Clustering: hierarchical

2024-09-05

- **M1 MIDS/MFA**
- **Université Paris Cité**
- Année 2024-2025
- Course Homepage

- Moodle

Setup

```
stopifnot(
  require(DT),
  require(skimr),
  require(GGally),
  require(patchwork),
  require(ggforce),
  require(glue),
  require(ggfortify),
  require(ggvoronoi),
  require(magrittr),
  require(broom),
  require(ggdendro),
  require(dendextend),
  require(plotly),
  require(tidyverse)
)

tidymodels::tidymodels_prefer(quiet = TRUE)

old_theme <-theme_set(
  theme_minimal(base_size=9,
                base_family = "Helvetica")
)
```

```
knitr::opts_chunk$set(
  message = FALSE,
  warning = FALSE,
  comment=NA,
  prompt=FALSE,
  cache=FALSE,
  echo=TRUE,
  results='asis'
)
```

```r
gc <- options(ggplot2.discrete.colour="viridis")
gc <- options(ggplot2.discrete.fill="viridis")
gc <- options(ggplot2.continuous.fill="viridis")
gc <- options(ggplot2.continuous.colour="viridis")
```

! **Objectives**

Preamble

Hierarchical clustering builds *dendrograms*

Explore the data structure: dendrograms (objects of class `dendrogram`) are represented by *lists of lists with attributes* (not by `tibbles`).

The dendrograms created from objects of class `hclust` represent *planar binary trees*.

Question

- How do you define abstractly planar binary trees?
- In dendrograms created from objects of class `hclust`, what do the leaf nodes represent?
- In dendrograms created from objects of class `hclust`, what do the internal nodes represent ?

Keep an eye on Introduction to dendextend by the package author Tal Galili.

Playing with a toy dendrogram

```r
dend <- 1:5 %>%
  dist %>%
  hclust(method="ward.D2") %>%
  as.dendrogram
```

Nodes are identified by their prefix order index (note that this depend on the chosen rotation).

```r
dend %>%
  rotate(c(1,2,4,5,3)) %>%
  get_nodes_attr("members",
                 id = c(1, 2, 5, 7))
```

```
[1] 5 2 3 1
```

```r
cophenetic(rotate(dend, c(1,2,4,5,3)))
```

```
         1        2        3        4
2 1.000000
3 3.872983 3.872983
4 3.872983 3.872983 1.000000
5 3.872983 3.872983 1.732051 1.732051
```

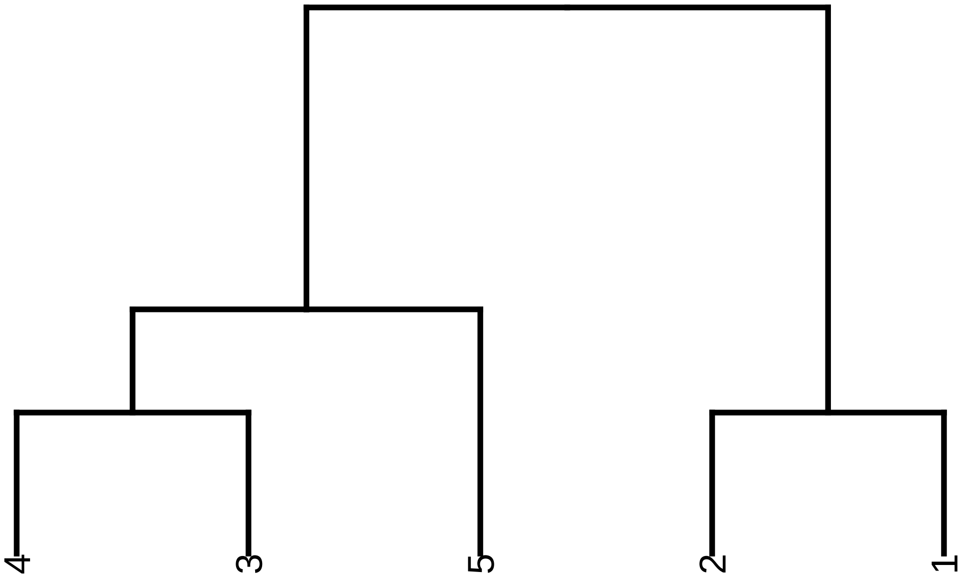
```r
cophenetic(dend)
```

```
         1        2        5        3
2 1.000000                            
5 3.872983 3.872983                   
3 3.872983 3.872983 1.732051          
4 3.872983 3.872983 1.732051 1.000000
```

```
dend %>%
  rotate(c(1,2,4,5,3)) %>%
  get_nodes_attr("height")
```

```
[1] 3.872983 1.000000 0.000000 0.000000 1.732051 1.000000 0.000000 0.000000
[9] 0.000000
```

```
as.ggdend(rev(dend))
```



```
# kmeans(tibble(x=1:5), centers = 2)
```

```
# Get various attributes
dend %>%
  get_nodes_attr("height") # node's height
```

```
[1] 3.872983 1.000000 0.000000 0.000000 1.732051 0.000000 1.000000 0.000000
[9] 0.000000
```

