Liver Disease Classification

Nabeel Khan

27-May-2020

Contents

1 Introduction 1

1.1 Background 1

1.2 Aim of Project 1

2 Dataset and Evaluation Metrics 1

2.1 Download Data 2

2.2 Metrics 4

3 Data Exploration 5

4 Data Analysis 6

4.1 Age 6

4.2 Gender 8

4.3 Alkaline Phosphotase 10

4.4 Total Protiens 13

4.5 Albumin 14

1 Introduction

This report is part of the 'HarvardX: PH125.9x Data Science: Capstone' course. In this report, we chose a dataset of our choice and apply various machine learning techniques to perform binary classification to diagnose liver disease.

1.1 Background

The liver plays an important role in keeping us healthy. The main job of liver is to filter the blood coming from the digestive tract, before passing it to the rest of the body. The liver also turns nutrients into chemicals our body needs, turns food into energy, and filters out poisons. So, malfunctioing of liver affects the whole body.

The classification techniques are used in various automatic medical diagnoses tools[1].The problems with liver patients are not easily discovered in an early stage. An early diagnosis of liver problems will help in increasing the survival rate of patiets. We can detect the liver disease by analyzing the levels of enzymes in the blood [2, 3]. A classification algorithm capable of automatically detecting the liver disease can assisst the doctors.

1.2 Aim of Project

The patients with liver disease are on the rise because of excessive consumption of alcohol, inhale of harmful gases, intake of contaminated food, pickles and drugs. The aim of this proect is to develop a bianry classifier, which can use blood enzymes information to diagnose liver disease.

2 Dataset and Evaluation Metrics

We use the Liver Patient Records, which are collected from North East of Andhra Pradesh, India. The data set contains:

1. 416 liver patient records and 167 non-liver patient records.

2.1 Download Data

The dataset is publically available online both at Kaggle and UCI repository. We download data from the website. Then, we split data into a training and validation sets.

- 10% of the data is used for validation, and 90

```
################################
#  Install packages (if not installed)
################################
# Note: this process could take a couple of minutes
repos_path<- "http://cran.us.r-project.org"
if(!require(tidyverse)) install.packages("tidyverse", repos =repos_path)
```

```
## Loading required package: tidyverse

## -- Attaching packages --------------------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts ------------------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
if(!require(caret)) install.packages("caret", repos = repos_path)
```

```
## Loading required package: caret

## Loading required package: lattice

##
## Attaching package: 'caret'

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

```
if(!require(data.table)) install.packages("data.table", repos =repos_path)
```

```
## Loading required package: data.table

##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## The following object is masked from 'package:purrr':
##
##     transpose
```

```
if(!require(lubridate)) install.packages("lubridate", repos = repos_path)
```

```
## Loading required package: lubridate

##
## Attaching package: 'lubridate'
## The following objects are masked from 'package:data.table':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday,
##     week, yday, year
```

```
## The following object is masked from 'package:base':
##
##     date
```

```r
if(!require(dplyr)) install.packages("dplyr", repos = repos_path)
if(!require(sjmisc)) install.packages("dplyr", repos = repos_path)
```

```
## Loading required package: sjmisc
```

```
##
## Attaching package: 'sjmisc'
```

```
## The following object is masked from 'package:purrr':
##
##     is_empty
```

```
## The following object is masked from 'package:tidyr':
##
##     replace_na
```

```
## The following object is masked from 'package:tibble':
##
##     add_case
```

```r
if(!require(sjmisc)) install.packages("scales", repos = repos_path)

#################################
# Load libraries
#################################
library(lubridate)
library(tidyverse)
library(dplyr)
library(lubridate)
library(sjmisc)
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

```r
#################################
# Downloading data
#################################
# Indian Live Patient Records :
 # https://www.kaggle.com/uciml/indian-liver-patient-records/
 # https://archive.ics.uci.edu/ml/machine-learning-databases/00225/Indian Liver Patient Dataset (ILPD).
```

```r
url <- "https://archive.ics.uci.edu/ml/machine-learning-databases/00225/Indian Liver Patient Dataset (IL
# Download csv
liverData <- read.csv(url)

# Rename columns of csv
colnames(liverData)<- c("Age","Gender","Total_Bilirubin","Direct_Bilirubin", "Alkaline_Phosphotase","Ala

################################
# Creating training and validation sets
################################

# Validation set will be 10% of whole data
set.seed(1, sample.kind = "Rounding")
```

```
## Warning in set.seed(1, sample.kind = "Rounding"): non-uniform 'Rounding'
## sampler used
```

```r
test_index <- createDataPartition(y = liverData$Dataset, times = 1, p = 0.1, list = FALSE)

training <- liverData[-test_index,]
validation <- liverData[test_index,]

 # Removing the objects from environment as no longer required
rm(liverData)
```

2.2 Metrics

To evaluate the performance of classifiers, we will use following metrics:

1. **Accuracy** It is the ratio of number of correct predictions to the total number of input samples.

$$Accuracy = \frac{Truepositives + Truenegatives}{TotalPredictions} \tag{1}$$

2. **Sensitivity** It is also referred as true positive rate or recall. It is the proportion of true positives that are correctly identified.

$$Sensitivity = \frac{Numberoftruepositives}{Numberoftruepositives + Numberoffalsenegatives} \tag{2}$$

3. **Precision** It is defined as the proportion of the true positives against all the positive results.

$$Precision = \frac{Numberoftruepositives}{Numberoftruepositives + Numberoffalsepositives} \tag{3}$$

4. **Specificity** It is the True negative rate. It is the proportion of true negatives that are correctly identified.

$$Specificity = \frac{Numberoftruenegatives}{Numberoftruenegatives + Numberoffalsepositives} \tag{4}$$

5. **F1 Score** One metric that is preferred over overall accuracy is the average of specificity and sensitivity, referred to as balanced accuracy. Because specificity and sensitivity are rates, it is more appropriate to compute the harmonic average. In fact, the F1-score is widely used to compute harmonic average of precision and recall.

$$F1Score = 2 * \frac{Precision - Recall}{Precision + Recall} \tag{5}$$

3 Data Exploration

The dataset contains 11 variables namely, ‘Age’,‘Gender’,‘Total_Bilirubin’, or “Alkaline_Phosphotase”. The ‘Dataset’ variable indicates if the liver has a disease or not. For instance, a value of 1 indicates a disease and 2 indicates no disease.

