Heatmap\_clustering

Gabriel Burcea

03/08/2020

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.3.1 ✓ purrr 0.3.4  
## ✓ tibble 3.0.3 ✓ dplyr 1.0.0  
## ✓ tidyr 1.1.0 ✓ stringr 1.4.0  
## ✓ readr 1.3.1 ✓ forcats 0.5.0

## ── Conflicts ────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────── tidyverse\_conflicts() ──  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(cluster)  
library(plotly)

##   
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':  
##   
## last\_plot

## The following object is masked from 'package:stats':  
##   
## filter

## The following object is masked from 'package:graphics':  
##   
## layout

library(fpc)  
library(dendextend)

##   
## ---------------------  
## Welcome to dendextend version 1.13.4  
## Type citation('dendextend') for how to cite the package.  
##   
## Type browseVignettes(package = 'dendextend') for the package vignette.  
## The github page is: https://github.com/talgalili/dendextend/  
##   
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues  
## Or contact: <tal.galili@gmail.com>  
##   
## To suppress this message use: suppressPackageStartupMessages(library(dendextend))  
## ---------------------

##   
## Attaching package: 'dendextend'

## The following object is masked from 'package:stats':  
##   
## cutree

library(factoextra)

## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa

library(FactoMineR)  
library(NbClust)  
library(caret)

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

library(DMwR)

## Loading required package: grid

## Registered S3 method overwritten by 'quantmod':  
## method from  
## as.zoo.data.frame zoo

data\_cluster <- read\_csv("/Users/gabrielburcea/rprojects/stats\_data\_whole/data\_categ\_no\_sev.csv")

## Parsed with column specification:  
## cols(  
## .default = col\_character(),  
## age = col\_double(),  
## date\_completed = col\_date(format = ""),  
## how\_unwell = col\_double(),  
## number\_days\_symptoms = col\_double(),  
## number\_morbidities = col\_double()  
## )

## See spec(...) for full column specifications.

## Warning: 9 parsing failures.  
## row col expected actual file  
## 2910 how\_unwell a double How Unwell '/Users/gabrielburcea/rprojects/stats\_data\_whole/data\_categ\_no\_sev.csv'  
## 10025 number\_days\_symptoms no trailing characters 0 '/Users/gabrielburcea/rprojects/stats\_data\_whole/data\_categ\_no\_sev.csv'  
## 19970 number\_days\_symptoms no trailing characters 9 '/Users/gabrielburcea/rprojects/stats\_data\_whole/data\_categ\_no\_sev.csv'  
## 22410 number\_days\_symptoms a double Plus de 21 '/Users/gabrielburcea/rprojects/stats\_data\_whole/data\_categ\_no\_sev.csv'  
## 31280 number\_days\_symptoms no trailing characters ?? ???? ?? '/Users/gabrielburcea/rprojects/stats\_data\_whole/data\_categ\_no\_sev.csv'  
## ..... .................... ...................... ........... .......................................................................  
## See problems(...) for more details.

level\_key\_temperature <- c("Yes" = "37.5-38",   
 "Yes" = "38.1-39",   
 "Yes" = "38.2-39",  
 "Yes" = "39.1-41")  
  
  
  
data\_select <- data\_cluster %>%  
 dplyr::select(ID, covid\_tested, chills, cough, diarrhoea, fatigue, headache, loss\_smell\_taste, muscle\_ache,   
 nasal\_congestion, nausea\_vomiting, shortness\_breath, sore\_throat, sputum, temperature, asthma, diabetes\_type\_one,  
 diabetes\_type\_two, obesity, hypertension, heart\_disease, lung\_condition, liver\_disease, kidney\_disease) %>%  
 dplyr::filter(covid\_tested != "none")   
  
covid\_tested\_levels <- c("positive" = "showing symptoms")  
  
data\_transf <- data\_select %>%   
 dplyr::mutate(covid\_tested = forcats::fct\_recode(covid\_tested, !!!covid\_tested\_levels),   
 temperature = forcats::fct\_recode(temperature, !!!level\_key\_temperature)) %>%  
 dplyr::filter(covid\_tested != "none")

