AHA 7: Clustering | Hierarchical

2024-02-21

Load packages (and set seed)

```
# Load packages
library(cluster); library(factoextra)
library(NbClust); library(igraph)
set.seed(42)
```

Load data

```
# Load data
bp_data <- read.csv("../data/bipolar_depression/bipolar_depression_clean.csv")
```

Data wrangling

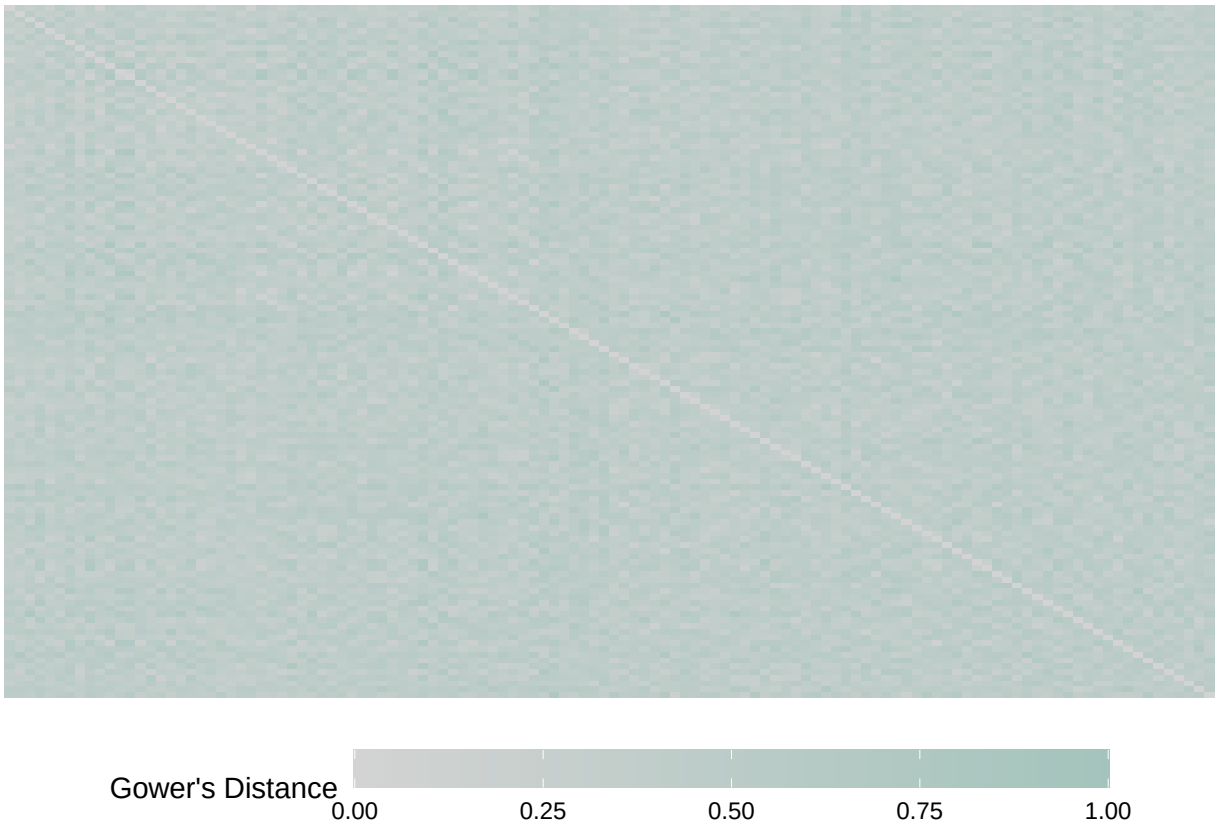
```
# Get expert diagnosis
expert <- bp_data$Expert.Diagnose

# Extract variables of interest
bp_voi <- apply(bp_data[,-c(1,19)], 2, as.numeric)
```