All other variables except Age",Gender“, and “Dataset” represent the amount of enzymes in the blood. These variables will be used to train our machine learning models for making predictions or diagnosis.

```
head(training)
```

Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Aminotransferase	Aspartate_Aminotran
62	Male	10.9	5.5	699	64	
62	Male	7.3	4.1	490	60	
58	Male	1.0	0.4	182	14	
72	Male	3.9	2.0	195	27	
46	Male	1.8	0.7	208	19	
26	Female	0.9	0.2	154	16	

The training dataset has 523 records and there are no null values (confirmed using summary).

```
sprintf("Rows of training dataset = %d", nrow(training))
```

```
## [1] "Rows of training dataset = 523"
```

```
print("========================")
```

```
## [1] "========================"
```

```
summary(training)
```

```
##       Age           Gender    Total_Bilirubin Direct_Bilirubin
##  Min.   : 4.00   Female:125   Min.   : 0.40   Min.    : 0.100
##  1st Qu.:33.00   Male   :398   1st Qu.: 0.80   1st Qu.: 0.200
##  Median :45.00                Median : 1.00   Median : 0.300
##  Mean   :45.33                Mean   : 3.22   Mean    : 1.446
##  3rd Qu.:58.00                3rd Qu.: 2.60   3rd Qu.: 1.300
##  Max.   :90.00                Max.   :75.00   Max.    :19.700
##
##  Alkaline_Phosphotase Alamine_Aminotransferase Aspartate_Aminotransferase
##  Min.   :  63.0       Min.   :  10.00          Min.   :  10.0
##  1st Qu.: 176.0       1st Qu.:  24.00          1st Qu.:  25.0
##  Median : 208.0       Median :  35.00          Median :  41.0
##  Mean   : 289.9       Mean   :  76.34          Mean   : 105.0
##  3rd Qu.: 298.0       3rd Qu.:  60.00          3rd Qu.:  86.5
##  Max.   :1896.0       Max.   :1680.00          Max.   :4929.0
##
##  Total_Protiens    Albumin      Albumin_and_Globulin_Ratio    Dataset
##  Min.   :2.70   Min.   :0.900   Min.   :0.3000             Min.   :1.000
##  1st Qu.:5.80   1st Qu.:2.600   1st Qu.:0.7000             1st Qu.:1.000
##  Median :6.60   Median :3.100   Median :0.9300             Median :1.000
##  Mean   :6.49   Mean   :3.147   Mean   :0.9458             Mean   :1.281
##  3rd Qu.:7.20   3rd Qu.:3.800   3rd Qu.:1.1000             3rd Qu.:2.000
##  Max.   :9.50   Max.   :5.500   Max.   :2.8000             Max.   :2.000
##                                 NA's   :4
```

3.0.1 Data Wrangling

The variable “Dataset" is our prediction and we will use it to analyse the performance of machine lerning models. To improve readability, we create a new column namely “Liver Disease”, which will have two values:

1. Malignant (M) indicating that the patienit has a liver disease.
2. Benign (B) indicating that the patient has no no liver disease.

After creating a new column, wr delete the ‘Dataset’ variable. We apply these operations to both training and validation datasets.

```
# Adding a new column, which will contain the disease information
training <- transform(training, Disease= ifelse(Dataset==1, "M","B"))
validation <- transform(validation, Disease= ifelse(Dataset==1, "M","B"))

# Deleting the column 'Dataset' as no longer required
training<-within(training, rm(Dataset))
validation<-within(validation, rm(Dataset))

# Displaying the first siz rows
head(training)
```

Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Aminotransferase	Aspartate_Aminotran
62	Male	10.9	5.5	699	64	
62	Male	7.3	4.1	490	60	
58	Male	1.0	0.4	182	14	
72	Male	3.9	2.0	195	27	
46	Male	1.8	0.7	208	19	
26	Female	0.9	0.2	154	16	

4 Data Analysis

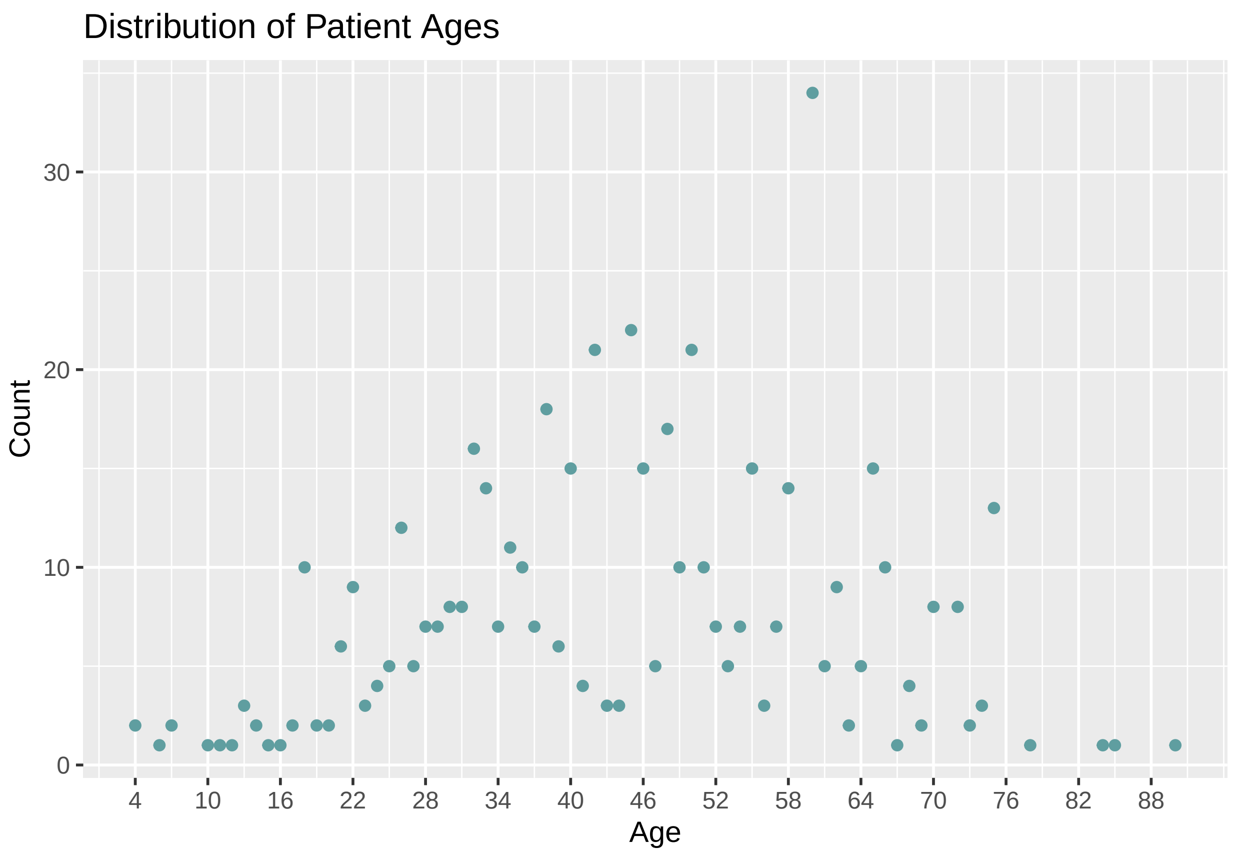
In this section, we extract insights from all variables to get in depth understanding.