How is attributed `height` computed? What is its purpose?

What kind of tree traversal is used by `get_nodes_...` helpers?

```
dend %>%
  get_nodes_attr("members")
```

```
[1] 5 2 1 1 3 1 2 1 1
```

Tweaking a dendrogram

Why should we do that?

How should we do that?

USArrests

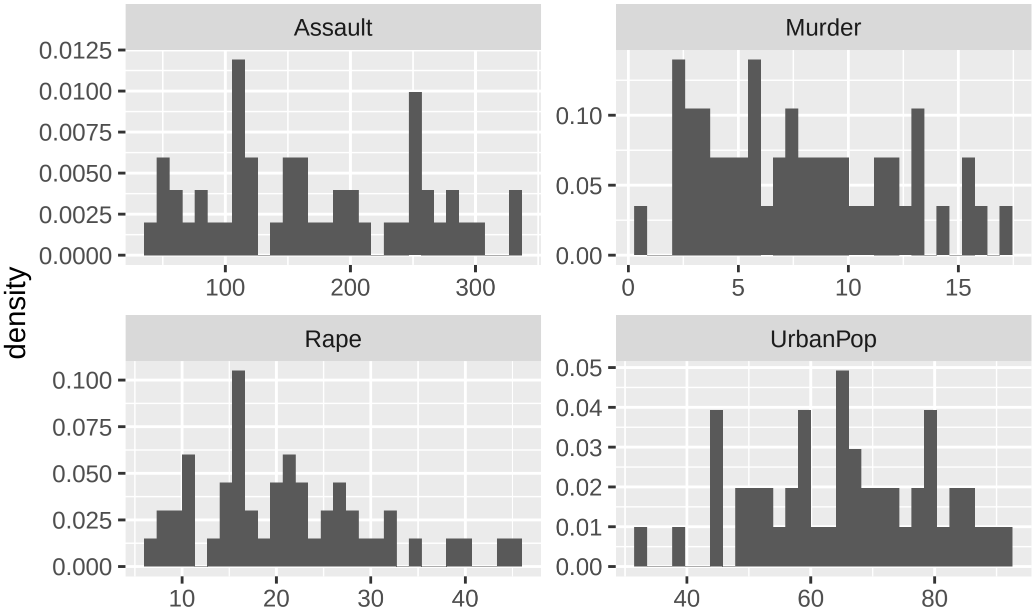
We work on `USArrests` dataset. We want to classify the 50 (united) states on the basis of the arrests profile and the urbanization rate. We rely on hierarchical, bottom-up classification.

```
data("USArrests")

USArrests <- USArrests %>%
    tibble::rownames_to_column(var="region")

USArrests <- USArrests %>%
    mutate(region = tolower(region))

rownames(USArrests) <- USArrests$region

glimpse(USArrests)
```

```
Rows: 50
Columns: 5
$ region   <chr> "alabama", "alaska", "arizona", "arkansas", "california", "co~
$ Murder   <dbl> 13.2, 10.0, 8.1, 8.8, 9.0, 7.9, 3.3, 5.9, 15.4, 17.4, 5.3, 2.~
$ Assault  <int> 236, 263, 294, 190, 276, 204, 110, 238, 335, 211, 46, 120, 24~
$ UrbanPop <int> 58, 48, 80, 50, 91, 78, 77, 72, 80, 60, 83, 54, 83, 65, 57, 6~
$ Rape     <dbl> 21.2, 44.5, 31.0, 19.5, 40.6, 38.7, 11.1, 15.8, 31.9, 25.8, 2~
```

```
source("../_UTILS/make_biotiful.R")
```

```
make_biotifoul(USArrests, .f=is.numeric)
```