Compute Gower's distance

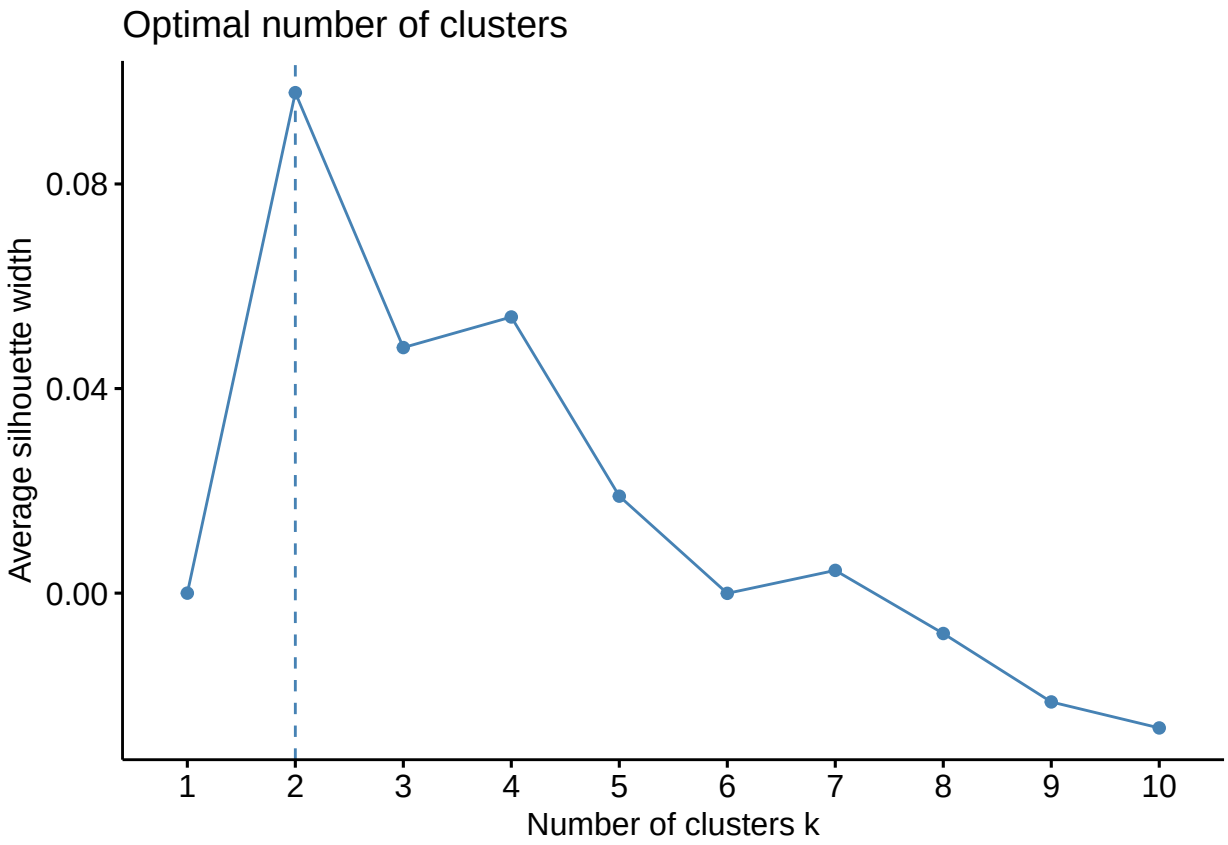
```
# Compute Gower's distance
bp_distance <- daisy(x = bp_voi, metric = "gower")

# Produce heatmap
EGAnet:::ggheatmap(bp_distance) +
  scale_fill_gradient(
    name = "Gower's Distance",  limits = c(0, 1),
    low = "lightgrey", high = "#A3C4BC"
  ) + theme(
    axis.text = element_blank(), axis.title = element_blank(),
    axis.ticks = element_blank(), legend.position = "bottom",
    legend.key.width = unit(2, "cm"),
    legend.key.height = unit(0.5, "cm")
  )
```