4.1 Age

The dataset consists of patients with varying ages, which makes this dataset robust and un-biased towards a specific age group.

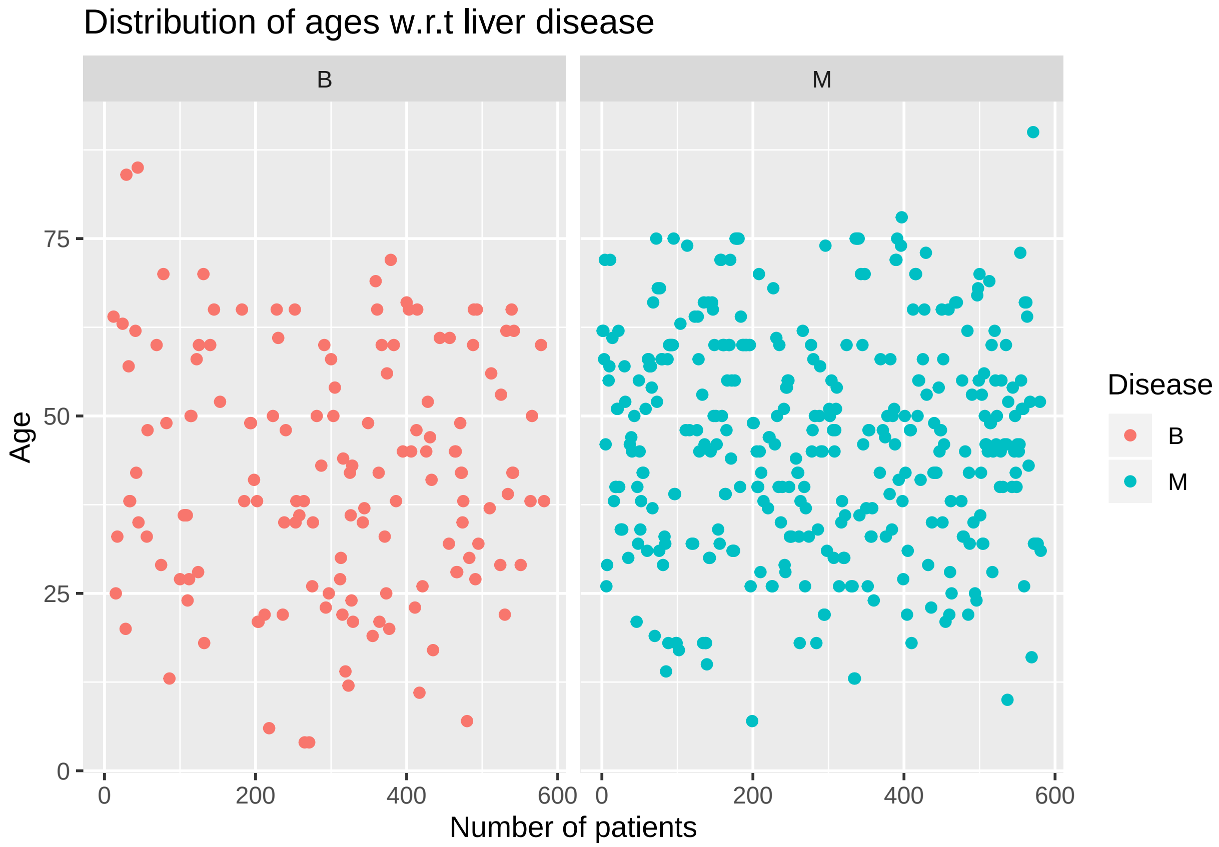
```
# Extracting frequency of patient ages
age_stats <-as.data.frame(table(training$Age))
names(age_stats)<- c("Age","Count")
# Remvoing the factor
age_stats$Age<-as.numeric(levels(age_stats$Age))

# Plotting distribution of ages
age_stats %>% ggplot(aes(Age, Count)) +
  geom_point(color="cadetblue") +
  scale_x_continuous(breaks = round(seq(min(age_stats$Age), max(age_stats$Age), by = 6),1)) +
  ggtitle("Distribution of Patient Ages")
```



We analyse the age distributions with respect to the liver disease. The distributions again suggest a good spread of age group.

```
# Plotting distributions of ages versus liver diseases
training %>%
  ggplot(aes(as.numeric(row.names(training)),Age, color=Disease)) +
  geom_point() +
  labs(y="Age", x = "Number of patients")+
  facet_wrap( ~ Disease) +
  ggtitle("Distribution of ages w.r.t liver disease")
```