The function `dist` is used to calculate pairwise distances between individuals.

Question

Compute pairwise distances between rows of `USArrests` (with and without scaling)

```
dist.1 <-  USArrests %>%
    select(where(is.numeric)) %>%
    dist()
```

```
dist.2 <- USArrests %>%
    select(where(is.numeric)) %>%
    scale %>%
    dist()
```

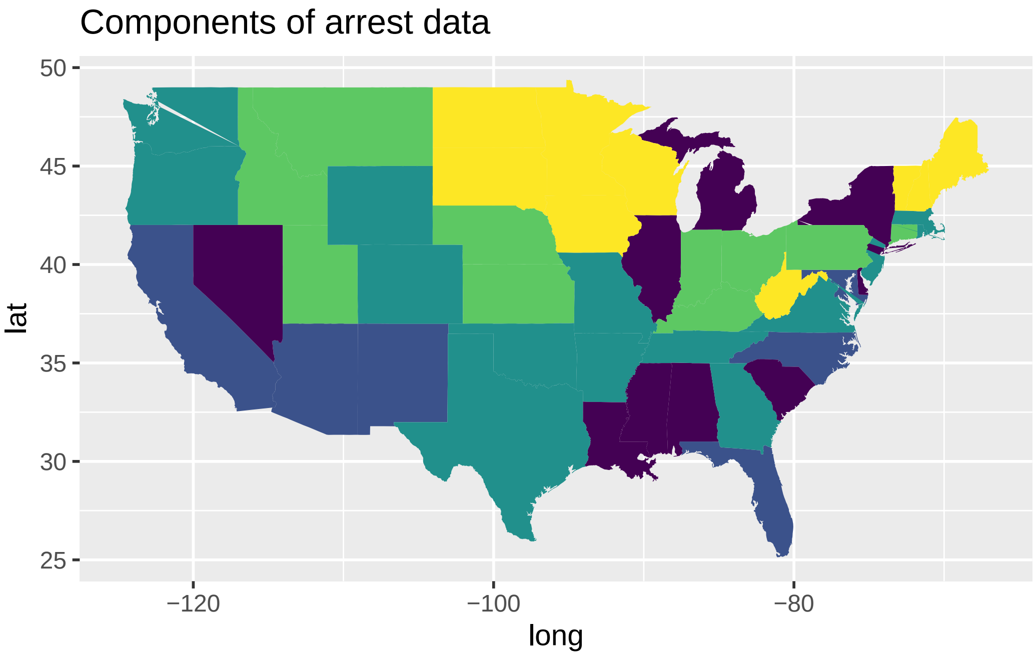
ℹ **Question**

Perform hierarchical clustering on *unscaled* and *scaled* dataset.

```
hcl.1 <- hclust(dist.1, method = "ward.D2")
hcl.2 <- hclust(dist.2, method = "ward.D2") # scaled
```

ℹ **Question**

```
mutate(USArrests,
       .cluster = factor(cutree(hcl.1, 5))) %>%
  inner_join(map_data("state"), by = "region") %>%
  ggplot() +
  aes(x=long, y=lat, group=region, fill=.cluster) +
  geom_polygon() +
  scale_fill_viridis_d() +
  ggtitle("Components of arrest data") +
  theme(legend.position = "none")
```



