# Section 2 (Week 10)

Loading the required libraries.

library(tidyr)  
library(dplyr)  
library(ggplot2)  
library(broom)  
library(randomForest)

# Q1: Data importing and cleaning steps are explained in the text and in the DataCamp exercises.(Tell me why you are doing the data cleaning activities that you perform). Follow a logical process.

## Explaining the steps on how to import and clean my data

### A brief overview about the source data

The actual database contains 76 attributes, but all published experiments refer to using a subset of 14 of them. In particular, the Cleveland database is the only one that has been used by ML researchers to this date. But we would be using all the other datasets also. The “goal” field refers to the presence of heart disease in the patient. It is integer valued from 0 (no presence) to 4. Experiments with the Cleveland database have concentrated on simply attempting to distinguish presence (values 1,2,3,4) from absence (value 0).

The names and social security numbers of the patients were removed from the database, replaced with dummy values. We are using all four processed files which also exist in the dataset directory.

### Summary of the various datasets are given below.

Database: # of instances:  
 Cleveland: 303  
 Hungarian: 294  
 Switzerland: 123  
 Long Beach VA: 200

### Attribute Information: The attributes that are defined in the below datasets are defined here.

It also shows the position of the attributes in the actual files:

Only 14 used:

V1. #3 (age) : Age in years  
V2. #4 (sex) : sex (1 = male; 0 = female)

V3. #9 (cp) : chest pain type (1:typical angina, 2:atypical angina, 3:non-anginal pain, 4: asymptomatic)

V4. #10 (trestbps) : resting blood pressure (in mm Hg on admission to the hospital)

V5. #12 (chol) : serum cholestoral in mg/dl

V6. #16 (fbs) : (fasting blood sugar > 120 mg/dl) (1 = true; 0 = false)

V7. #19 (restecg) : resting electrocardiographic results 0: normal, 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV) 2: showing probable or definite left ventricular hypertrophy by Estes’ criteria

V8. #32 (thalach) : maximum heart rate achieved

V9. #38 (exang) : exercise induced angina (1 = yes; 0 = no)

V10. #40 (oldpeak) : ST depression induced by exercise relative to rest  
V11. #41 (slope) : the slope of the peak exercise ST segment 1: upsloping 2: flat 3: downsloping

V12. #44 (ca) : number of major vessels (0-3) colored by flourosopy

V13. #51 (thal) : 3 = normal; 6 = fixed defect; 7 = reversable defect  
V14. #58 (num) : the predicted attribute diagnosis of heart disease (angiographic disease status) 0: < 50% diameter narrowing (No heart disease) 1: > 50% diameter narrowing ( Yes Heart disease)

### Explanation for why clean up is necessary

Before we apply any algorithm on our data, it is obvious that the data should be tidy or structured. But in the real world, the data mostly we initially see is unstructured. So in order to make it tidy and to further apply any algorithm to derive the insights, data has to be cleaned. The major reason why the data is not tidy is because of the presence of missing values and outliers. So as a first step I would start by importing each of the 4 files which have already being processed, into 4 different data frames and looking at those separately in order to get a better understanding of the values of the attributes and how they are distributed.

Given below is the detailed step-by-step process of cleaning the data.

#populating the cleveland file into dataframe  
wd <- getwd()  
cleveland\_f <- "processed.cleveland.data"  
path\_to\_file <- paste(wd,'/dataset/',cleveland\_f, sep = "")  
## path\_to\_file  
  
cleveland\_df <- read.csv(path\_to\_file, header = FALSE)  
## dim(cleveland\_df)  
## summary(cleveland\_df)  
## head(cleveland\_df)  
  
#populating the Hungarian file into dataframe  
hungarian\_f <- "processed.hungarian.data"  
path\_to\_file <- paste(wd,'/dataset/',hungarian\_f, sep = "")  
## path\_to\_file  
  
hungarian\_df <- read.csv(path\_to\_file, header = FALSE)  
## dim(hungarian\_df)  
## summary(hungarian\_df)  
## head(hungarian\_df)  
  
#populating the Switzerland file into dataframe  
switzerland\_f <- "processed.switzerland.data"  
path\_to\_file <- paste(wd,'/dataset/',switzerland\_f, sep = "")  
## path\_to\_file  
  
switzerland\_df <- read.csv(path\_to\_file, header = FALSE)  
## dim(switzerland\_df)  
## summary(switzerland\_df)  
## head(switzerland\_df)  
  
#populating the Long Beach, CA data file into dataframe  
long\_beach\_f <- "processed.va.data"  
path\_to\_file <- paste(wd,'/dataset/',long\_beach\_f, sep = "")  
## path\_to\_file  
  
long\_beach\_df <- read.csv(path\_to\_file, header = FALSE)  
## dim(long\_beach\_df)  
## summary(long\_beach\_df)  
## head(long\_beach\_df)  
  
dim(cleveland\_df)

[1] 303 14

dim(hungarian\_df)

[1] 294 14

dim(switzerland\_df)

[1] 123 14

dim(long\_beach\_df)

[1] 200 14

After having a initial look at the datasets, now adding the 4 individual dataframes into separate data datasets and also adding the column names for each dataset in order to do some more analysis on each of those, as we know that the data do not have the column names.

