Assignment 4

2023-02-14

### Load Packages Needed for Both Questions

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.4.1 ✔ purrr 1.0.1  
## ✔ tibble 3.1.8 ✔ dplyr 1.1.0  
## ✔ tidyr 1.3.0 ✔ stringr 1.5.0  
## ✔ readr 2.1.4 ✔ forcats 1.0.0  
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(janitor)

##   
## Attaching package: 'janitor'  
##   
## The following objects are masked from 'package:stats':  
##   
## chisq.test, fisher.test

library(ggbiplot)

## Loading required package: plyr  
## ------------------------------------------------------------------------------  
## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)  
## ------------------------------------------------------------------------------  
##   
## Attaching package: 'plyr'  
##   
## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize  
##   
## The following object is masked from 'package:purrr':  
##   
## compact  
##   
## Loading required package: scales  
##   
## Attaching package: 'scales'  
##   
## The following object is masked from 'package:purrr':  
##   
## discard  
##   
## The following object is masked from 'package:readr':  
##   
## col\_factor  
##   
## Loading required package: grid

library(stats)  
library(factoextra)

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

library(cluster)  
library(caret)

## Loading required package: lattice  
##   
## Attaching package: 'caret'  
##   
## The following object is masked from 'package:purrr':  
##   
## lift

## Part 1: Implementing a Simple Prediction Pipeline

The New York City Department of Health administered a questionnaire on general health and physical activity among residents. Using the dataset class4\_p1.csv, fit and evaluate two prediction models using linear regression. The aim of the models is to predict the number of days in a month an individual reported having good physical health (feature name: healthydays). A codebook is provided so you can look-up the meaning and values of each feature in the dataset. (Note the codebook lists information on features that are not included in your dataset).

Your analytic pipeline should include the following:

* Perform basic data cleaning. Note which features are continuous, which are categorical and ensure they are being stored that way in your R dataset (That is, if categorical variables have been read-in as continuous variables, convert them to factors)
* Partition data into training and testing (use a 70/30 split)
* Fit two prediction models using different subsets of the features in the training data. Features can overlap in the two models, but the feature sets should not be exactly the same across models. Clearly state which features were used in the two models.
* Apply both models within the test data and determine which model is the preferred prediction model using the appropriate evaluation metric(s).
* Describe one setting (in 1-2 sentences) where the implementation of your final model would be useful.

### Step 1: Load and Prepare Dataset

set.seed(123)  
  
#Loading in Data  
a4.data<-read.csv("~/Desktop/Nimish/Columbia MPH/4. Spring 2023/Machine Learning in Epi/Assignments/Assignment 4/class4\_p1.csv")  
  
#Rename Dataset based on the code book  
a4.data <- a4.data %>%   
 dplyr::rename(hypertension = chronic1,  
 diabetes = chronic3,  
 asthma = chronic4,  
 cig\_use = tobacco1,  
 alc\_drink = alcohol1,  
 pa\_min = gpaq8totmin,  
 walk\_days = gpaq11days,  
 pa\_yn = habits5,  
 diet = habits7,  
 gender = dem3,  
 hispanic = dem4,  
 us\_citizen = dem8)  
  
#Convert all categorical variables to factors   
a4.data <- a4.data %>%   
 mutate(hypertension = as.factor(hypertension),  
 diabetes = as.factor(diabetes),  
 asthma = as.factor(asthma),  
 cig\_use = as.factor(cig\_use),  
 alc\_drink = as.factor(alc\_drink),  
 pa\_yn = as.factor(pa\_yn),  
 agegroup = as.factor(agegroup),  
 diet = as.factor(diet),  
 gender = as.factor(gender),  
 hispanic = as.factor(hispanic),  
 us\_citizen = as.factor(us\_citizen),  
 povertygroup = as.factor(povertygroup))  
  
#Strip ID  
a4.data$X <- NULL  
  
#Remove Missing Variables  
a4.data <- na.omit(a4.data)  
  
#Check Data Structure  
str(a4.data)