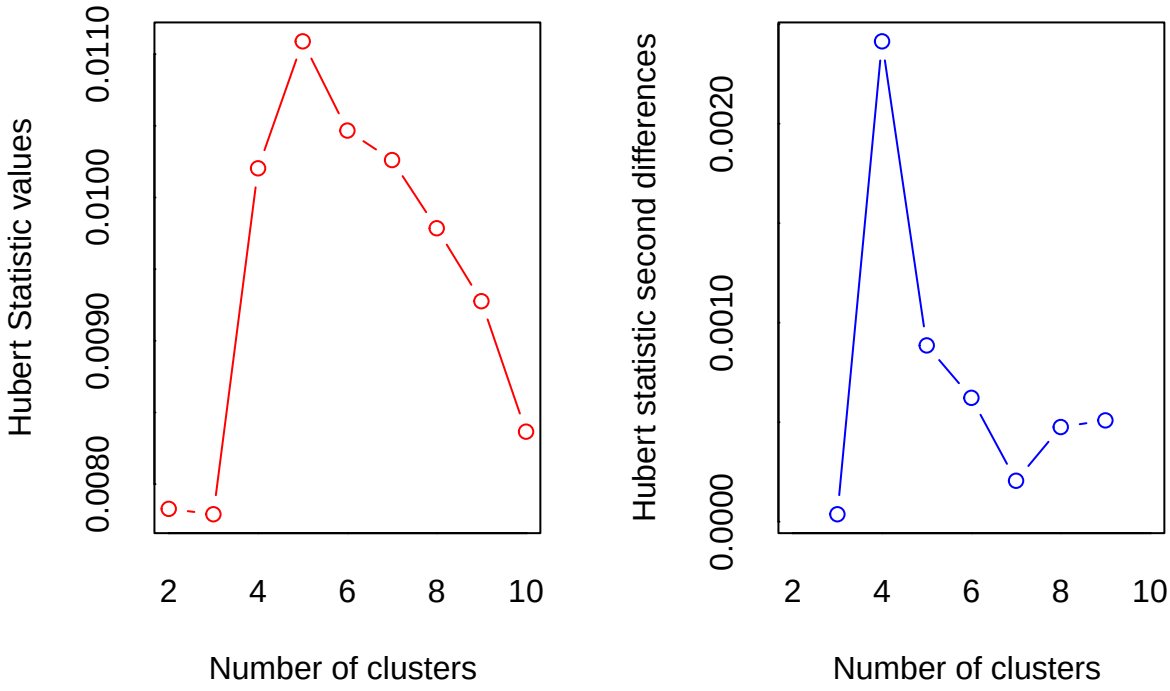


Identify number of clusters with k-mediods

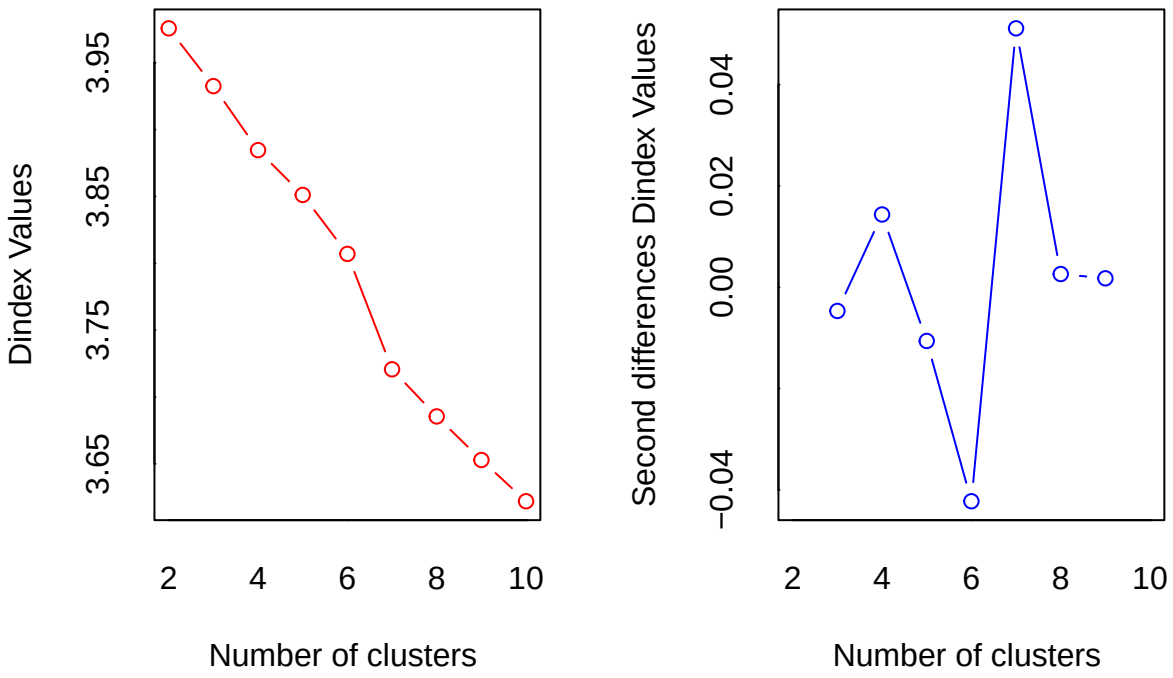
```
# Plot silhouette method
fviz_nbclust(
  x = bp_voi, # supply data
  FUNcluster = hcut, # cluster function
  hc_func = "agnes", # specify hierarchical clustering
  hc_method = "complete", # specify linkage
  diss = bp_distance, # supply distance
  method = "silhouette" # silhouette
)
```




```
# {NbClust} has over 30 different metrics to evaluate
# the number of clusters -- majority approach:
majority <- NbClust(
  data = bp_voi, # supply data
  diss = bp_distance, # supply distance
  distance = NULL, # using our own distance
  max.nc = 10, # maximum number of clusters
  method = "median", # perhaps more consistent with mediods
  index = "all" # all metrics
)
```



*** : The Hubert index is a graphical method of determining the number of clusters.
In the plot of Hubert index, we seek a significant knee that corresponds to a significant increase of the value of the measure i.e the significant peak in Hubert index second differences plot.




```
*** : The D index is a graphical method of determining the number of clusters.
                In the plot of D index, we seek a significant knee (the significant peak in Dindex
                second differences plot) that corresponds to a significant increase of the value of
                the measure.

*******************************************************************
* Among all indices:
* 9 proposed 2 as the best number of clusters
* 1 proposed 3 as the best number of clusters
* 1 proposed 4 as the best number of clusters
* 1 proposed 6 as the best number of clusters
* 10 proposed 7 as the best number of clusters
* 1 proposed 9 as the best number of clusters
* 1 proposed 10 as the best number of clusters

                    ***** Conclusion *****

* According to the majority rule, the best number of clusters is  7


*******************************************************************
```

7 is the most but 2 is provided by Silhouette and suggested by nearly as many as methods as 7. I'll proceed with 2 clusters.

Perform k-mediods

```
# Perform agglomerative clustering based on number regions
hierarchical <- agnes(
  x = bp_distance, # supply distance
  method = "complete" # linkage method
)

# Agglomerative coefficient
hierarchical$ac
```

```
[1] 0.7865813
```

```
# Get agglomerative clusters based on silhouette
silhouette_clusters <- cutree(hierarchical, k = 2)
```

The agglomerative coefficient is lower than we would like suggesting there is a lower tendency for clustering. There are no hard cut-offs but generally we'd like to see this coefficient about 0.90.