## Warning: Unknown levels in `f`: 38.2-39

data\_piv <- data\_transf %>%  
 pivot\_longer(cols = 16:24,   
 names\_to = "Comorbidities",  
 values\_to = "Bolean") %>%  
 dplyr::filter(Bolean == "Yes") %>%  
 pivot\_longer(cols = 3:15,   
 names\_to = "Symptoms",   
 values\_to = "Yes\_No") %>%  
 dplyr::group\_by(Comorbidities, Symptoms, Yes\_No) %>%  
 dplyr::summarise(Count = n()) %>%  
 dplyr::mutate(Freq = Count / sum(Count)\*100) %>%   
 dplyr::filter(Yes\_No == "Yes")

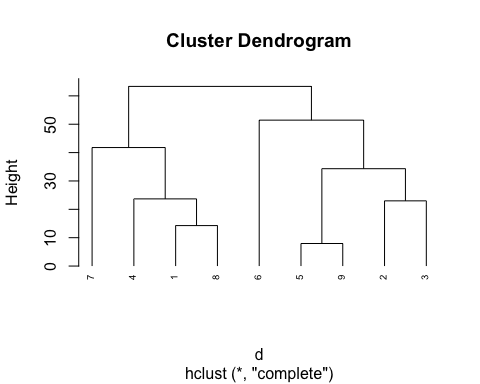
## `summarise()` regrouping output by 'Comorbidities', 'Symptoms' (override with `.groups` argument)

data\_piv$Bolean <- NULL  
data\_piv$Yes\_No <- NULL  
data\_piv$Count <- NULL  
  
data\_piv <- data\_piv %>%  
 pivot\_wider(names\_from = Symptoms, values\_from = Freq)  
  
data\_piv$Comorbidities <- as.character(data\_piv$Comorbidities)  
  
# scale data

# Disimilarity matrix   
d <- dist(data\_piv, method = "euclidean")

## Warning in dist(data\_piv, method = "euclidean"): NAs introduced by coercion

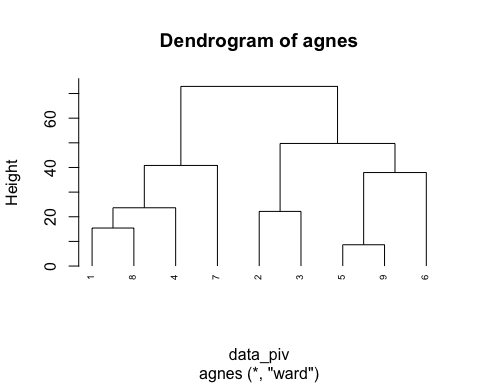
#Hierarchicla clustering using Complete Linkage  
hc\_complete <- hclust(d, method = "complete")  
  
#Plot the obtained dendogram   
plot(hc\_complete, cex = 0.6, hang = -31)



# Compute with agnes   
hc\_agnes <- agnes(data\_piv, method = "complete")  
  
# Agglomerative coeffiecient   
hc\_agnes$ac

## [1] 0.626452

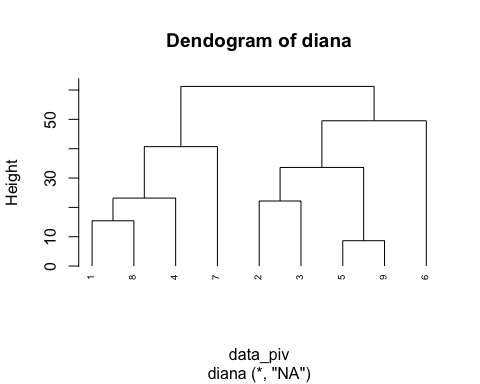
hc\_agnes\_2 <- agnes(data\_piv, method = "ward")  
  
pltree(hc\_agnes\_2, cex = 0.6, hang = -1, main = "Dendrogram of agnes")