The dendrogram class

ℹ **Question**

Exploration of results of hierarchical clustering (objects of class `hclust`) is facilitated by converting to class `dendrogram`.

```
dend.1 <- hcl.1 %>%
  as.dendrogram()

length(dend.1[[1]][[2]])
```

```
[1] 2
```

```
class(dend.1)
```

```
[1] "dendrogram"
```

```
class(unclass(dend.1))
```

```
[1] "list"
```

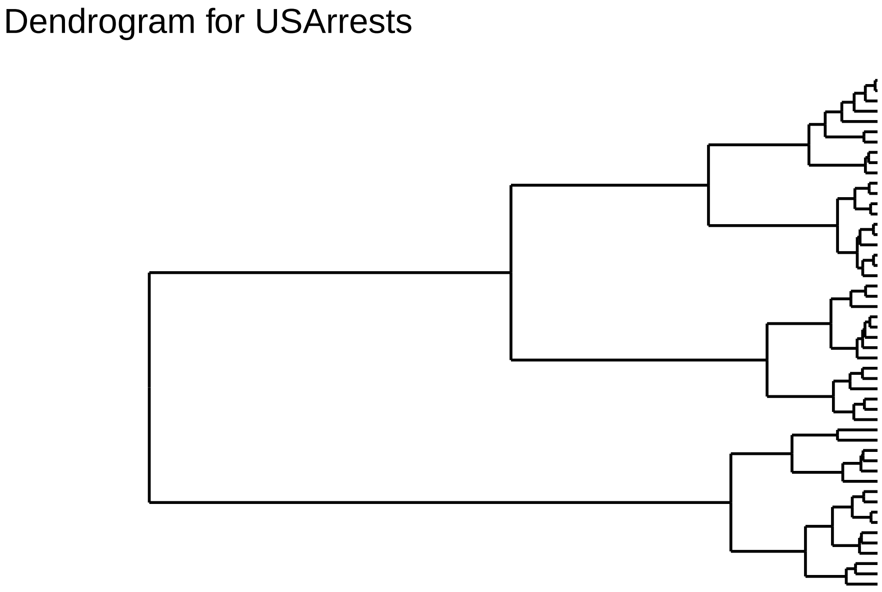
```
methods(class=class(dend.1)) %>% head()
```

```
[1] "[[.dendrogram"             "all.equal.dendrogram"
[3] "as.dendrogram.dendrogram" "as.ggdend.dendrogram"
[5] "as.hclust.dendrogram"     "click_rotate.dendrogram"
```

```
dend.1 %>% head()
```

```
--[dendrogram w/ 2 branches and 50 members at h = 701]
  |--[dendrogram w/ 2 branches and 16 members at h = 141]
  |  |--[dendrogram w/ 2 branches and 10 members at h = 69.3]
  |  |  |--[dendrogram w/ 2 branches and 3 members at h = 30.1] ..
  |  |  `--[dendrogram w/ 2 branches and 7 members at h = 43.4] ..
  |  `--[dendrogram w/ 2 branches and 6 members at h = 82.3]
  |     |--[dendrogram w/ 2 branches and 4 members at h = 33.4] ..
  |     `--[dendrogram w/ 2 branches and 2 members at h = 38.5] ..
  `--[dendrogram w/ 2 branches and 34 members at h = 353]
     |--[dendrogram w/ 2 branches and 14 members at h = 106]
     |  |--[dendrogram w/ 2 branches and 6 members at h = 42.5] ..
     |  `--[dendrogram w/ 2 branches and 8 members at h = 44.8] ..
     `--[dendrogram w/ 2 branches and 20 members at h = 163]
        |--[dendrogram w/ 2 branches and 10 members at h = 38.5] ..
        `--[dendrogram w/ 2 branches and 10 members at h = 66] ..
etc...
```

```
dend.1 %>%
  ggdendrogram(rotate = TRUE,labels = T) +
  ggtitle("Dendrogram for USArrests") +
  ggdendro::theme_dendro() +
  scale_y_reverse(expand = c(0.2, 0))
```