Get results

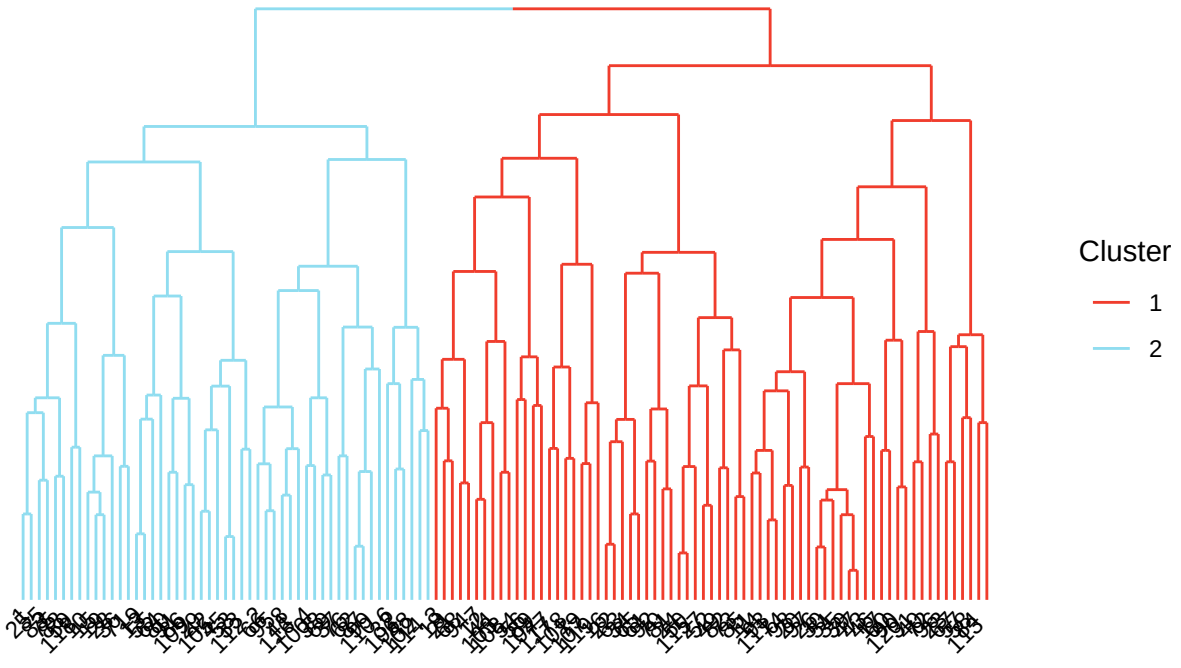
```
# Compute your own "centroids"
t(data.frame(
  "Cluster_1" = apply(bp_voi[silhouette_clusters == 1,], 2, median),
  "Cluster_2" = apply(bp_voi[silhouette_clusters == 2,], 2, median)
))
```

```
          Sadness Euphoric Exhausted Sleep.dissorder Mood.Swing
Cluster_1       3        1         3               3          0
Cluster_2       2        2         2               2          0
          Suicidal.thoughts Anorxia Authority.Respect Try.Explanation
Cluster_1                 1       0                 0               0
Cluster_2                 0       0                 0               0
          Aggressive.Response Ignore...Move.On Nervous.Break.down
Cluster_1                   0               0                  1
Cluster_2                   1               0                  0
          Admit.Mistakes Overthinking Sexual.Activity Concentration Optimisim
Cluster_1              0            1               3             4         3
Cluster_2              0            0               5             4         6
```

```
# Plot with clusters
pseudo_info <- list(
  clusters = silhouette_clusters,
  clusterTree = hierarchical,
  JSD = as.matrix(bp_distance)
)

# Set class
class(pseudo_info) <- "infoCluster"

# Rotate
plot(pseudo_info, rotate = FALSE)
```



It seems like there is a cluster with higher suicidality, mood swings, and overthinking.

Compare clusters with expert's opinion

```
# Adjusted Rand Index
compare(silhouette_clusters, expert, method = "adjusted.rand")
```

```
[1] 0.2286088
```

```
# Normalized Mutual Information
compare(silhouette_clusters, expert, method = "nmi")
```

```
[1] 0.2747622
```

There is some similarity but not much between these clusters and the expert's diagnoses. Hierarchical clustering is more similar to experts than k-mediods.