 Divisive Hierarchicla Clustering

# compute divisive hierarchical clustering   
hc\_diana<- diana(data\_piv)  
  
# Divisive coefficient; amount of clustering structure found   
hc\_diana$dc

## [1] 0.626452

#plot dendogram   
  
pltree(hc\_diana, cex = 0.6, hang = -1, main = "Dendogram of diana")

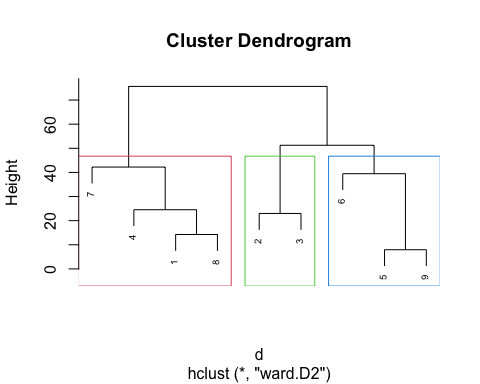


Working with Dendrograms

# Ward's method   
  
hc\_ward\_method <- hclust(d, method = "ward.D2")  
  
# Cut tree into 4 groups   
  
sub\_grp <- cutree(hc\_ward\_method, k = 4)  
  
#Number of countries in each cluster  
table(sub\_grp)

## sub\_grp  
## 1 2 3 4   
## 3 2 3 1

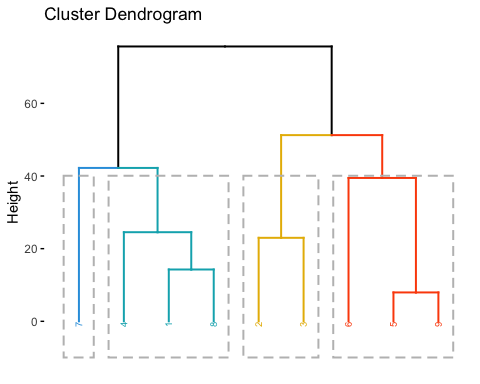
plot(hc\_ward\_method, cex =0.6)  
  
rect.hclust(hc\_ward\_method , k = 3, border = 2:5)



res\_hc <- data\_piv %>%  
 dist(method = "euclidean") %>%  
 hclust(method = "ward.D2")

## Warning in dist(., method = "euclidean"): NAs introduced by coercion

fviz\_dend(res\_hc, k = 4,   
 cex = 0.5,   
 k\_colors = c("#2E9FDF", "#00AFBB", "#E7B800", "#FC4E07"),   
 color\_labels\_by\_k = TRUE,   
 rect = TRUE)



#data\_piv$chills <- as.numeric(data\_piv$chills )  
# data\_piv$cough <- as.integer(data\_piv$cough )  
# data\_piv$diarrhoea <- as.integer(data\_piv$diarrhoea)  
# data\_piv$fatigue <- as.integer(data\_piv$fatigue)  
# data\_piv$headache <- as.integer(data\_piv$headache)  
# data\_piv$loss\_smell\_taste <- as.integer(data\_piv$loss\_smell\_taste)  
# data\_piv$muscle\_ache <- as.integer(data\_piv$muscle\_ache)  
# data\_piv$nasal\_congestion <- as.integer(data\_piv$nasal\_congestion)  
# data\_piv$nausea\_vomiting <- as.integer(data\_piv$nausea\_vomiting)  
# data\_piv$shortness\_breath <- as.integer(data\_piv$shortness\_breath)  
# data\_piv$sore\_throat <- as.integer(data\_piv$sore\_throat)  
# data\_piv$sputum <- as.integer(data\_piv$sputum)  
# data\_piv$temperature <- as.integer(data\_piv$temperature)  
  