## 'data.frame': 2195 obs. of 16 variables:  
## $ hypertension: Factor w/ 2 levels "1","2": 2 2 1 2 1 2 2 2 2 2 ...  
## $ diabetes : Factor w/ 2 levels "1","2": 2 2 1 2 1 2 2 2 2 2 ...  
## $ asthma : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 1 ...  
## $ bmi : num 19.9 33.7 24.2 22.7 25.7 ...  
## $ cig\_use : Factor w/ 3 levels "1","2","3": 3 3 3 2 3 3 3 3 3 3 ...  
## $ alc\_drink : Factor w/ 3 levels "1","2","3": 3 3 3 2 3 2 3 3 2 3 ...  
## $ pa\_min : int 0 0 30 0 0 30 0 0 0 0 ...  
## $ walk\_days : int 7 7 4 6 4 7 5 0 7 7 ...  
## $ pa\_yn : Factor w/ 4 levels "1","2","3","4": 2 2 2 2 2 2 1 2 2 3 ...  
## $ diet : Factor w/ 5 levels "1","2","3","4",..: 2 5 3 4 4 5 2 2 3 4 ...  
## $ agegroup : Factor w/ 4 levels "1","2","3","4": 2 2 2 2 3 1 2 2 2 3 ...  
## $ gender : Factor w/ 2 levels "1","2": 1 2 2 1 2 1 2 2 1 2 ...  
## $ hispanic : Factor w/ 2 levels "1","2": 2 1 2 2 2 2 2 2 1 1 ...  
## $ us\_citizen : Factor w/ 2 levels "1","2": 2 2 2 1 2 2 1 1 1 2 ...  
## $ povertygroup: Factor w/ 6 levels "1","2","3","4",..: 1 1 1 5 2 6 4 3 2 2 ...  
## $ healthydays : int 30 27 30 30 23 30 30 30 30 0 ...  
## - attr(\*, "na.action")= 'omit' Named int [1:1616] 1 2 3 4 5 6 7 8 9 10 ...  
## ..- attr(\*, "names")= chr [1:1616] "1" "2" "3" "4" ...

#Finding correlated predictors  
a4.data.numeric <- a4.data %>% dplyr::select(where(is.numeric))  
correlations<-cor(a4.data.numeric, use="complete.obs")  
high.correlations<-findCorrelation(correlations, cutoff=0.4) #None Found  
  
#Centering and Scaling  
set.up.preprocess<-preProcess(a4.data, method=c("center", "scale"))  
  
#Output pre-processed values  
transformed.vals<-predict(set.up.preprocess, a4.data)

### Step 2: Data Partitioning

set.seed(123)  
  
train.index<-createDataPartition(transformed.vals$healthydays, p=0.7, list=FALSE)  
  
a4.data.train<-transformed.vals[train.index,]  
a4.data.test<-transformed.vals[-train.index,]  
  
  
#Construct k-folds in your data  
train.folds<-createFolds(transformed.vals$healthydays, k=10, list=FALSE)

### Step 3: Training the Model using a Linear Regression Method

set.seed(123)  
  
#Model 1: Comorbidities and Some Modifiable Factors  
model1 <- lm(healthydays ~ hypertension + diabetes + bmi + alc\_drink + cig\_use, data = a4.data.train)  
  
#Model 2: Modifiable Factors and Demographic  
model2 <- lm(healthydays ~ cig\_use + alc\_drink + diet + agegroup + gender + povertygroup + bmi,  
 data = a4.data.train)

### Step 4: Testing the Model

set.seed(123)  
  
#Applying the Predictions  
prediction1 <- predict(model1, a4.data.test)  
prediction2 <- predict(model2, a4.data.test)

### Step 5: Model Evaluation

set.seed(123)  
  
#Checking the Mean Square Errors  
rmse1 <- RMSE(prediction1, a4.data.test$healthydays)  
rmse2 <- RMSE(prediction2, a4.data.test$healthydays)

The Root Mean Squared Error for Model 1 is **0.9693983**. The Root Mean Squared Error for Model 2 is **0.9523164**.

Since the Mean Squared Error for Model 2 is smaller, the preferred model is **Model 2.**

**Describe one setting (in 1-2 sentences) where the implementation of your final model would be useful.**

This model would be useful in a research setting where health is measured as a function of “healthy days” and we want to study the effect of modifiable factors such as cigarette use, alcohol habits, and diet on overall health. The model would also account for how sociodemographic variables affect the health of patients.

## Part 2: Conducting an Unsupervised Analysis

Using the dataset from the Group assignment Part 3 (USArrests), identify clusters using hierarchical analysis. Use an agglomerative algorithm for hierarchical clustering. Use a Euclidian distance measure to construct your dissimilarity matrix.

Conduct a hierarchical clustering analysis. Be sure to specify the linkage method used. Within your analysis, make sure you do both of the following:

* Determine the optimal number of clusters using a clear, data-driven strategy.
* Describe the composition of each cluster in terms of the original input features.