# Creating new data frames with the existing dataframe towards our final dataset.  
data.cle <- cleveland\_df  
data.hun <- hungarian\_df  
data.sw <- switzerland\_df  
data.va <- long\_beach\_df

### Adding the new columns for each of the data source name.

Also adding an extra variable to keep the Source name along with the data set. Now looking at the data briefly. This would help in the model building phase after I combine all the datasets.

names(data.cle) <- c("age", "sex", "cp", "trestbps", "chol", "fbs", "restecg", "thalach", "exang", "oldpeak", "slope", "ca", "thal", "out")  
names(data.hun) <- c("age", "sex", "cp", "trestbps", "chol", "fbs", "restecg", "thalach", "exang", "oldpeak", "slope", "ca", "thal", "out")  
names(data.sw) <- c("age", "sex", "cp", "trestbps", "chol", "fbs", "restecg", "thalach", "exang", "oldpeak", "slope", "ca", "thal", "out")  
names(data.va) <- c("age", "sex", "cp", "trestbps", "chol", "fbs", "restecg", "thalach", "exang", "oldpeak", "slope", "ca", "thal", "out")  
  
# Adding the extra column to identify the source for the file  
data1 <- data.cle %>% mutate(datasrc = "Cleveland")   
data2 <- data.hun %>% mutate(datasrc = "Hungarian")   
data3 <- data.sw %>% mutate(datasrc = "Switzerland")   
data4 <- data.va %>% mutate(datasrc = "Long Beach VA")  
  
head(data1)  
head(data2)  
head(data3)  
head(data4)

We are trying to clean up the dataset individually.

### Working on cleaning and tidying the Cleveland Dataset

Here for our problem, we are only going to attempt to distinguish the presence of heart disease (values 1,2,3,4) from absence of heart disease (value 0). Therefore, we replace all labels greater than 1 by 1. Then taking summary of each of the data sets. Showing below our findings on each of the various datasets.

data1$out[data1$out > 1] <- 1  
summary(data1)  
glimpse(data1)

We see that the data set is showing mean for categorical variables also. Hence we need to re-specify the column types. We know a categorical variable (a variable that takes on a finite amount of values) is a factor. As we can see, sex is incorrectly treated as a number when in reality it can only be 1 if male and 0 if female. We can use the transform method to change the in built type of each feature.

data1 <- transform(  
 data1,  
 age=as.integer(age),  
 sex=as.factor(sex),  
 cp=as.factor(cp),  
 trestbps=as.integer(trestbps),  
 chol=as.integer(chol),  
 fbs=as.factor(fbs),  
 restecg=as.factor(restecg),  
 thalach=as.integer(thalach),  
 exang=as.factor(exang),  
 oldpeak=as.numeric(oldpeak),  
 slope=as.factor(slope),  
 ca=as.factor(ca),  
 thal=as.factor(thal),  
 out=as.factor(out),  
 datasrc=as.character(datasrc)  
)  
summary(data1)

## age sex cp trestbps chol fbs   
## Min. :29.00 0: 97 1: 23 Min. : 94.0 Min. :126.0 0:258   
## 1st Qu.:48.00 1:206 2: 50 1st Qu.:120.0 1st Qu.:211.0 1: 45   
## Median :56.00 3: 86 Median :130.0 Median :241.0   
## Mean :54.44 4:144 Mean :131.7 Mean :246.7   
## 3rd Qu.:61.00 3rd Qu.:140.0 3rd Qu.:275.0   
## Max. :77.00 Max. :200.0 Max. :564.0   
## restecg thalach exang oldpeak slope ca thal   
## 0:151 Min. : 71.0 0:204 Min. :0.00 1:142 ? : 4 ? : 2   
## 1: 4 1st Qu.:133.5 1: 99 1st Qu.:0.00 2:140 0.0:176 3.0:166   
## 2:148 Median :153.0 Median :0.80 3: 21 1.0: 65 6.0: 18   
## Mean :149.6 Mean :1.04 2.0: 38 7.0:117   
## 3rd Qu.:166.0 3rd Qu.:1.60 3.0: 20   
## Max. :202.0 Max. :6.20   
## out datasrc   
## 0:164 Length:303   
## 1:139 Class :character   
## Mode :character   
##   
##   
##

glimpse(data1)