4.2 Gender

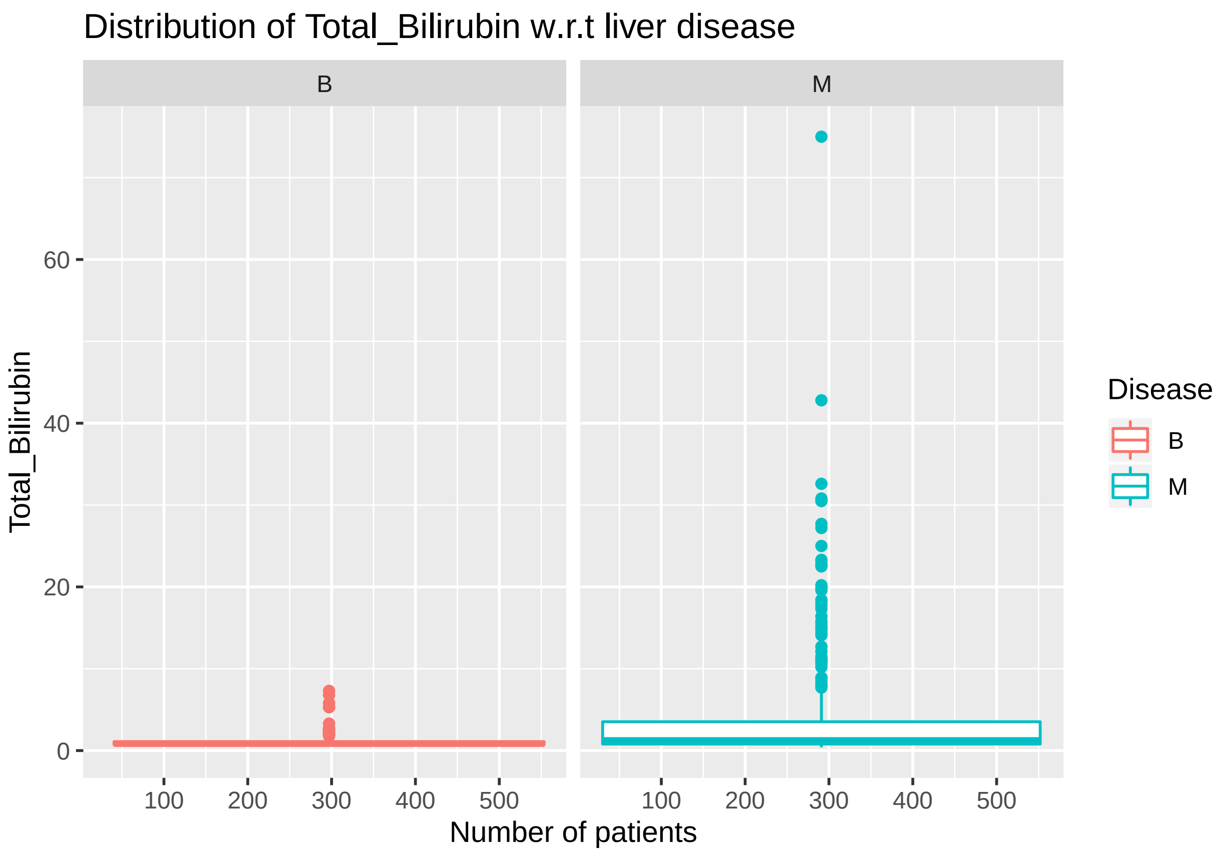
The analysis indicates that 76% of the patient records belong to males. It would have been good to have more and less equal distribution of genders. Although, it may not make difference in the performance of models.

```
# Getting summary of genders
summary(training$Gender)
```

```
## Female   Male
##    125    398
```

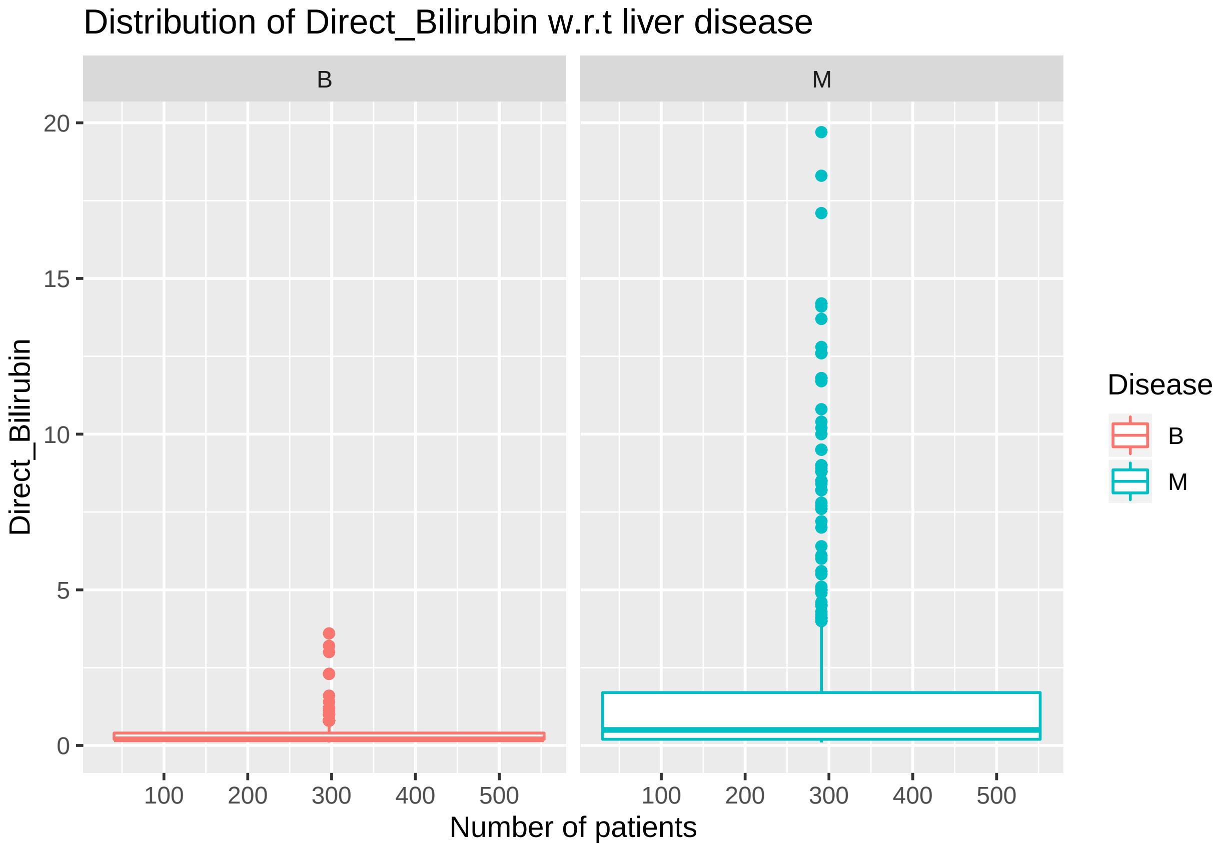
\subsection{Total_Bilirubin} Bilirubin refears to any form of a yellowish pigment made in the liver when red blood cells are broken down. We can see pattern that levels of Total_Bilirubin are high for patients with liver diseases.

```
# Plotting distributions of Total_Bilirubin versus liver diseases
training %>%
  ggplot(aes(as.numeric(row.names(training)),Total_Bilirubin, color=Disease)) +
  geom_boxplot() +
  labs(y="Total_Bilirubin", x = "Number of patients")+
  facet_wrap( ~ Disease) +
  ggtitle("Distribution of Total_Bilirubin w.r.t liver disease")
```



\subsection{Direct__Bilirubin} Again, we can see that levels of Direct_Bilirubin are also high for patients with liver diseases.

```
# Plotting distributions of Total_Bilirubin versus liver diseases
training %>%
  ggplot(aes(as.numeric(row.names(training)),Direct_Bilirubin, color=Disease)) +
  geom_boxplot() +
  labs(y="Direct_Bilirubin", x = "Number of patients")+
  facet_wrap( ~ Disease) +
  ggtitle("Distribution of Direct_Bilirubin w.r.t liver disease")
```



The correlations indicate that there is a weak correlation between liver disease and bilirubin. Though, the values are comparatively higher with liver diseases.

