

Clustering Applied to Tennis Data

STA 325

In this lab, we'll look at tennis data. Let's read in the data and see what it looks like. Specifically, let's read in the file titled `Wimbledon.csv` into the variable `myTennisDataD`. In your `read.csv()` remember to set the `stringsAsFactors()` parameter to `FALSE`. We will remove the first two columns of the dataset, paste the names together, and replace the row numberings with these match names. (Now, let's look at the code that puts all these steps together in order to just read in the data and slightly re-shape it.)

```
# read in the data
myTennisDataD <- read.csv('Wimbledon.csv', stringsAsFactors = FALSE)

# remove named columns
myTennisDataC <- myTennisDataD[,-1:-2]

# rename rows with removed columns
row.names(myTennisDataC) <- as.character(apply(myTennisDataD[,1:2], 1,
  paste, collapse = " "))
```

Lab Tasks

1. There are two columns in your dataset that contain only NA values. Remove these. For all other columns that contain missing values, replace the missing values with the median values for that column. Store this cleaned dataset into the variable `iFinalTennisData`.
2. Create a log-Euclidean distance matrix for your data and perform single linkage and complete linkage clustering on the data and store these in the variables `singeLinkage` and `completeLinkage` respectively.
3. Using the command `intCriteria()` in the `clusterCrit` package, write a function that takes as its inputs the number of clusters to be tested (say 10), a clustered object (e.g `SingleLinkage` from above) and the original data frame (e.g. `iFinalTennisData`), and computes the CH Index for each assumed number of clusters and creates a plot of the resulting CH indices for each cluster count with a dotted vertical line indicating the maximum value. Also return the maximum CH Index computed. Test this function for up to 10 clusters with both single and complete linkage on `iFinalTennisData`, and include the maximum CH Index calculated and the related plots. What should the CH Index value be when the number of clusters is 1 and why?

Hint:

```
# computeCH <- function(num.clust, clust.obj, data.set){  
  
#}  
##### #  
  
#This function creates a plot of CH indices  
# and returns the largest CH index  
# when the number of clusters is less than a given value  
#  
# Inputs:  
#   num.clust:  a positive integer,  
#               will test all cluster numbers up to this  
#   clust.obj:  a cluster object that can be cut into various  
#               clusters  
#   data.set:  a data frame containing data that generated clust.obj  
#  
# Output:  
#   A plot of the CH indices  
#   The largest CH index for # of clusters below num.clust
```

4. For each clustering (based on linkage) what is the optimal number of clusters?
5. Create a dendrogram for each of your two clustering assignments. Based on the data type, what is the most appropriate type of clustering and why?