level\_key\_comorbidities <-  
 c("kidney disease" = "kidney\_disease",  
 "lung condition" = "lung\_condition",  
 "diabetes type one" = "diabetes\_type\_one",  
 "diabetes type two" = "diabetes\_type\_two",  
 "liver disease" = "liver\_disease",  
 "heart disease" = "heart\_disease")  
  
  
data\_pi<- data\_piv %>%  
 tidyr::replace\_na(list(nausea\_vomiting = 0)) %>%  
 dplyr::mutate(Comorbidities = forcats::fct\_recode(Comorbidities, !!!level\_key\_comorbidities)) %>%  
 dplyr::rename('loss of smell and taste' = loss\_smell\_taste, 'muscle ache' = muscle\_ache, 'nasal congestion'= nasal\_congestion,  
 'nausea and vomiting' = nausea\_vomiting, 'shortness of breath' = shortness\_breath, 'sore throat' = sore\_throat)

## Warning: Unknown levels in `f`: kidney\_disease, lung\_condition,  
## diabetes\_type\_one, diabetes\_type\_two, liver\_disease, heart\_disease

## Warning: Unknown levels in `f`: kidney\_disease, lung\_condition,  
## diabetes\_type\_two, liver\_disease, heart\_disease

## Warning: Unknown levels in `f`: kidney\_disease, lung\_condition,  
## diabetes\_type\_one, liver\_disease, heart\_disease

## Warning: Unknown levels in `f`: kidney\_disease, lung\_condition,  
## diabetes\_type\_one, diabetes\_type\_two, liver\_disease

## Warning: Unknown levels in `f`: kidney\_disease, lung\_condition,  
## diabetes\_type\_one, diabetes\_type\_two, liver\_disease, heart\_disease

## Warning: Unknown levels in `f`: lung\_condition, diabetes\_type\_one,  
## diabetes\_type\_two, liver\_disease, heart\_disease

## Warning: Unknown levels in `f`: kidney\_disease, lung\_condition,  
## diabetes\_type\_one, diabetes\_type\_two, heart\_disease

## Warning: Unknown levels in `f`: kidney\_disease, diabetes\_type\_one,  
## diabetes\_type\_two, liver\_disease, heart\_disease

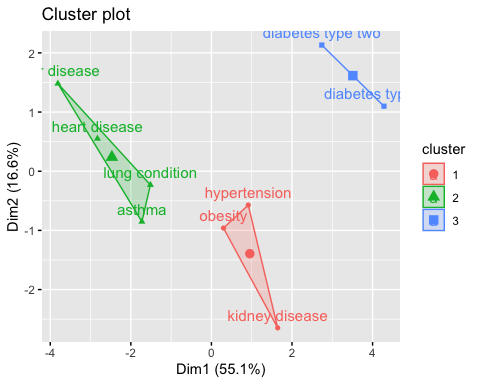
## Warning: Unknown levels in `f`: kidney\_disease, lung\_condition,  
## diabetes\_type\_one, diabetes\_type\_two, liver\_disease, heart\_disease

data\_pi <- as.data.frame(data\_pi)  
data\_pi$Comorbidities <- as.character(data\_pi$Comorbidities)  
rownames(data\_pi) <- data\_pi$Comorbidities  
#data\_scaled <- as.data.frame(scale(data\_piv[2:14]))

km\_res <- kmeans(data\_pi[,2:14], centers = 3, nstart = 25)  
  
print(km\_res)