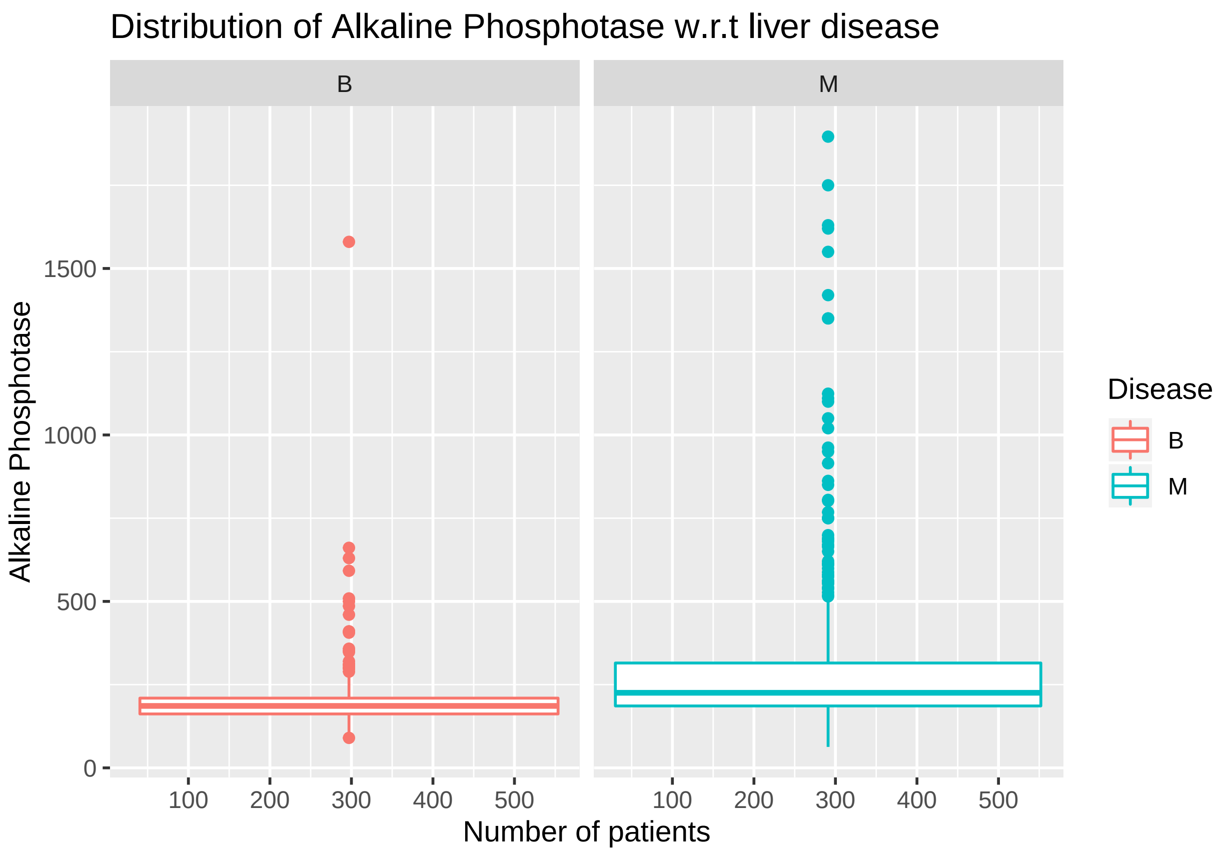
```
subset_train <- training[c("Total_Bilirubin","Direct_Bilirubin")]
subset_train <- transform(subset_train, Disease= ifelse(training$Disease=="M", 1,0))
cor(subset_train)
```

```
##                  Total_Bilirubin Direct_Bilirubin   Disease
## Total_Bilirubin        1.0000000        0.8584292 0.2065553
## Direct_Bilirubin       0.8584292        1.0000000 0.2347388
## Disease                0.2065553        0.2347388 1.0000000
```

4.3 Alkaline Phosphotase

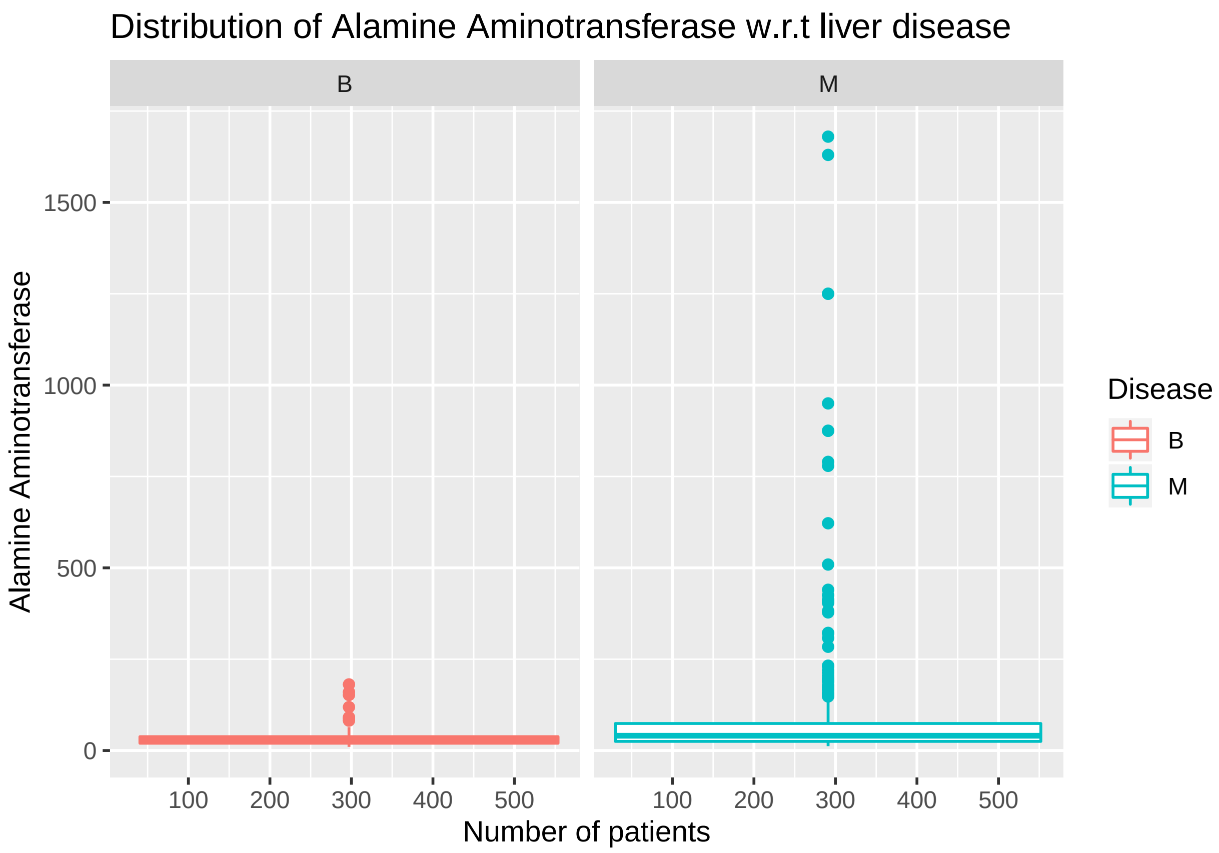
Alkaline phosphatase (ALP) is an enzyme in a person's blood that helps break down proteins. We observe that levels of Alkaline Phosphotase are comparatively high for patients with liver diseases.

```
# Plotting distributions of Alkaline Phosphotase versus liver diseases
training %>%
  ggplot(aes(as.numeric(row.names(training)),Alkaline_Phosphotase, color=Disease)) +
  geom_boxplot() +
  labs(y="Alkaline Phosphotase", x = "Number of patients")+
  facet_wrap( ~ Disease) +
  ggtitle("Distribution of Alkaline Phosphotase w.r.t liver disease")
```



