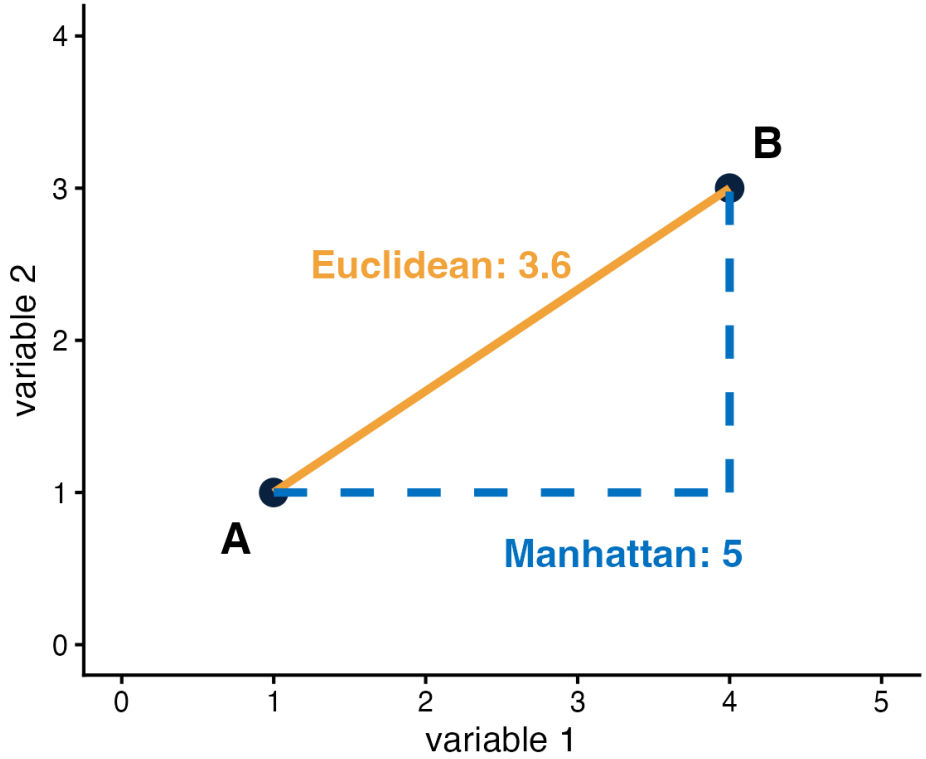
`complete`	tight groups; R's default
`Ward.D2`	compact, even groups: like k-means
`average`	in between
`single`	long chains, odd shapes; one outlier can be a cluster

hclust(d, method = **"ward.D2"**) Try two; trust groups that survive both.