## K-means clustering with 3 clusters of sizes 3, 4, 2  
##   
## Cluster means:  
## chills cough diarrhoea fatigue headache loss of smell and taste  
## 1 15.31119 57.93767 19.454553 49.45755 36.29471 18.81212  
## 2 31.02899 57.99607 22.495263 60.68110 43.61200 26.20437  
## 3 13.91683 59.47351 9.691482 50.21797 27.31388 14.88934  
## muscle ache nasal congestion nausea and vomiting shortness of breath  
## 1 42.30317 32.00114 4.633963 40.54518  
## 2 53.46396 35.38179 10.843595 51.05471  
## 3 42.82361 23.60832 7.444668 25.90543  
## sore throat sputum temperature  
## 1 51.88268 38.40512 21.93232  
## 2 51.52098 46.03112 26.12976  
## 3 34.62441 36.73709 35.89873  
##   
## Clustering vector:  
## asthma diabetes type one diabetes type two heart disease   
## 2 3 3 2   
## hypertension kidney disease liver disease lung condition   
## 1 1 2 2   
## obesity   
## 1   
##   
## Within cluster sum of squares by cluster:  
## [1] 729.2733 1201.6567 245.3031  
## (between\_SS / total\_SS = 63.9 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss" "tot.withinss"  
## [6] "betweenss" "size" "iter" "ifault"

fviz\_cluster(km\_res, data = data\_pi[2:14])



# Cut agnes() tree into 3 groups  
hc\_a <- agnes(data\_pi, method = "ward")  
cutree(as.hclust(hc\_a), k = 3)

## asthma diabetes type one diabetes type two heart disease   
## 1 2 2 1   
## hypertension kidney disease liver disease lung condition   
## 3 3 1 1   
## obesity   
## 3

# Cut diana() tree into 3 groups  
hc\_d <- diana(data\_pi)  
cutree(as.hclust(hc\_d), k = 3)

## asthma diabetes type one diabetes type two heart disease   
## 1 2 2 1   
## hypertension kidney disease liver disease lung condition   
## 2 3 1 1   
## obesity   
## 2

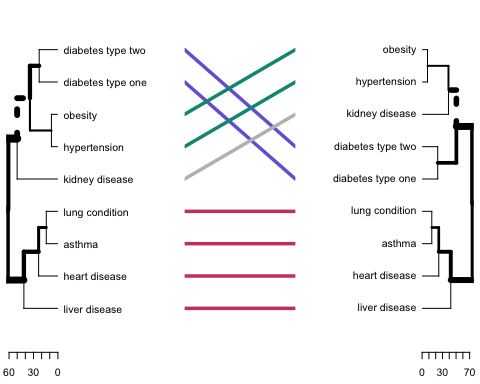
Comparing two deprograms. Comparing hierarchical clustering with complete linkage versus Ward’s method.

The output displays “unique” nodes, with a combination of labels/items not present in the other tree highlighted with dashed line. The quality of the alignment of the two trees can be measured using the function entanglement. Entanglement is a measure between 1 (full entanglement) 0 (no entanglement). A lower entanglement coefficient corresponds to a good alignment.

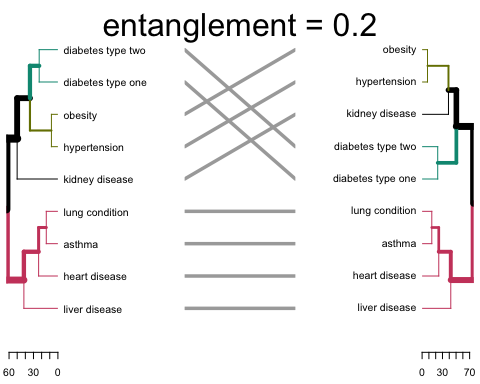
# Compute distance matrix  
res\_dist <- dist(data\_pi, method = "euclidean")

## Warning in dist(data\_pi, method = "euclidean"): NAs introduced by coercion

# Compute 2 hierarchical clusterings  
hc\_complete <- hclust(res\_dist, method = "complete")  
hc\_ward <- hclust(res\_dist, method = "ward.D2")  
  