Question

```
# label(dend.1)

dend.2 <-  as.dendrogram(hcl.1)
# order it the closest we can to the order of the observations:
dend.2 <- rotate(dend.2, 1:50)
# Color the branches based on the clusters:
dend.2 <- color_branches(dend.2, k=3) #, groupLabels=iris_species)
# Manually match the labels, as much as possible, to the real classification of the flower
# labels_colors(dend.2) <-
#    rainbow_hcl(3)[sort_levels_values(
#        as.numeric(iris[,5])[order.dendrogram(dend.2)]
#    )]
```

Ward method

The `meth=ward.D2` option allows you to aggregate individuals according to the method of Ward, that is, according to the variance.

Question

What is the distance used? Describe the method of *classification by variance*?

The output `clas$height` gives the jump height of the dendrogram to each new iteration. In the case of Ward's method, she is proportional to the loss of inter-class variance.

Question

1. How many groups are there at step 0? at the last step?
2. How many iterations are there?
3. Recall the definition of inter-class variance.
4. What is the inter-class variance at step 0? at the last step? How is it going according to the number of groups (or according to the number of iterations)?
5. By comparing the total inertia and the 'clas$height' output, find the coefficient of proportionality between the loss of inter-class variance and height of jumps.

Choice of the number of classes

Question

1. Plot the curve corresponding to the loss of variance inter in as a function of the number of iterations :
2. Select the "optimal" number of classes.
3. Verify that, for the number of classes chosen, the number by class is sufficient (we can use the `cutree` function).
4. These classes can be represented using a dendrogram
5. You can also colour the leaves of the tree corresponding to a class. To do this, install and load the package 'dendextend'.

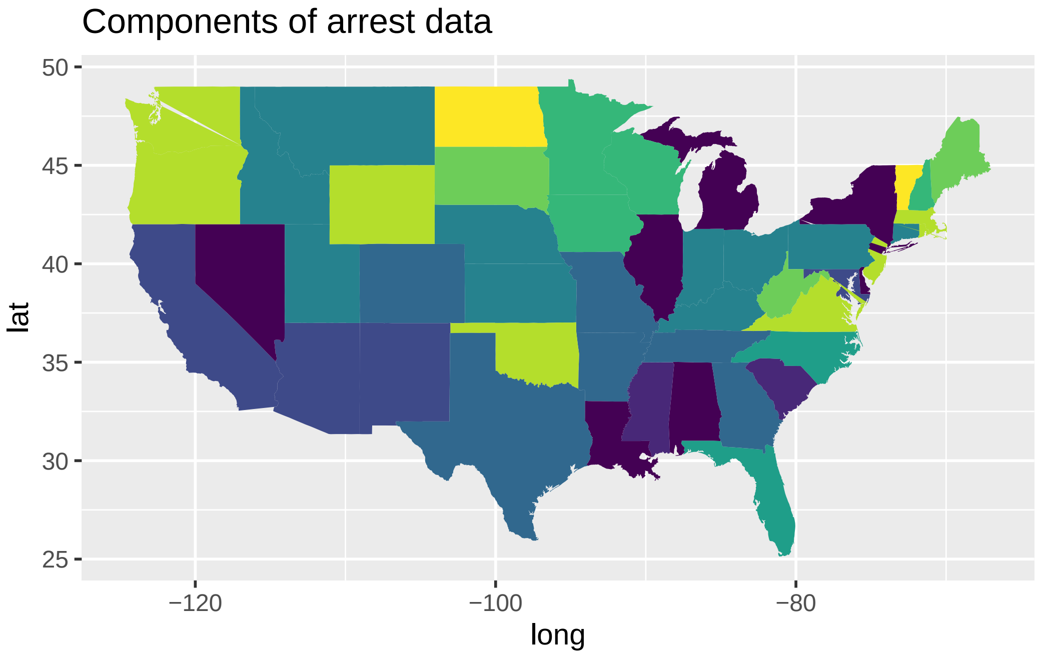
Link with PCA.