1 Trees
2 Build a tree
3 Linkage
4 **Distance**
5 Real data
6 Which method

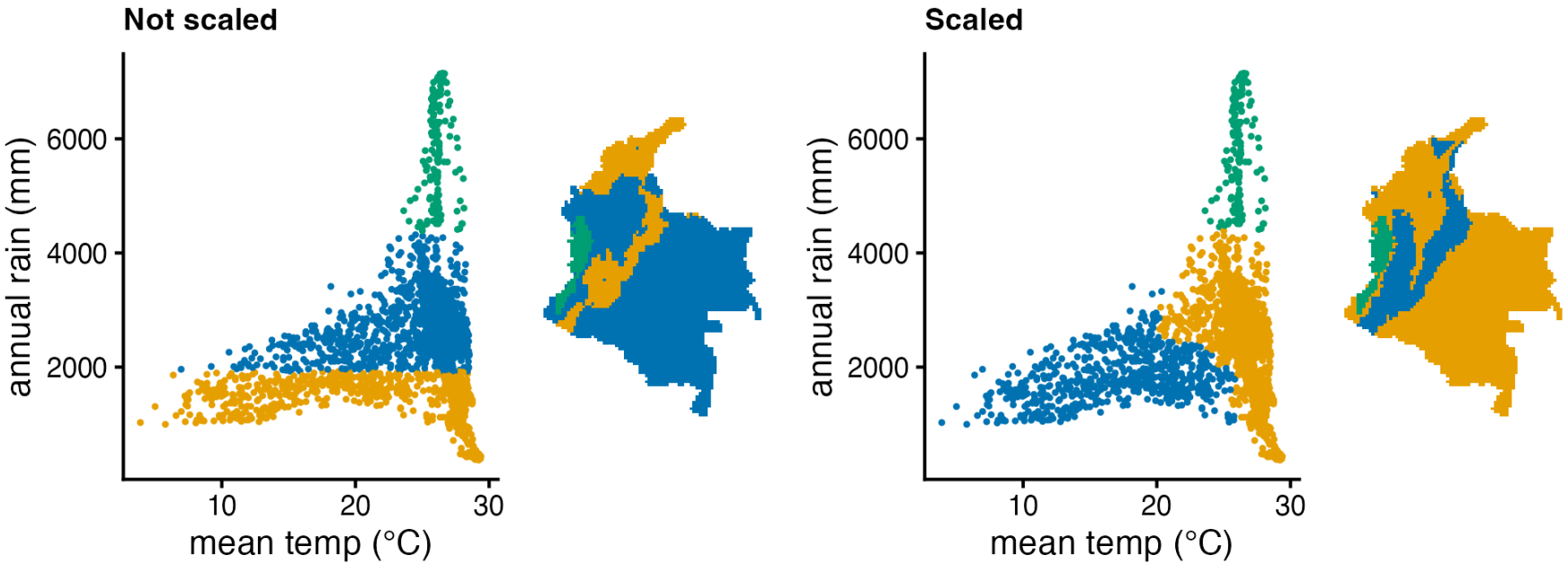


Straight line or city blocks



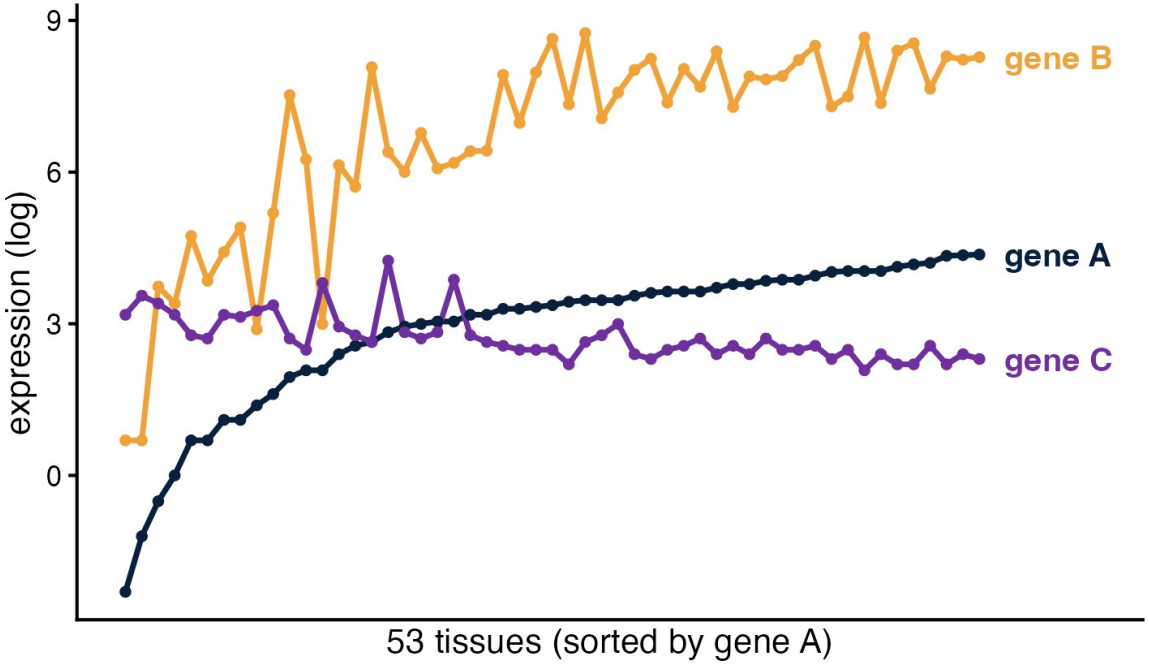
dist(x) is **Euclidean** by default; **method = "manhattan"** for city blocks

Scale first, or the units win



Rain SD **931 mm**, temperature SD **4.3 °C** → **dist(scale(x))**

Same pattern, different level

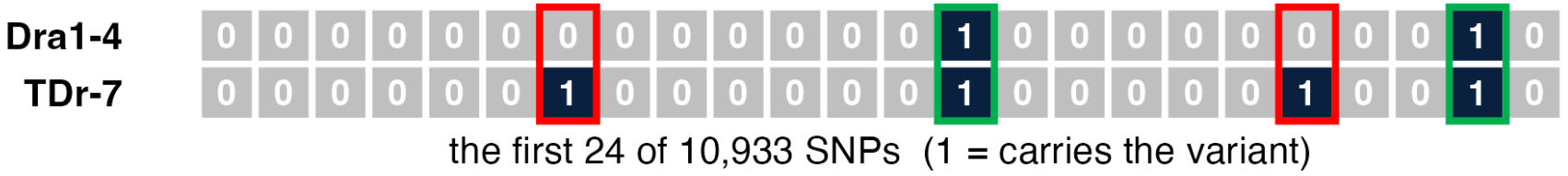


Euclidean:
A–B 28.5
A–C 13.0 ← closer

1 – correlation:
A–B 0.09 ← closer
A–C 1.61

Gene expression: cluster on **shape** → as.dist(1 – cor(t(x)))

0/1 data: binary distance



$$d = \frac{\text{markers that differ}}{\text{markers where either has a 1}} = \frac{1{,}640}{2{,}739} = 0.60$$

binary distance: shared 0s are ignored; most plants lack most variants

Pick the distance that matches 'similar'

continuous, same units	`dist(x)`
continuous, mixed units	`dist(scale(x))`
profiles: shape matters	`as.dist(1 - cor(t(x)))`
0/1 presence	`dist(x, "binary")`