Pretend that the data are from 2020 and not 1973. Describe one research question that can be addressed using the newly identified clusters. Briefly comment on any scientific or ethical considerations one should review before using these clusters for your specific question. NOTE: The clusters can be used as an exposure, an outcome or a covariate.

### Step 1: Load data and prepare for analysis

set.seed(123)  
  
#Load in the Data  
arrest.data <- USArrests  
  
#Deleting any missing data  
arrest.data<-na.omit(arrest.data)  
  
#Check means and SDs to determine if scaling is necessary  
colMeans(arrest.data, na.rm=TRUE)

## Murder Assault UrbanPop Rape   
## 7.788 170.760 65.540 21.232

apply(arrest.data, 2, sd, na.rm=TRUE)

## Murder Assault UrbanPop Rape   
## 4.355510 83.337661 14.474763 9.366385

#Is scaling necessary? Yes, scaling is necessary.  
#Centering and Scaling  
set.up.preprocess2 <- preProcess(arrest.data, method=c("center", "scale"))  
  
#Output pre-processed values  
arrest.transformed <- predict(set.up.preprocess2, arrest.data)  
  
#Check means and SDs to check if scaling worked  
colMeans(arrest.transformed, na.rm=TRUE)

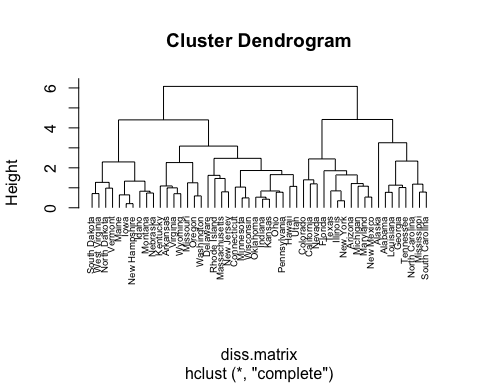
## Murder Assault UrbanPop Rape   
## -7.271961e-17 1.115774e-16 -4.074519e-16 6.661338e-17

apply(arrest.transformed, 2, sd, na.rm=TRUE)

## Murder Assault UrbanPop Rape   
## 1 1 1 1

### Step 2: Conduct a hierarchical clustering analysis

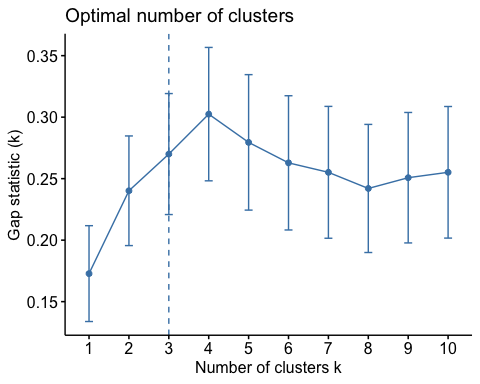
set.seed(123)  
  
# Create Dissimilarity matrix  
diss.matrix <- dist(arrest.transformed, method = "euclidean")  
  
# Hierarchical clustering using Complete Linkage  
clusters.h<- hclust(diss.matrix, method = "complete")  
  
# Plot the obtained dendrogram  
plot(clusters.h, cex = 0.6, hang = -1)



gap\_stat2 <- clusGap(arrest.transformed, FUN = hcut, K.max = 10, B = 50)  
gap\_stat2

## Clustering Gap statistic ["clusGap"] from call:  
## clusGap(x = arrest.transformed, FUNcluster = hcut, K.max = 10, B = 50)  
## B=50 simulated reference sets, k = 1..10; spaceH0="scaledPCA"  
## --> Number of clusters (method 'firstSEmax', SE.factor=1): 3  
## logW E.logW gap SE.sim  
## [1,] 3.458369 3.631167 0.1727975 0.03896440  
## [2,] 3.142869 3.383016 0.2401470 0.04457094  
## [3,] 2.978518 3.248506 0.2699880 0.04918469  
## [4,] 2.833430 3.135886 0.3024565 0.05422817  
## [5,] 2.759739 3.039231 0.2794925 0.05507099  
## [6,] 2.689377 2.952214 0.2628371 0.05456848  
## [7,] 2.616505 2.871649 0.2551443 0.05360321  
## [8,] 2.556388 2.798408 0.2420191 0.05207751  
## [9,] 2.478796 2.729565 0.2507688 0.05303471  
## [10,] 2.408612 2.663773 0.2551613 0.05349496

fviz\_gap\_stat(gap\_stat2)