## Observations: 303  
## Variables: 15  
## $ age <int> 63, 67, 67, 37, 41, 56, 62, 57, 63, 53, 57, 56, 56, 44, 52, …  
## $ sex <fct> 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, …  
## $ cp <fct> 1, 4, 4, 3, 2, 2, 4, 4, 4, 4, 4, 2, 3, 2, 3, 3, 2, 4, 3, 2, …  
## $ trestbps <int> 145, 160, 120, 130, 130, 120, 140, 120, 130, 140, 140, 140, …  
## $ chol <int> 233, 286, 229, 250, 204, 236, 268, 354, 254, 203, 192, 294, …  
## $ fbs <fct> 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, …  
## $ restecg <fct> 2, 2, 2, 0, 2, 0, 2, 0, 2, 2, 0, 2, 2, 0, 0, 0, 0, 0, 0, 0, …  
## $ thalach <int> 150, 108, 129, 187, 172, 178, 160, 163, 147, 155, 148, 153, …  
## $ exang <fct> 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, …  
## $ oldpeak <dbl> 2.3, 1.5, 2.6, 3.5, 1.4, 0.8, 3.6, 0.6, 1.4, 3.1, 0.4, 1.3, …  
## $ slope <fct> 3, 2, 2, 3, 1, 1, 3, 1, 2, 3, 2, 2, 2, 1, 1, 1, 3, 1, 1, 1, …  
## $ ca <fct> 0.0, 3.0, 2.0, 0.0, 0.0, 0.0, 2.0, 0.0, 1.0, 0.0, 0.0, 0.0, …  
## $ thal <fct> 6.0, 3.0, 7.0, 3.0, 3.0, 3.0, 3.0, 3.0, 7.0, 7.0, 6.0, 3.0, …  
## $ out <fct> 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, …  
## $ datasrc <chr> "Cleveland", "Cleveland", "Cleveland", "Cleveland", "Clevela…

Now also we find that there are missing values for some of the variables in each dataset. Here we see missing/unknown values for the below variables. ca : number of major vessels (0-3) colored by flourosopy thal : 3 = normal; 6 = fixed defect; 7 = reversable defect  
Now after replacing the missing values (“?”) with NA we can use the colSums function to count the number of missing values. It shows that we have 4 and 2 rows with missing values for thal and ca respectively.

data1[ data1 == "?"] <- NA  
colSums(is.na(data1))

age sex cp trestbps chol fbs restecg thalach   
 0 0 0 0 0 0 0 0

exang oldpeak slope ca thal out datasrc 0 0 0 4 2 0 0 Let us have a look at those 6 rows and see if we could find anything interesting in those rows.

data1 %>% filter(is.na(ca) | is.na(thal))

age sex cp trestbps chol fbs restecg thalach exang oldpeak slope ca thal 1 53 0 3 128 216 0 2 115 0 0.0 1 0.0 2 52 1 3 138 223 0 0 169 0 0.0 1 3.0 3 43 1 4 132 247 1 2 143 1 0.1 2 7.0 4 52 1 4 128 204 1 0 156 1 1.0 2 0.0 5 58 1 2 125 220 0 0 144 0 0.4 2 7.0 6 38 1 3 138 175 0 0 173 0 0.0 1 3.0 out datasrc 1 0 Cleveland 2 0 Cleveland 3 1 Cleveland 4 1 Cleveland 5 0 Cleveland 6 0 Cleveland Now as we are not experts in this type of data let us take these 6 rows out from the cleveland dataset.Now looking at the new cleaner dataset we see there are no missing records now and the count of the dataset is 303 - 6 = 296. This is only being done on the Cleveland dataset as that seems to be missing the least number of variables. We would use this dataset set separately as our final data set for prediction purposes.

data11 <- data1 %>% filter(!is.na(ca)) %>% filter(!is.na(thal))  
colSums(is.na(data11))

## age sex cp trestbps chol fbs restecg thalach   
## 0 0 0 0 0 0 0 0   
## exang oldpeak slope ca thal out datasrc   
## 0 0 0 0 0 0 0

# Recasting the above 2 columns as factor shows the summary info correctly  
data11$ca <- factor(data11$ca)  
data11$thal <- factor(data11$thal)  
summary(data11)