1 Trees

2 Build a tree

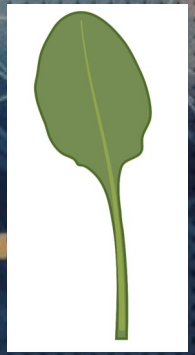
3 Linkage

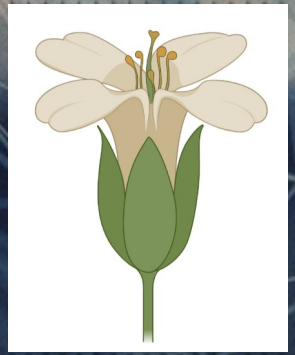
4 Distance

▶ **5 Real data**

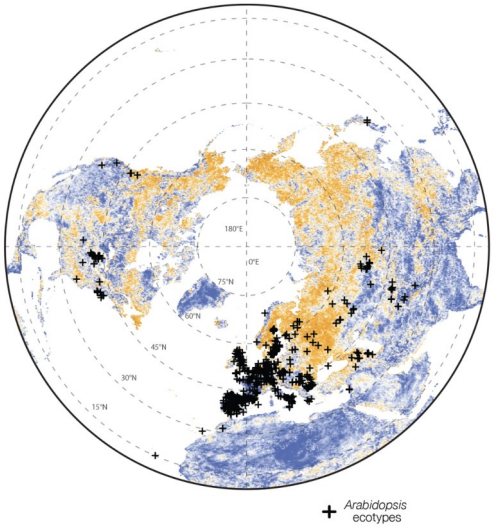
6 Which method











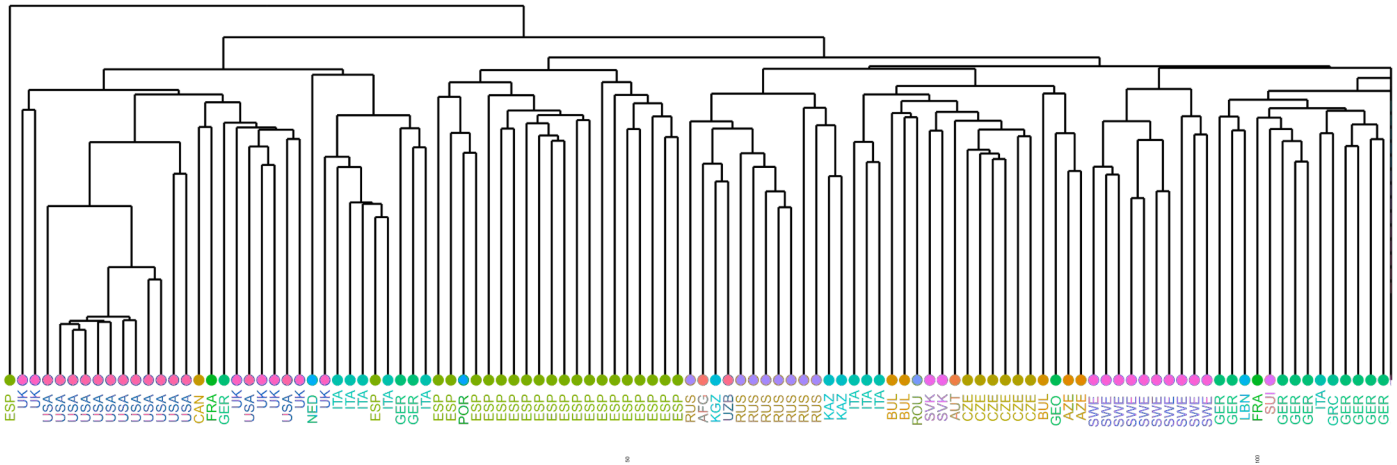


Population structure in *Arabidopsis thaliana*

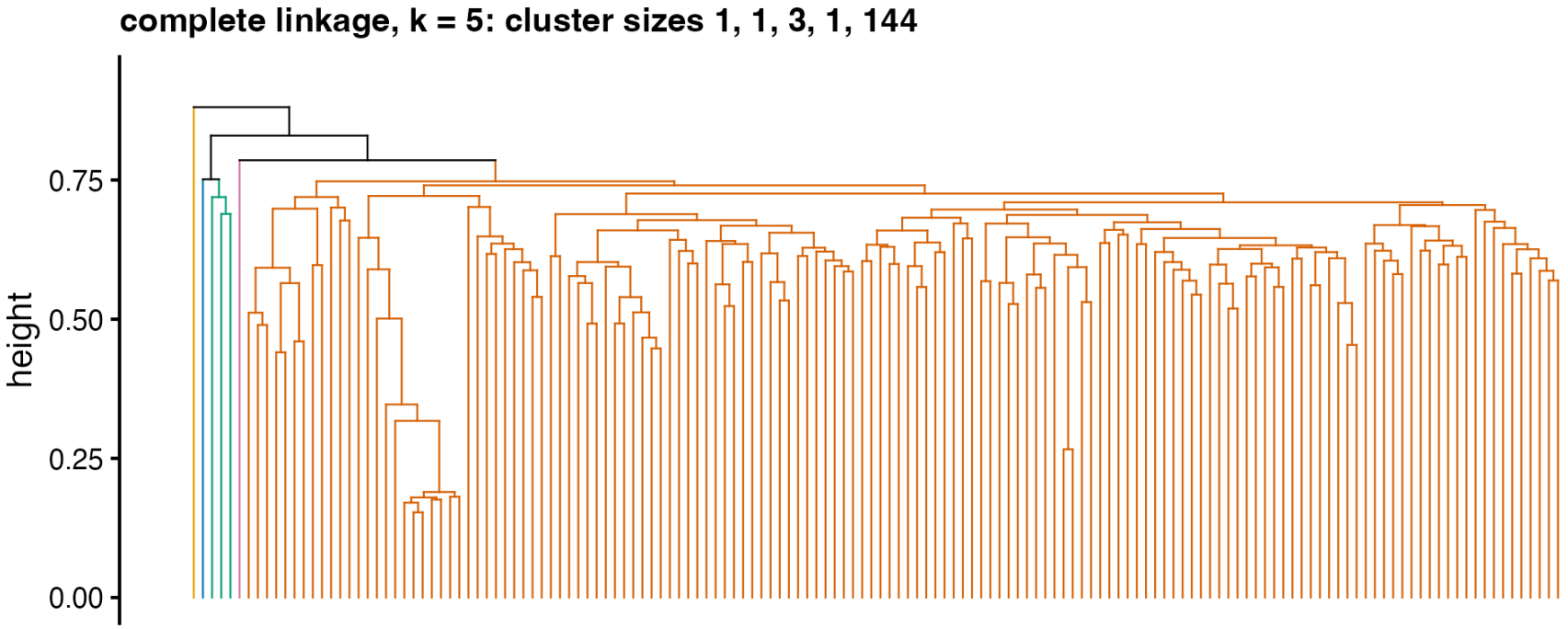
Data: >10,000 genetic markers

150 plants (observations)