We will represent the classes obtained in the factorial design(s) obtained by the PCA. This will make it possible to represent the classes and describe them according to the variables initials.

Question

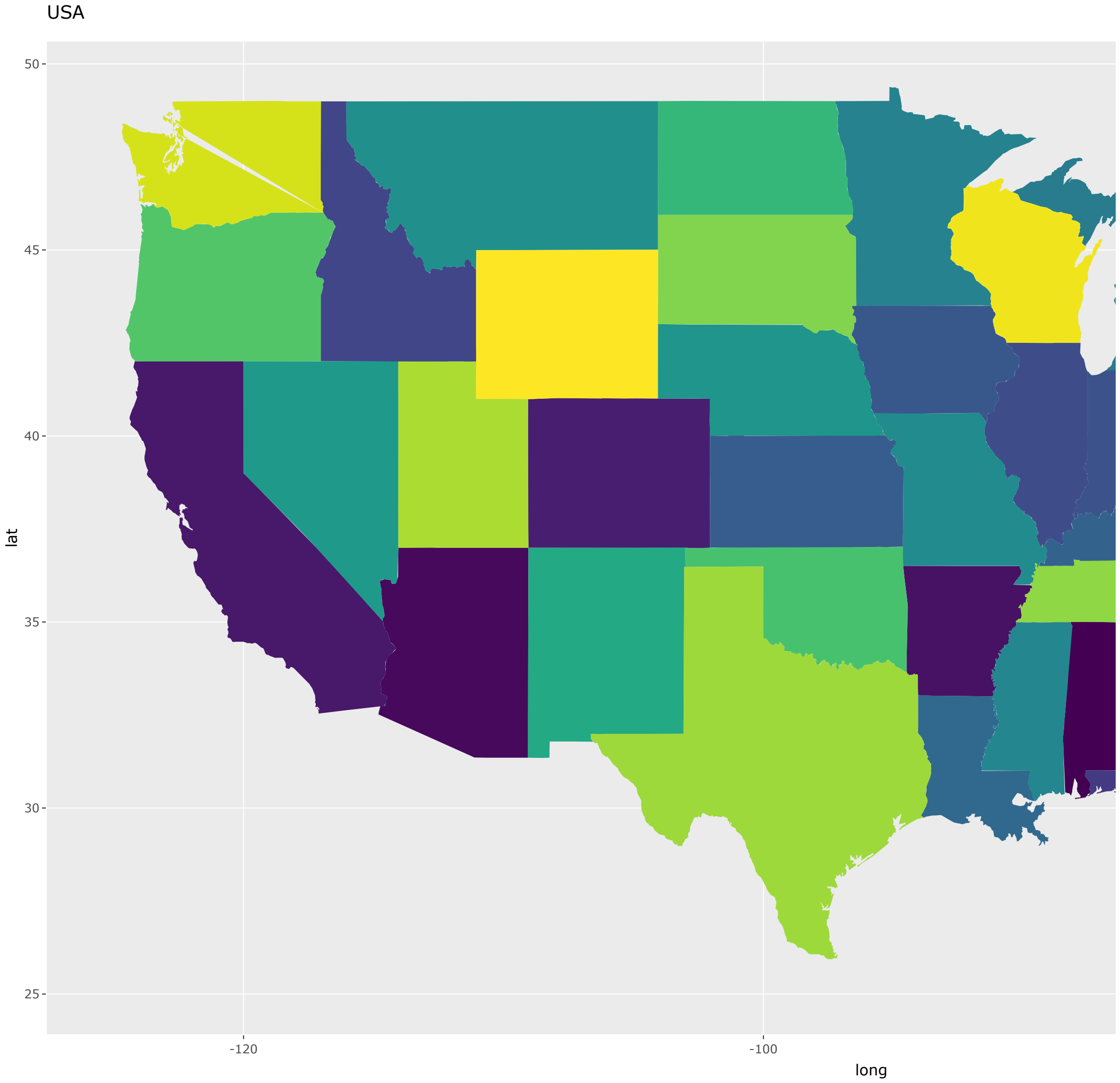
Represent the coordinates of the individuals in each group in the first factorial plane (with one color for each class). The vector generated by 'cutree' can be used to form a color vector. Interpretation.

```
mutate(USArrests,
       .cluster = factor(cutree(hcl.1, 10))) %>%
  inner_join(map_data("state"), by = "region") %>%
  ggplot() +
  aes(x=long, y=lat, group=region, fill=.cluster) +
  geom_polygon() +
  scale_fill_viridis_d() +
  ggtitle("Components of arrest data") +
  theme(legend.position = "none")
```



```
#data(france)
```

```
(map_data("state") %>%
  ggplot() +
  aes(x=long,
      y=lat,
      label=factor(region),
      fill = factor(region)) +
  geom_polygon() +
  scale_fill_viridis_d() +
  ggtitle("USA") +
  theme(legend.position = "none")) |>
    ggplotly()
```