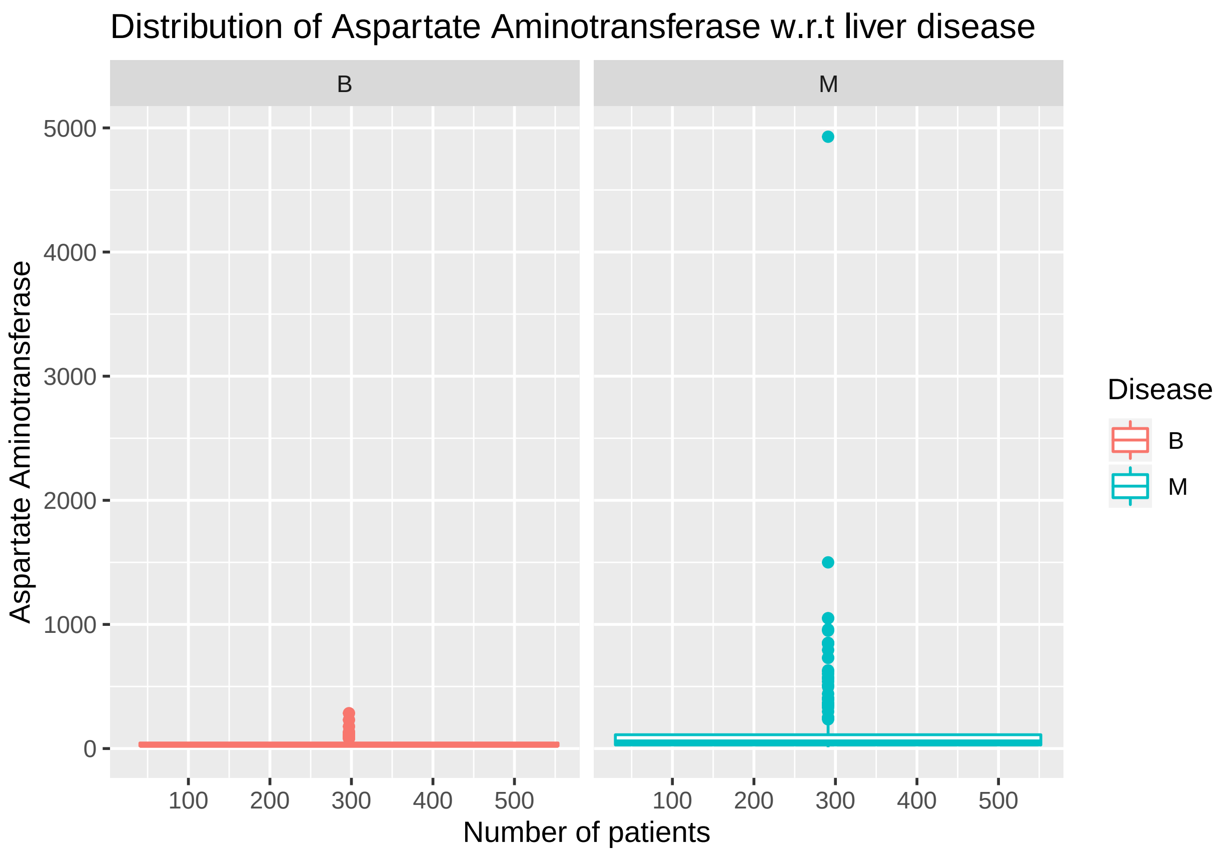
\subsection{Alamine_Aminotransferase} Alanine aminotransferase (ALT) is an enzyme found primarily in the liver and kidney. ALT is increased with liver damage and is used to screen liver disease. We can see that levels of Alamine_Aminotransferase are high for patients with liver diseases.

```
# Plotting distributions of Alamine Aminotransferase versus liver diseases
training %>%
  ggplot(aes(as.numeric(row.names(training)),Alamine_Aminotransferase, color=Disease)) +
  geom_boxplot() +
  labs(y="Alamine Aminotransferase", x = "Number of patients")+
  facet_wrap( ~ Disease) +
  ggtitle("Distribution of Alamine Aminotransferase w.r.t liver disease")
```



\subsection{Aspartate_Aminotransferase} Aspartate aminotransferase (AST) is an enzyme found in cells throughout the body but mostly in the heart and liver. In healthy individuals, levels of AST in the blood are low. When liver is damaged, they release AST into the blood thus raising the levels.We can see that levels of AST are comparatively high for very few patients with liver diseases.