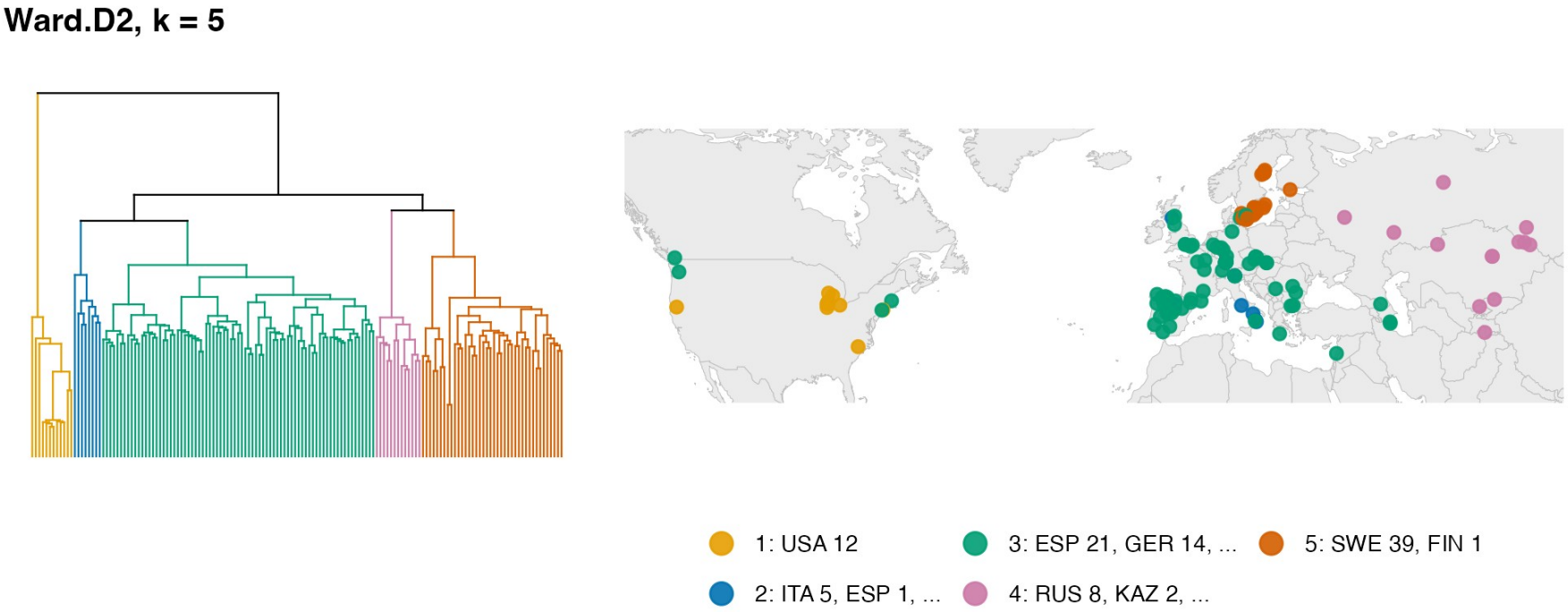

ESP UK UK USA USA USA

Complete linkage: one blob and a few stragglers



144 of 150 plants in one cluster: the most unusual plants rule every merge

Ward.D2: clusters follow geography



No location went into the clustering. **The map is a finding.**

Check clusters against what you know

```
> table(cluster, plants$Country)[, c("SWE", "RUS", "USA",
"ITA", "ESP", "GER")]

cluster SWE RUS USA ITA ESP GER
      1   0   0  12   0   0   0
      2   0   0   0   5   1   1
      3   1   0   3   4  21  14
      4   0   8   0   0   0   0
      5  39   0   0   0   0   0
```

table(cutree(...), metadata) **is how you read a clustering**

Article

Mass-spectrometry-based draft of the *Arabidopsis* proteome

https://doi.org/10.1038/s41586-020-2094-2
Received: 4 June 2019
Accepted: 17 January 2020
Published online: 11 March 2020
Check for updates

Julia Mergner[1], Martin Frejno[1], Markus List[2], Michael Papacek[3], Xia Chen[4], Ajeet Chaudhary[4], Patroklos Samaras[1], Sandra Richter[5], Hiromasa Shikata[6,7], Maxim Messerer[8], Daniel Lang[8], Stefan Altmann[9], Philipp Cyprys[10], Daniel P. Zolg[1], Toby Mathieson[11], Marcus Bantscheff[11], Rashmi R. Hazarika[12,13], Tobias Schmidt[1], Corinna Dawid[14], Andreas Dunkel[14], Thomas Hofmann[14], Stefanie Sprunck[10], Pascal Falter-Braun[9,15], Frank Johannes[12,13], Klaus F. X. Mayer[8,16], Gerd Jürgens[5], Mathias Wilhelm[1], Jan Baumbach[2], Erwin Grill[3], Kay Schneitz[4], Claus Schwechheimer[6] & Bernhard Kuster[1,13,17]