## age sex cp trestbps chol fbs   
## Min. :29.00 0: 96 1: 23 Min. : 94.0 Min. :126.0 0:254   
## 1st Qu.:48.00 1:201 2: 49 1st Qu.:120.0 1st Qu.:211.0 1: 43   
## Median :56.00 3: 83 Median :130.0 Median :243.0   
## Mean :54.54 4:142 Mean :131.7 Mean :247.4   
## 3rd Qu.:61.00 3rd Qu.:140.0 3rd Qu.:276.0   
## Max. :77.00 Max. :200.0 Max. :564.0   
## restecg thalach exang oldpeak slope ca thal   
## 0:147 Min. : 71.0 0:200 Min. :0.000 1:139 0.0:174 3.0:164   
## 1: 4 1st Qu.:133.0 1: 97 1st Qu.:0.000 2:137 1.0: 65 6.0: 18   
## 2:146 Median :153.0 Median :0.800 3: 21 2.0: 38 7.0:115   
## Mean :149.6 Mean :1.056 3.0: 20   
## 3rd Qu.:166.0 3rd Qu.:1.600   
## Max. :202.0 Max. :6.200   
## out datasrc   
## 0:160 Length:297   
## 1:137 Class :character   
## Mode :character   
##   
##   
##

### Repeating the same above processes on the remaining dataset and coming up with a clean/tidy dataset

## Hungarian Dataset  
data2$out[data2$out > 1] <- 1  
## summary(data2)  
## glimpse(data2)  
  
data2 <- transform(  
 data2,  
 age=as.integer(age),  
 sex=as.factor(sex),  
 cp=as.factor(cp),  
 trestbps=as.integer(trestbps),  
 chol=as.integer(chol),  
 fbs=as.factor(fbs),  
 restecg=as.factor(restecg),  
 thalach=as.integer(thalach),  
 exang=as.factor(exang),  
 oldpeak=as.numeric(oldpeak),  
 slope=as.factor(slope),  
 ca=as.factor(ca),  
 thal=as.factor(thal),  
 out=as.factor(out),  
 datasrc=as.character(datasrc)  
)  
## summary(data2)  
## glimpse(data2)  
  
data2[ data2 == "?"] <- NA  
colSums(is.na(data2))

## Swiss Dataset  
data3$out[data3$out > 1] <- 1  
## summary(data2)  
## glimpse(data2)  
  
data3 <- transform(  
 data3,  
 age=as.integer(age),  
 sex=as.factor(sex),  
 cp=as.factor(cp),  
 trestbps=as.integer(trestbps),  
 chol=as.integer(chol),  
 fbs=as.factor(fbs),  
 restecg=as.factor(restecg),  
 thalach=as.integer(thalach),  
 exang=as.factor(exang),  
 oldpeak=as.numeric(oldpeak),  
 slope=as.factor(slope),  
 ca=as.factor(ca),  
 thal=as.factor(thal),  
 out=as.factor(out),  
 datasrc=as.character(datasrc)  
)  
## summary(data2)  
## glimpse(data2)  
  
data3[ data3 == "?"] <- NA  
colSums(is.na(data3))

## Long Beach, VA Dataset  
data4$out[data4$out > 1] <- 1  
## summary(data2)  
## glimpse(data2)  
  
data4 <- transform(  
 data4,  
 age=as.integer(age),  
 sex=as.factor(sex),  
 cp=as.factor(cp),  
 trestbps=as.integer(trestbps),  
 chol=as.integer(chol),  
 fbs=as.factor(fbs),  
 restecg=as.factor(restecg),  
 thalach=as.integer(thalach),  
 exang=as.factor(exang),  
 oldpeak=as.numeric(oldpeak),  
 slope=as.factor(slope),  
 ca=as.factor(ca),  
 thal=as.factor(thal),  
 out=as.factor(out),  
 datasrc=as.character(datasrc)  
)  
## summary(data2)  
## glimpse(data2)  
  
data4[ data4 == "?"] <- NA  
colSums(is.na(data4))

### Combining the dataframes together now.

This is a combined data set with all the 4 datasets combined into one big dataset.But after looking at the various data sets it became clear that except for the Cleveland dataset all others do have a lot of missing data for some of the variables. Showing the summary on the combined dataset it looks like that the dataset from Long Beach VA and Swiss has a significantly higher number of outcomes that shows more patients having heart disease(out = 1) than without (out = 0).

combined.data <- rbind(data1,data2,data3,data4)  
  
combined.data %>% group\_by(datasrc,out) %>% summarise(n())  
## glimpse(combined.data)  
## head(combined.data)  
colSums(is.na(combined.data))

## What does the final data set look like

With a clean dataset, show what the final data set looks like. However, do not print off a data frame with 200+ rows; show me the data in the most condensed form possible.