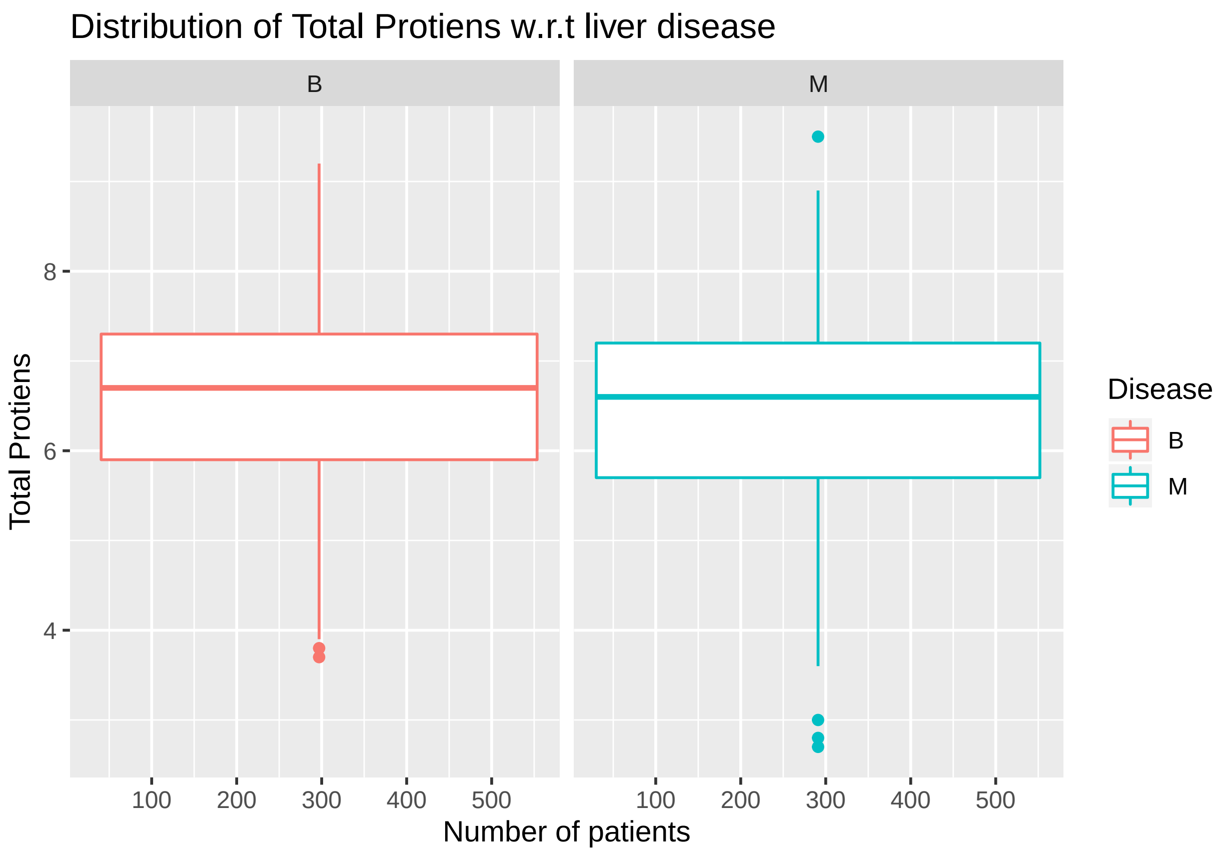
```
# Plotting distributions of Aspartate_Aminotransferase versus liver diseases
training %>%
  ggplot(aes(as.numeric(row.names(training)),Aspartate_Aminotransferase, color=Disease)) +
  geom_boxplot() +
  labs(y="Aspartate Aminotransferase", x = "Number of patients")+
  facet_wrap( ~ Disease) +
  ggtitle("Distribution of Aspartate Aminotransferase w.r.t liver disease")
```



4.4 Total Protiens

The total protein test measures the total amount albumin and globulin in your body. It is used as part of your routine health checkup. The anaysis suggest that there is no correlation between liver disease and total protiens in our dataset. So, we will not use it for model training.

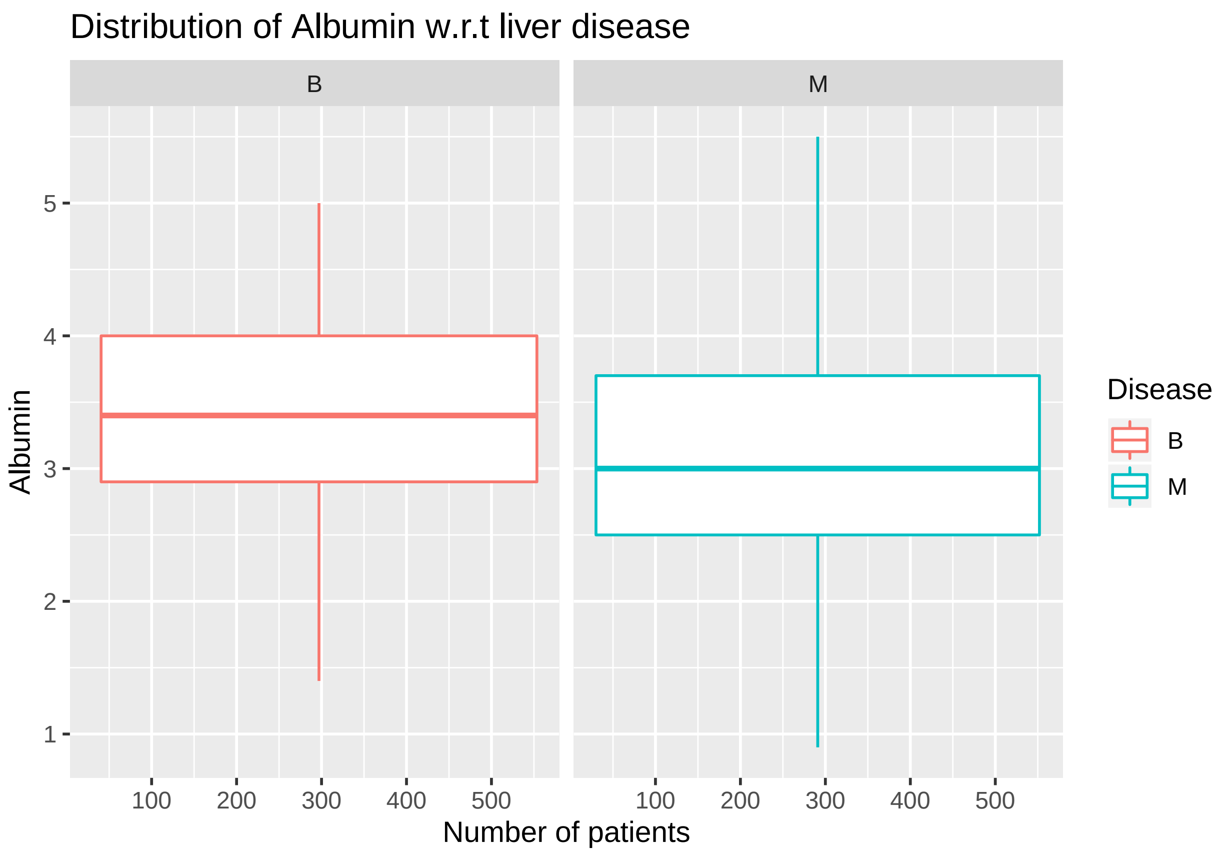
```
# Plotting distributions of Total Protiens versus liver diseases
training %>%
  ggplot(aes(as.numeric(row.names(training)),Total_Protiens, color=Disease)) +
  geom_boxplot() +
  labs(y="Total Protiens", x = "Number of patients")+
  facet_wrap( ~ Disease) +
  ggtitle("Distribution of Total Protiens w.r.t liver disease")
```



4.5 Albumin

Albumin is a protein made by your liver. Albumin helps keep fluid in your bloodstream so it doesn't leak into other tissues. Low albumin levels can indicate a problem with your liver or kidneys. In our dataset, this variable has no strong correlation with the liver disease.