Gene expression data

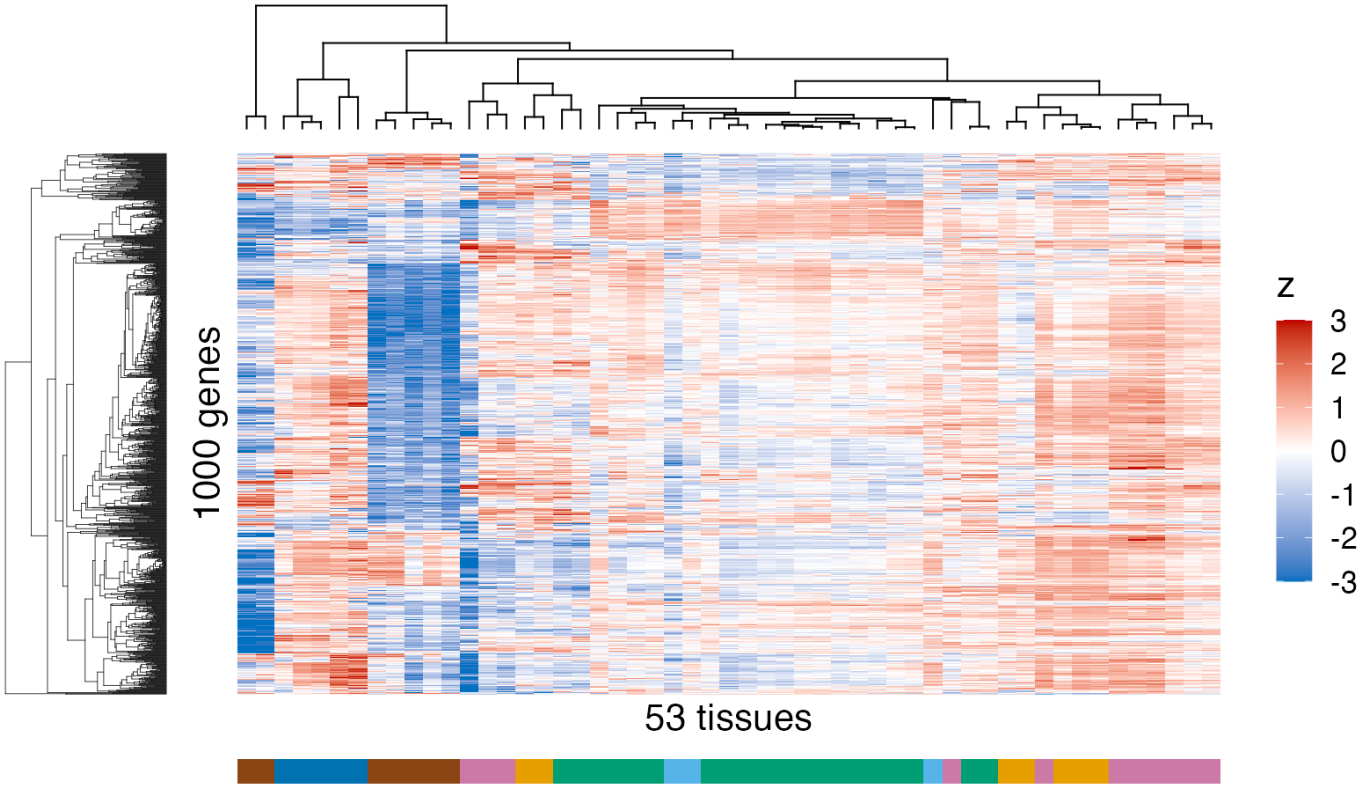
>50 tissues

1000 genes

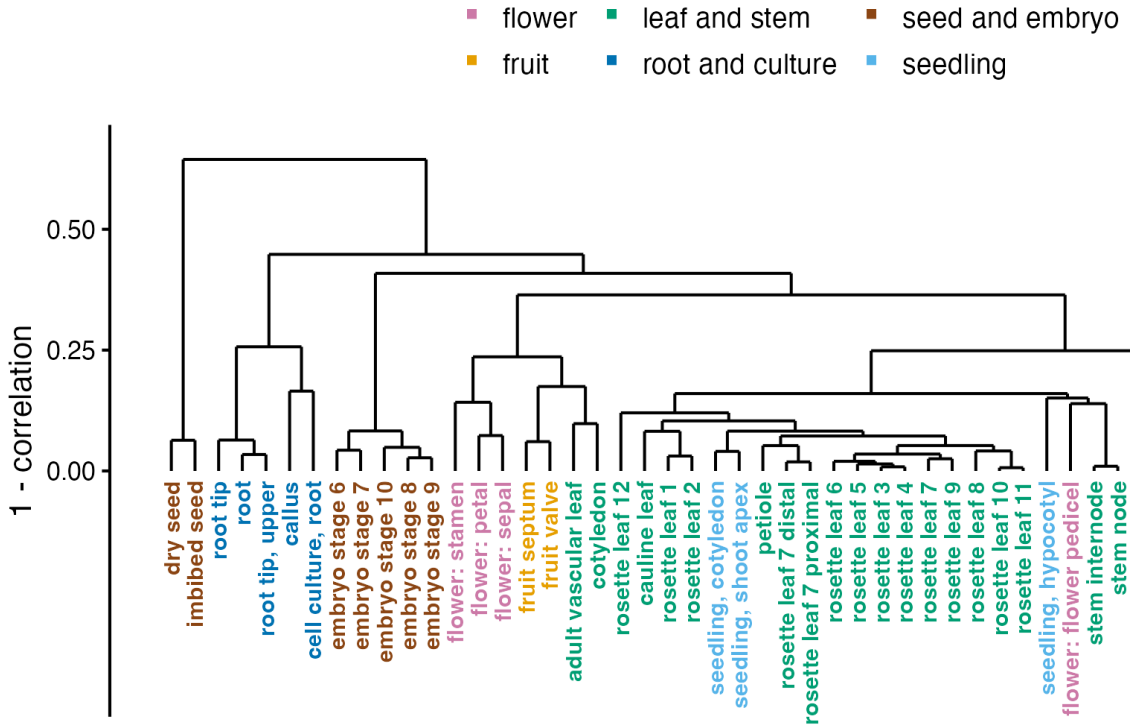
adult, adult vascular leaf	adult, cauline leaf	adult, cotyledon	adult, fruit septum	adult, fruit valve	adult, petiole, rosette leaf 7 petiole	adult, root	adult, rosette leaf, rosette leaf 1
3.8066625	3.4657359	3.8501476	3.3322045	3.0910425	3.8066625	3.5835189	3.7841896
2.6390573	3.1354942	2.9957323	2.3025851	2.4849066	2.8332133	2.9957323	3.3322045
2.8332133	3.1780538	3.1354942	3.0910425	2.9957323	2.9957323	3.4011974	3.0445224
2.7725887	1.9459101	1.7917595	1.9459101	2.6390573	1.0986123	0.6931472	1.6094379
3.0445224	3.4965076	2.9444390	3.5263605	3.5263605	2.9444390	3.2958369	3.3322045
4.7791235	4.8675345	4.3174881	4.3040651	4.1431347	4.3174881	4.5747110	4.3944492
3.7376696	3.6635616	3.4657359	3.5553481	4.0604430	3.6375862	3.7376696	3.3672958
3.6635616	3.6375862	3.5553481	3.7376696	3.7135721	3.2580965	3.4965076	3.7612001
5.8141305	5.4026774	4.9416424	5.3565863	6.1377271	3.8066625	2.8903718	5.3082677
3.6375862	5.4424177	3.9318256	4.3307333	5.1357984	5.5683445	2.9957323	5.4026774
0.0000000	1.3862944	0.0000000	0.6931472	0.6931472	1.0986123	0.6931472	1.0986123