# Create two dendrograms  
dend\_complete <- as.dendrogram (hc\_complete)  
dend\_ward <- as.dendrogram (hc\_ward)  
  
tanglegram(dend\_complete, dend\_ward,   
 margin\_inner = 10)



dend\_list <- dendlist(dend\_complete, dend\_ward)  
  
tanglegram(dend\_complete, dend\_ward,   
 margin\_inner = 10,  
 highlight\_distinct\_edges = FALSE, #Turn-off dashed line  
 common\_subtrees\_color\_lines = FALSE, # Turn-off line colors  
 common\_subtrees\_color\_branches = TRUE, # Color common branches  
 main = paste("entanglement =", round(entanglement(dend\_list),2)))



Partitioning clustering

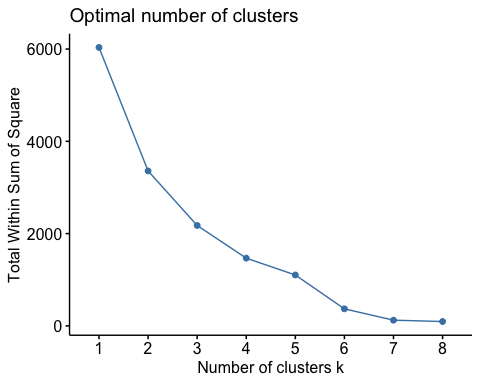
I. K-means clustering: 1. each cluster represented by the center (mean of the data points) 2. sensitive to outliers

1. K-medoids clustering /PAM (Partitioning around Medoids)
2. each cluster represented by one of the objects in the cluster
3. less sensitive to outliers
4. CLARA algorithm (Clustering Large Applications) = PAM for large datasets ??

Optimal number of clusters

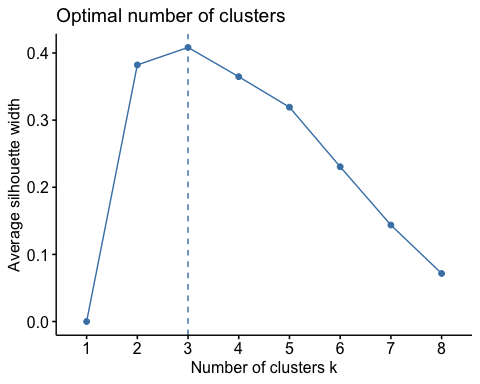
1. Elbow method: location of bend in WSS(within-cluster sum of square for eac K) plot - minimize intra-cluster variation
2. Average silhouette method: maximum of average silhouette curve (average silhouette of observations for each K)
3. Gap statistic method: compares the total within intra-cluster variation for different values of K with their expected values under null reference distribution of the data

fviz\_nbclust(data\_pi[2:14], kmeans, method = "wss",k.max = 8 )



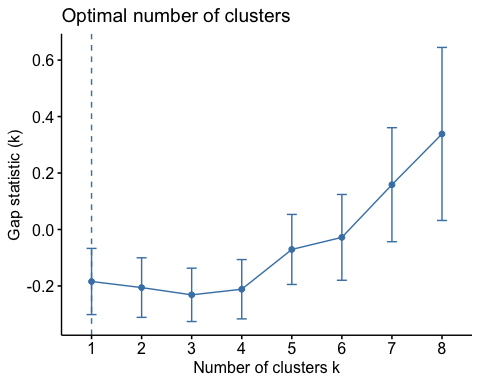
Average Silhouette Method - this method does suggest the same number of clusters just as elbow has indicated.

fviz\_nbclust(data\_pi[2:4], kmeans, method = "silhouette", k.max = 8)



Gap Statistic Method - the same

gap\_stat <- clusGap(data\_pi[2:14], kmeans, nstart = 25, K.max = 8, B = 50)  
  
fviz\_gap\_stat(gap\_stat)



# data\_piv %>%  
# mutate(Cluster = km\_res$cluster) %>%  
# group\_by(Cluster) %>%  
# summarise\_all("mean")

library(pheatmap)  
pheatmap(t(data\_pi[-1]), cutree\_cols = 3)