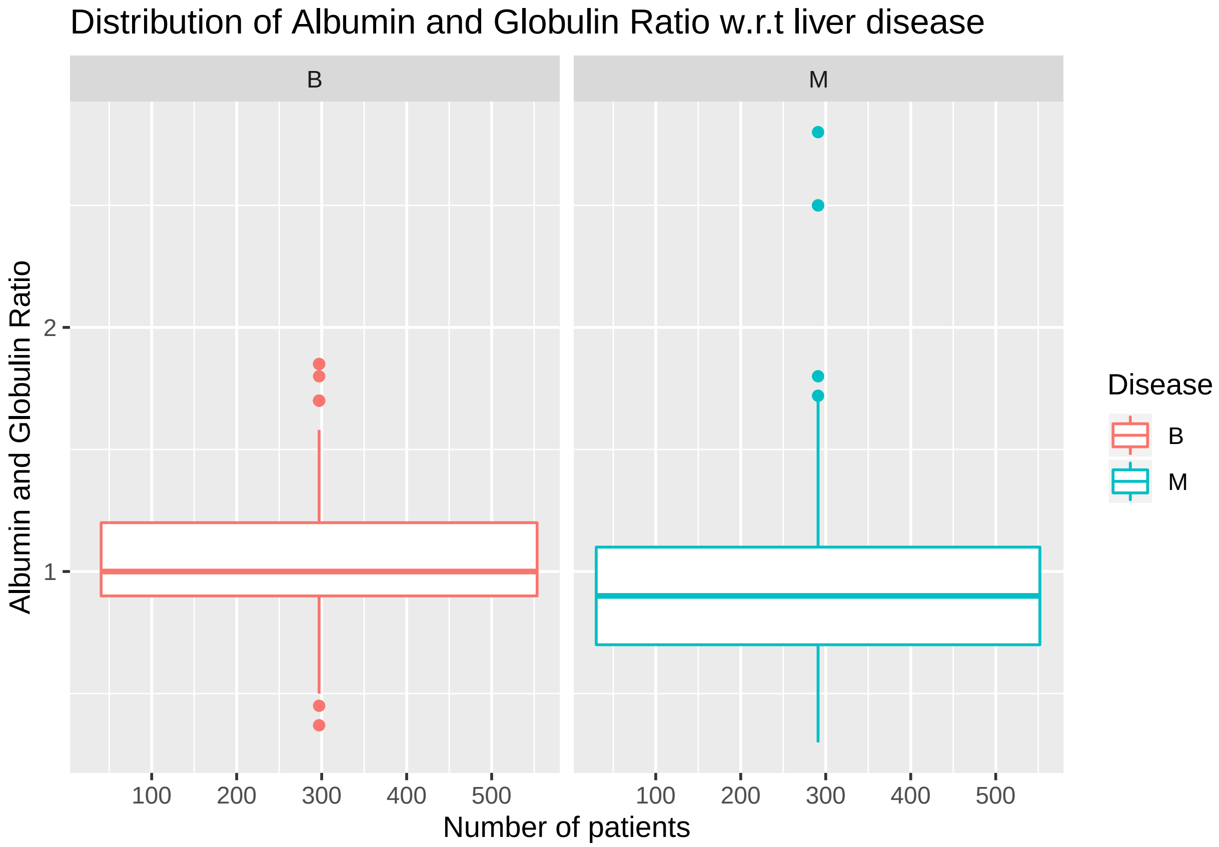
```
# Plotting distributions of Albumin versus liver diseases
training %>%
  ggplot(aes(as.numeric(row.names(training)),Albumin, color=Disease)) +
  geom_boxplot() +
  labs(y="Albumin", x = "Number of patients")+
  facet_wrap( ~ Disease) +
  ggtitle("Distribution of Albumin w.r.t liver disease")
```



\subsection{Albumin_and_Globulin_Ratio} There are two classes of proteins are found in the blood. They are important for body growth, development, and health. They form the structural part of most organs and make up enzymes and hormones that regulate body functions. We can see that these protiens have no correlation with the liver disease.

```
# Plotting distributions of Albumin versus liver diseases
training %>%
  ggplot(aes(as.numeric(row.names(training)),Albumin_and_Globulin_Ratio, color=Disease)) +
  geom_boxplot() +
  labs(y="Albumin and Globulin Ratio", x = "Number of patients")+
  facet_wrap( ~ Disease) +
  ggtitle("Distribution of Albumin and Globulin Ratio w.r.t liver disease")
```

```
## Warning: Removed 4 rows containing non-finite values (stat_boxplot).
```



We can that Bilirubin enzymes are highly correlated with the liver disease. We are going to use these 5 variables to predict the liver disease.

```
samples <- training
samples <- transform(samples, Disease= ifelse(training$Disease=="M", 1,0))
samples<-within(samples, rm(Age,Gender,Total_Protiens,Albumin,Albumin_and_Globulin_Ratio))
colnames(samples)<-c ("T_Bil","T_Bil","A_Phos","Al_Amin","Asp_Amino","Disease")
cor(samples)
```

```
##                T_Bil     T_Bil    A_Phos   Al_Amin Asp_Amino   Disease
## T_Bil     1.0000000 0.8584292 0.2356568 0.1686019 0.2003604 0.2065553
## T_Bil     0.8584292 1.0000000 0.2681528 0.1897871 0.2219221 0.2347388
## A_Phos    0.2356568 0.2681528 1.0000000 0.1692258 0.2156400 0.1782120
## Al_Amin   0.1686019 0.1897871 0.1692258 1.0000000 0.7317857 0.1616705
## Asp_Amino 0.2003604 0.2219221 0.2156400 0.7317857 1.0000000 0.1488358
## Disease   0.2065553 0.2347388 0.1782120 0.1616705 0.1488358 1.0000000
```

References

[1] Ethan Du-Crowa, Lucy Warrenb, Susan M Astleya and Johan Hullemanc,"Is there a safety-net effect with Computer-Aided Detection (CAD)?", Medical Imaging 2019.

[2] Eugene, R., Sorrell, Michael F.; Maddrey, Willis C., "Schiff's Diseases of the Liver", 10th Edition, Lippincott Williams & Wilkins by Schiff.

[3] Bendi, Venkata . R, M. S. Prasad Babu, and N. B. Venkateswarlu, "Critical Comparative Study of Liver Patients from USA and INDIA: An Exploratory Analysis", International Journal of Computer Science Issues, May 2012.