**Determine the optimal number of clusters using a clear, data-driven strategy.**

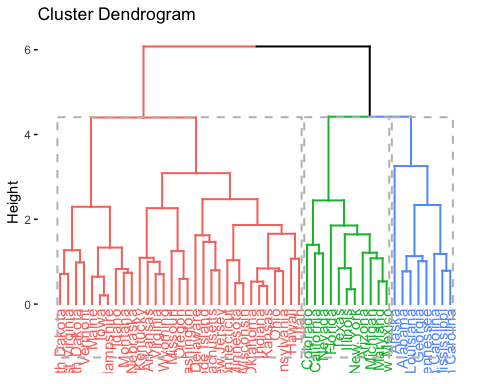
* Using *hierarchical clustering*, we determined that the optimal number of clusters is **3 Clusters.**

set.seed(123)  
  
#Create 3 clusters  
clusters.hcut<-hcut(arrest.transformed, k=3, hc\_func="hclust", hc\_method="complete", hc\_metric="euclidian")  
  
#Output size, dendrogram, and plot of the 3 clusters  
clusters.hcut$size

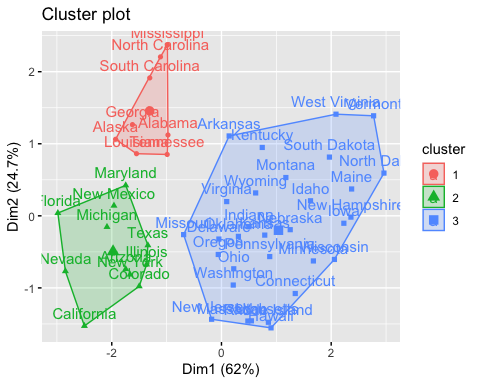
## [1] 8 11 31

fviz\_dend(clusters.hcut, rect=TRUE)

## Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" instead as  
## of ggplot2 3.3.4.  
## ℹ The deprecated feature was likely used in the factoextra package.  
## Please report the issue at <]8;;https://github.com/kassambara/factoextra/issueshttps://github.com/kassambara/factoextra/issues]8;;>.



fviz\_cluster(clusters.hcut)



input.feature.vals<-cbind(arrest.transformed,cluster=clusters.hcut$cluster)  
  
cluster\_info <- input.feature.vals %>%  
 group\_by(cluster) %>%  
 summarise\_all(mean)  
  
#Cluster Table  
knitr::kable(cluster\_info)

| cluster | Murder | Assault | UrbanPop | Rape |
| --- | --- | --- | --- | --- |
| 1 | 1.4463290 | 0.9838289 | -0.8317925 | 0.3529110 |
| 2 | 0.7499801 | 1.1199128 | 0.9361748 | 1.2156432 |
| 3 | -0.6393681 | -0.6512797 | -0.1175349 | -0.5224311 |

**Describe the composition of each cluster in terms of the original input features**

Based on a hierarchical clustering analysis, we determined that there are 3 optimal clusters based on the features in the dataset USArrests:

* Cluster 1 is composed of **8 states** that have **higher** mean rates of arrest per 10,000 people for murder, assault, and rape than the national average rates, but a **lower** than national average urban population.
* Cluster 2 is composed of **11 states** that have **higher** mean rates of arrest per 10,000 people for murder, assault, and rape than the national average rates, as well as a **higher** than national average urban population.
* Cluster 3 is composed of **31 states** that have **lower** mean rates of arrest per 10,000 people for murder, assault, and rape than the national average rates, as well as a **lower** than national average urban population.

**Pretend that the data are from 2020 and not 1973. Describe one research question that can be addressed using the newly identified clusters. Briefly comment on any scientific or ethical considerations one should review before using these clusters for your specific question. NOTE: The clusters can be used as an exposure, an outcome or a covariate.**

Using a cluster analysis, we can find unique clusters on different levels and combinations of arrest rates for violent crimes in the United States. By studying the underlying similarities in these clusters, we can answer the research question, **“How do sociodemographic factors such as poverty, education, race and ethnic composition, political composition, and urban composition affect affect the arrest rates for violent crimes in the United States in 2020?”**

Based on previous discussions in this class, we must ensure that we are working on data that is representative, and has been collected without methods that introduce bias. Secondly, clustering methodology should be used to further identify and study the factors that have led to the formation of these clusters. The algorithm should not be misinterpreted and misused to predict states with high arrest rates or to indirectly target policies against vulnerable communities in specific high-crime states.