Cophenetic distance

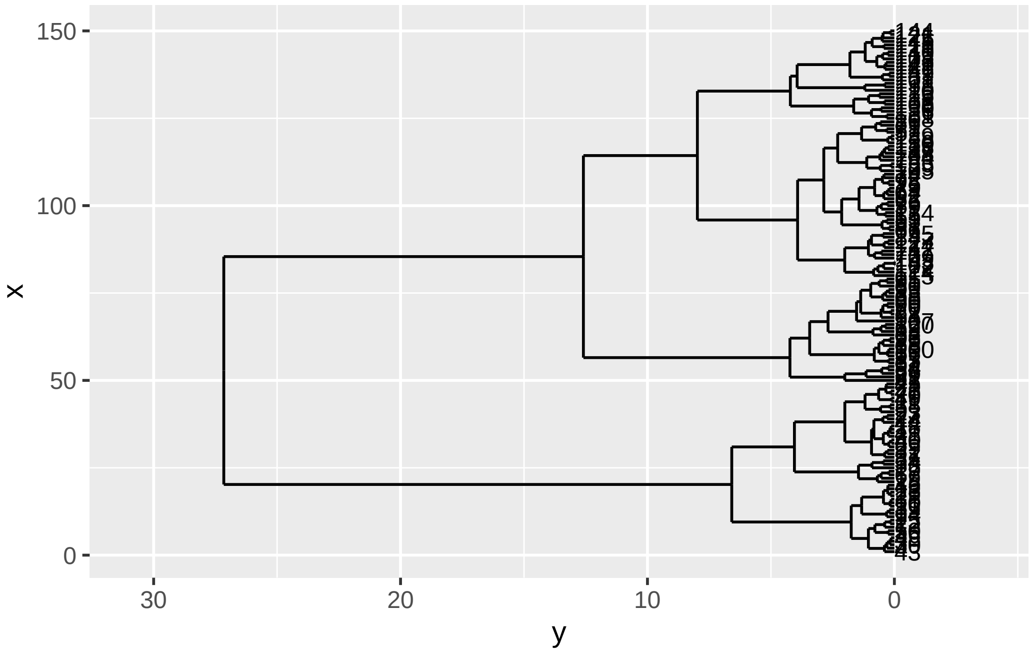
Question

```
dist.coph.1 <- cophenetic(dend.1)
```

Cophenetic distance between dendrograms

```
data("iris")

hcl.iris <- iris %>%
  select(where(is.numeric)) %>%
  scale() %>%
  dist() %>%
  hclust(meth="ward.D2")

dend.iris <-  dendro_data(hcl.iris)
```

```r
dend.iris %$% (
  ggplot() +
  geom_segment(data = segments,
               aes(x = x, y = y,
                   xend = xend, yend = yend)
  ) +
  geom_text(data = labels,
            aes(x = x, y = y,
                label = label, hjust = 0),
            size = 3
  ) +
  coord_flip() +
  scale_y_reverse(expand = c(0.2, 0))
)
```



References

ggdendro

dendroextra

hier_clust, tidyclust