The final data set is presented below after analyzing all the different available datasets. So after doing the initial round of Exploratory Data Analysis, I am planning on using the Clean and Tidy Cleveland data set for my predictions. I will name it the final\_data. The summary detail is shown below. This dataset do not have any missing values or outliers.

final\_data <- data11  
  
glimpse(final\_data)

## Observations: 297  
## Variables: 15  
## $ age <int> 63, 67, 67, 37, 41, 56, 62, 57, 63, 53, 57, 56, 56, 44, 52, …  
## $ sex <fct> 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, …  
## $ cp <fct> 1, 4, 4, 3, 2, 2, 4, 4, 4, 4, 4, 2, 3, 2, 3, 3, 2, 4, 3, 2, …  
## $ trestbps <int> 145, 160, 120, 130, 130, 120, 140, 120, 130, 140, 140, 140, …  
## $ chol <int> 233, 286, 229, 250, 204, 236, 268, 354, 254, 203, 192, 294, …  
## $ fbs <fct> 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, …  
## $ restecg <fct> 2, 2, 2, 0, 2, 0, 2, 0, 2, 2, 0, 2, 2, 0, 0, 0, 0, 0, 0, 0, …  
## $ thalach <int> 150, 108, 129, 187, 172, 178, 160, 163, 147, 155, 148, 153, …  
## $ exang <fct> 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, …  
## $ oldpeak <dbl> 2.3, 1.5, 2.6, 3.5, 1.4, 0.8, 3.6, 0.6, 1.4, 3.1, 0.4, 1.3, …  
## $ slope <fct> 3, 2, 2, 3, 1, 1, 3, 1, 2, 3, 2, 2, 2, 1, 1, 1, 3, 1, 1, 1, …  
## $ ca <fct> 0.0, 3.0, 2.0, 0.0, 0.0, 0.0, 2.0, 0.0, 1.0, 0.0, 0.0, 0.0, …  
## $ thal <fct> 6.0, 3.0, 7.0, 3.0, 3.0, 3.0, 3.0, 3.0, 7.0, 7.0, 6.0, 3.0, …  
## $ out <fct> 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, …  
## $ datasrc <chr> "Cleveland", "Cleveland", "Cleveland", "Cleveland", "Clevela…

colSums(is.na(final\_data))

## age sex cp trestbps chol fbs restecg thalach   
## 0 0 0 0 0 0 0 0   
## exang oldpeak slope ca thal out datasrc   
## 0 0 0 0 0 0 0

summary(final\_data)

## age sex cp trestbps chol fbs   
## Min. :29.00 0: 96 1: 23 Min. : 94.0 Min. :126.0 0:254   
## 1st Qu.:48.00 1:201 2: 49 1st Qu.:120.0 1st Qu.:211.0 1: 43   
## Median :56.00 3: 83 Median :130.0 Median :243.0   
## Mean :54.54 4:142 Mean :131.7 Mean :247.4   
## 3rd Qu.:61.00 3rd Qu.:140.0 3rd Qu.:276.0   
## Max. :77.00 Max. :200.0 Max. :564.0   
## restecg thalach exang oldpeak slope ca thal   
## 0:147 Min. : 71.0 0:200 Min. :0.000 1:139 0.0:174 3.0:164   
## 1: 4 1st Qu.:133.0 1: 97 1st Qu.:0.000 2:137 1.0: 65 6.0: 18   
## 2:146 Median :153.0 Median :0.800 3: 21 2.0: 38 7.0:115   
## Mean :149.6 Mean :1.056 3.0: 20   
## 3rd Qu.:166.0 3rd Qu.:1.600   
## Max. :202.0 Max. :6.200   
## out datasrc   
## 0:160 Length:297   
## 1:137 Class :character   
## Mode :character   
##   
##   
##

## Questions for future step

What do you not know how to do right now that you need to learn to answer your questions?  
I do not know confidently how to manage datasets having missing values for some of the variables. I am planning on learning that for my next steps. So as of now I am planning on using the Cleveland data set as that is mostly complete and so would help in the model building.  
  
Do you plan on incorporating any machine learning techniques to answer your research questions? Explain.  
I am planning on applying random forest model for this classification problem along with the Generalized Linear Model and see the effectiveness of each models prediction.   
Also before I apply any machine learning techniques I am planning on visualizing the dataset using scatter plots for some of the variables.(e.g, age,sex and some others)   
Also I would like to find the correlation and test between the various predictor variables.