Order rows and columns by their trees



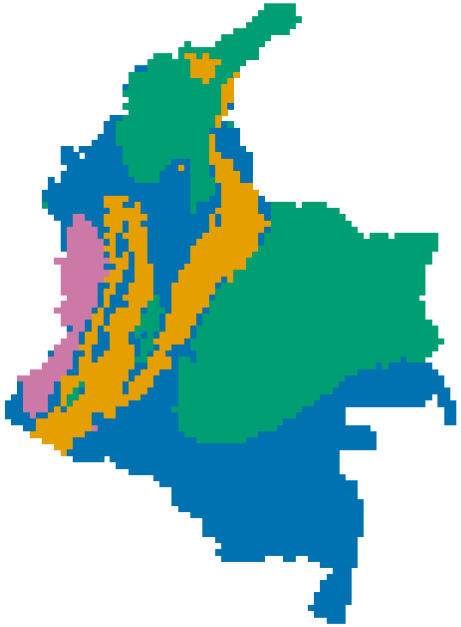
Tissues cluster by organ



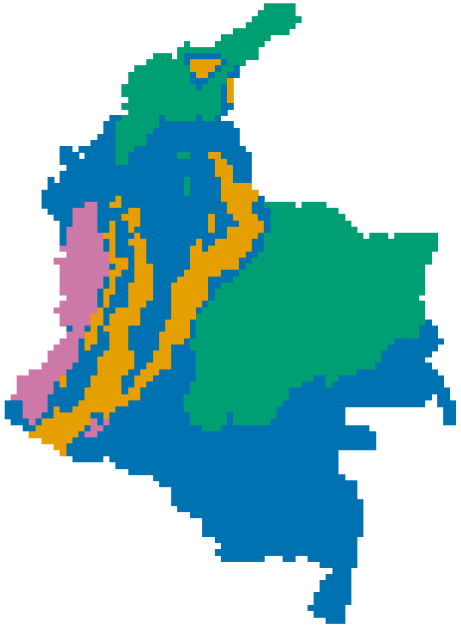
All five **root and culture** samples form one branch; the **embryo stages** group together

Colombia: k-means vs hierarchical

kmeans(centers = 4)



hclust(Ward.D2), cutree(k = 4)



```
> table(hclust_cl, kmeans_cl)
          kmeans_cl
hclust_cl    1    2    3    4
        1  402    1    0    0
        2  126 1397  288    0
        3    0   13 1150    0
        4    0   32    0  162
```

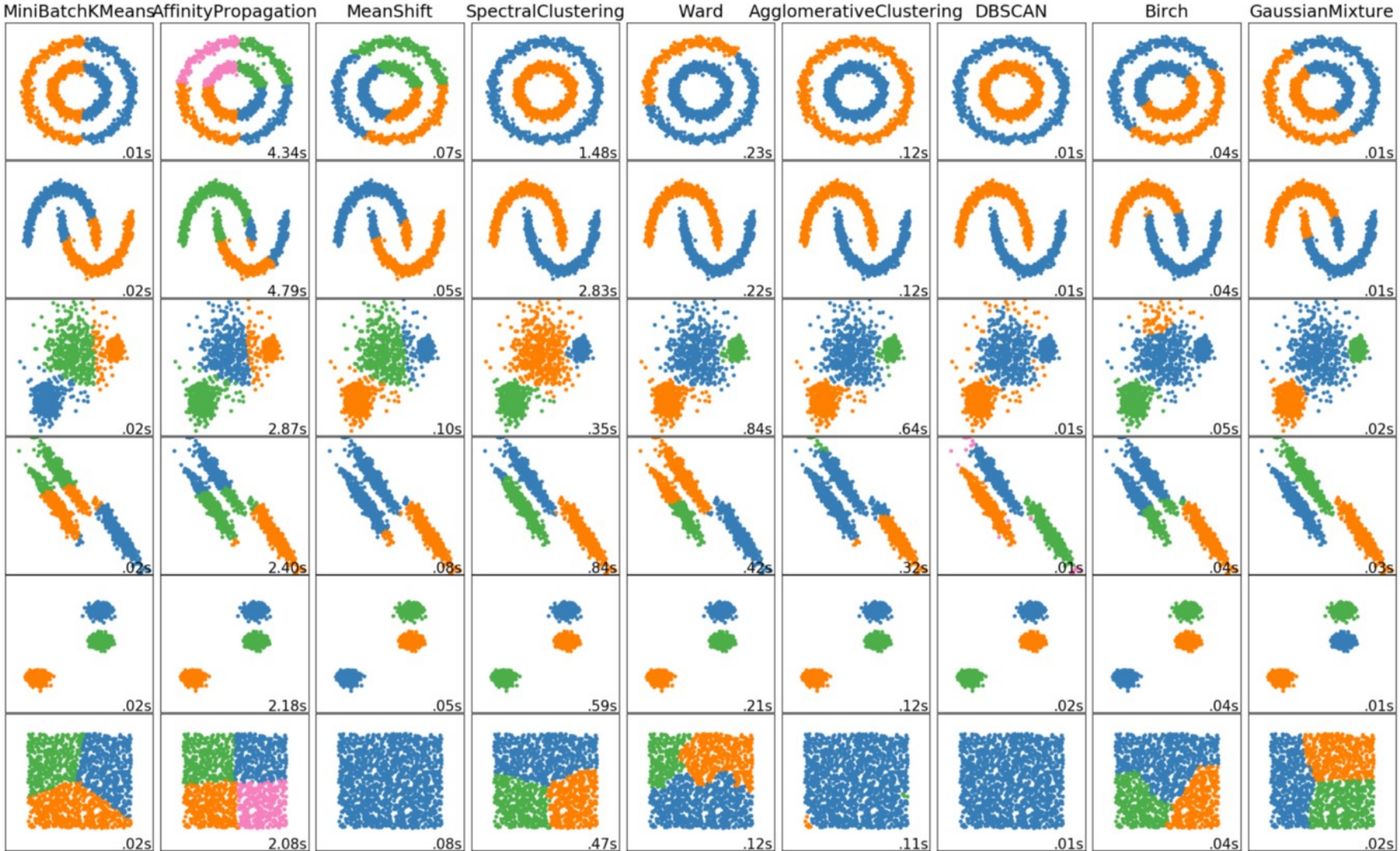
87% of 3,571 cells get the same cluster from both methods

1 Trees
2 Build a tree
3 Linkage
4 Distance
5 Real data
6 **Which method**

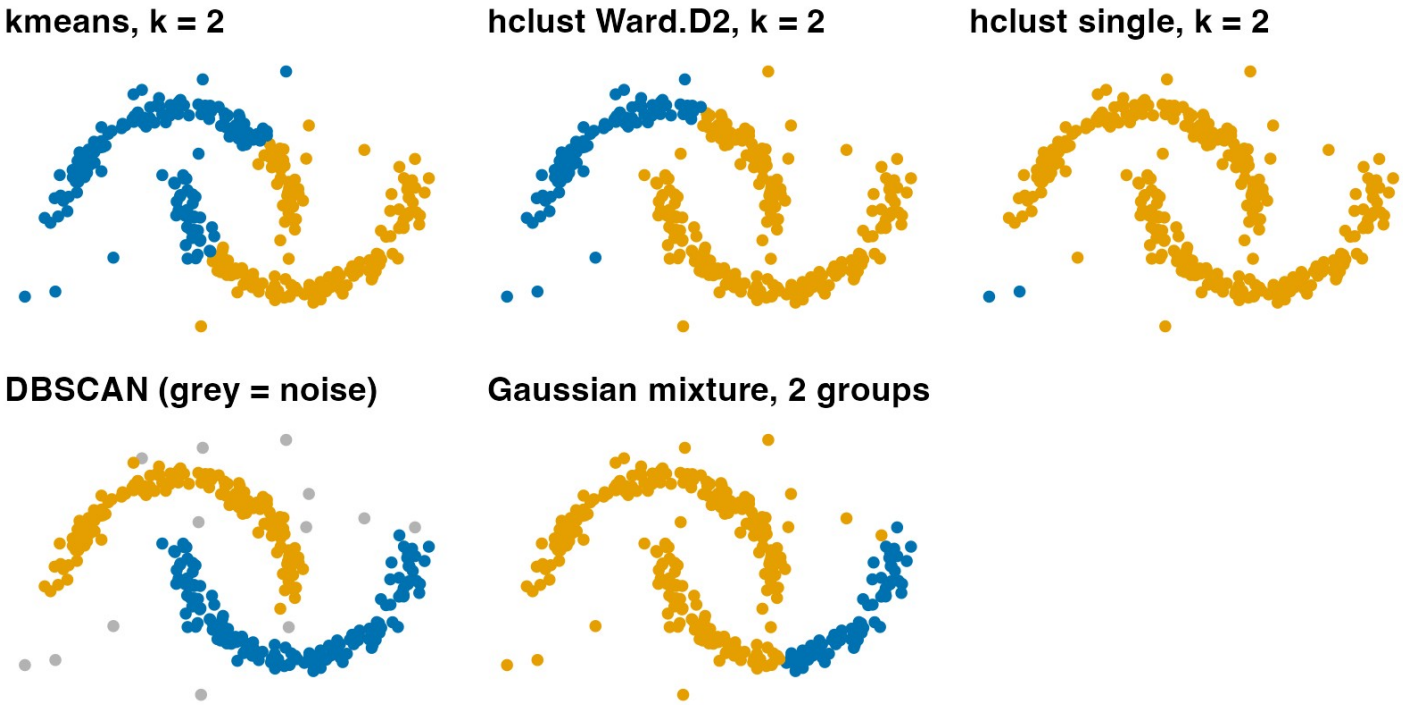


k-means or hierarchical?

	k-means	hierarchical
choose K	before	after: cut the tree
output	K groups	a tree: every K
big data (100,000s)	fast	slow: n × n distances
any distance	Euclidean only	yes
same answer twice	set.seed()	always

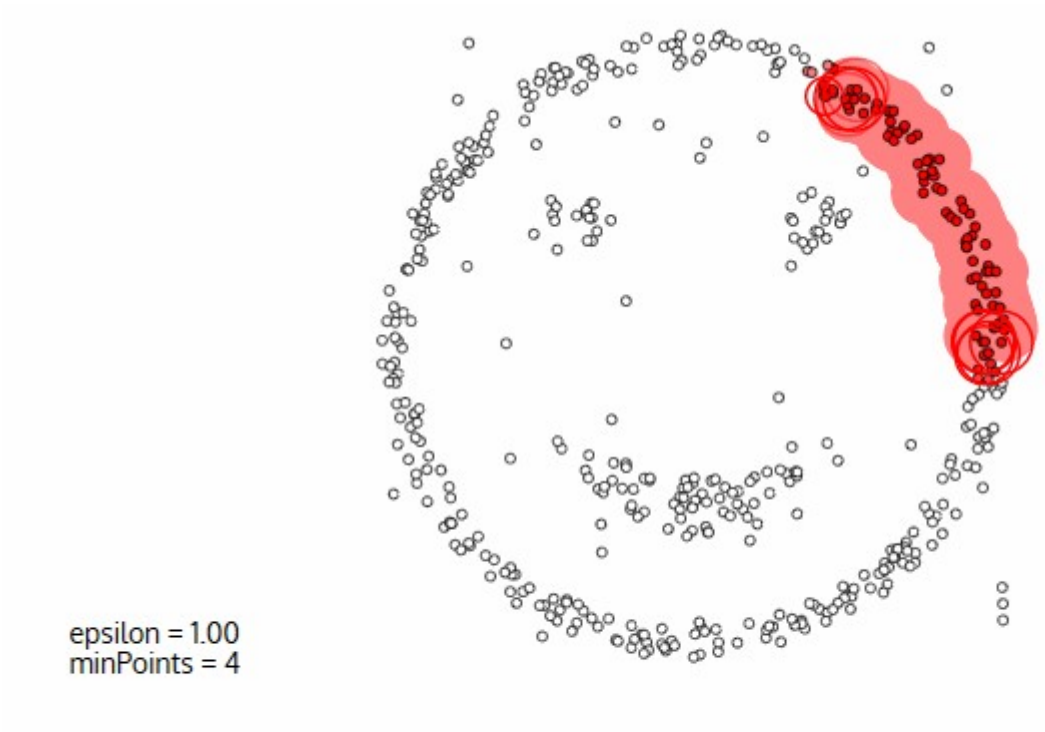

MiniBatchKMeans
AffinityPropagation
MeanShift
SpectralClustering
Ward
AgglomerativeClustering
DBSCAN
Birch
GaussianMixture
.01s
4.34s
.07s
1.48s
.23s
.12s
.01s
.04s
.01s
.02s
4.79s
.05s
2.83s
.22s
.12s
.01s
.04s
.01s
.02s
2.87s
.10s
.35s
.84s
.64s
.01s
.05s
.02s
.02s
2.40s
.08s
.84s
.42s
.32s
.01s
.04s
.03s
.02s
2.18s
.05s
.59s
.21s
.12s
.02s
.04s
.01s
.02s
2.08s
.08s
.47s
.12s
.11s
.01s
.04s
.02s

Odd shapes and noise



DBSCAN: dense regions, any shape, flags noise. **Gaussian mixture**: soft, oval groups.

Density-Based Spatial Clustering of Applications with Nois



https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68

Live coding: hclust



`Week8_live_2_hclust.R`

toy → plants → genes → Colombia

dist(), hclust(), cutree()

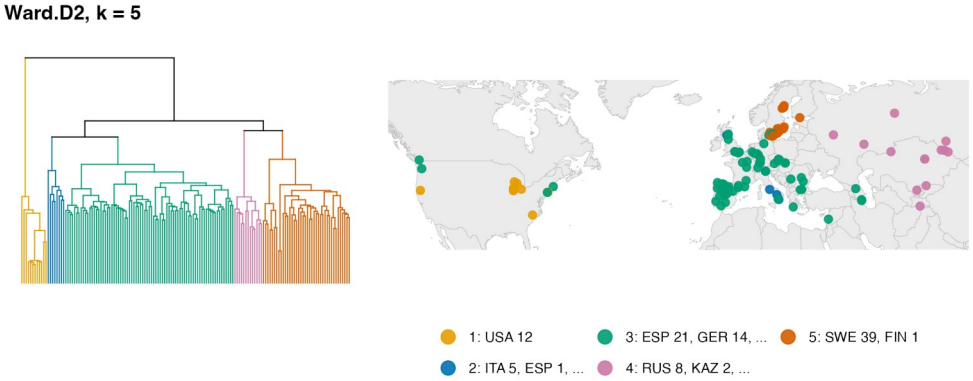
four linkages on one data set

plants on a map

heatmap ordered by two trees

table(hclust, kmeans)

Problem Set 8: Clustering



k-means and hierarchical clustering

Choose K, a distance and a linkage

Due Monday, November 23