PROBLEM STATEMENT:

Developing a Multi-nominal Logistic Regression Model in R for Predicting Chronic HepatitisC Infection from the Absence of Fibrosis to End-Stage Liver Cirrhosis.

DATASET DETAILS:

University of California, Irvine (UCI) Machine Learning Repository: HCV dataset. The data set contains laboratory values of blood donors and Hepatitis C patients and demographic values like age.

Data Set Characteristics:	Multivariate	**Number of Instances:**	615
Attribute Characteristics:	Integer, Real	**Number of Attributes:**	14
Associated Tasks:	Classification, Clustering	**Missing Values?**	Yes

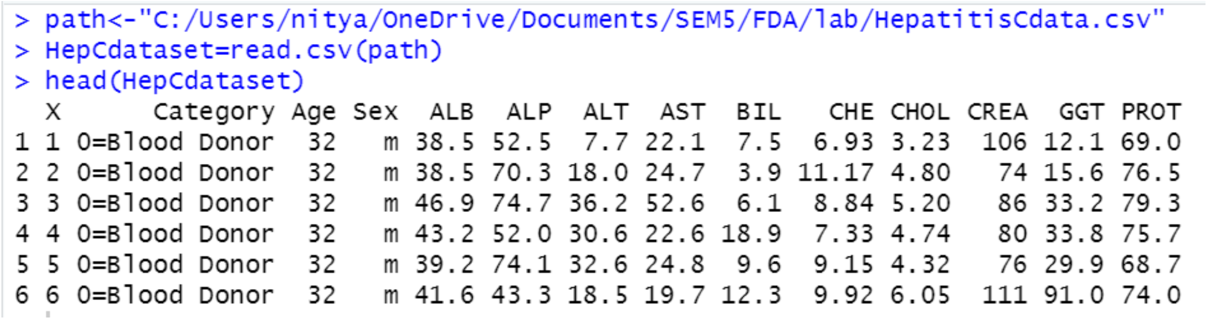
For classification model, the target attribute for classification is Category (blood donors vs. Hepatitis C (including its progress ('just' Hepatitis C, Fibrosis, Cirrhosis).

Attribute Information: All attributes except Category and Sex are numerical. The laboratory data are the attributes 5-14.
1) X (Patient ID/No.)
2) Category (diagnosis) (values: '0=Blood Donor', '0s=suspect Blood Donor', '1=Hepatitis', '2=Fibrosis', '3=Cirrhosis')
3) Age (in years)
4) Sex (f,m)
5) ALB
6) ALP
7) ALT
8) AST
9) BIL
10) CHE
11) CHOL
12) CREA
13) GGT
14) PROT

Code	Hepatitis C category	Frequencies
0	Blood donor	533
0s	Suspect blood donor	7
1	Hepatitis	24
2	Fibrosis	21
3	Cirrhosis	30

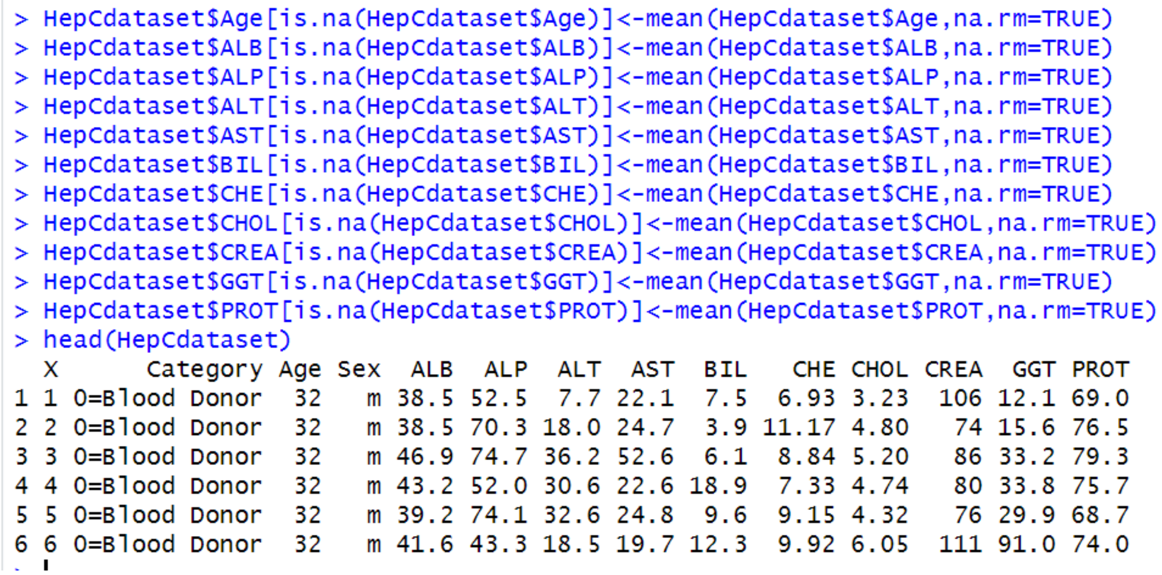
Data Visualization:

>>Dataset



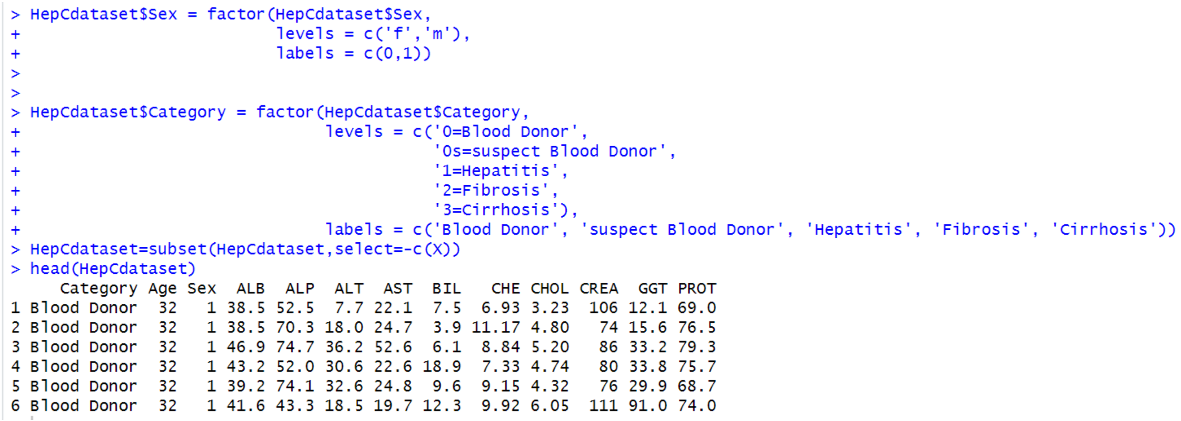
```
> path<-"C:/Users/nitya/OneDrive/Documents/SEM5/FDA/lab/HepatitisCdata.csv"
> HepCdataset=read.csv(path)
> head(HepCdataset)
  X      Category Age Sex  ALB  ALP  ALT  AST  BIL   CHE CHOL CREA  GGT PROT
1 1 0=Blood Donor  32   m 38.5 52.5  7.7 22.1  7.5  6.93 3.23  106 12.1 69.0
2 2 0=Blood Donor  32   m 38.5 70.3 18.0 24.7  3.9 11.17 4.80   74 15.6 76.5
3 3 0=Blood Donor  32   m 46.9 74.7 36.2 52.6  6.1  8.84 5.20   86 33.2 79.3
4 4 0=Blood Donor  32   m 43.2 52.0 30.6 22.6 18.9  7.33 4.74   80 33.8 75.7
5 5 0=Blood Donor  32   m 39.2 74.1 32.6 24.8  9.6  9.15 4.32   76 29.9 68.7
6 6 0=Blood Donor  32   m 41.6 43.3 18.5 19.7 12.3  9.92 6.05  111 91.0 74.0
```

>>Handling null values



```
> HepCdataset$Age[is.na(HepCdataset$Age)]<-mean(HepCdataset$Age,na.rm=TRUE)
> HepCdataset$ALB[is.na(HepCdataset$ALB)]<-mean(HepCdataset$ALB,na.rm=TRUE)
> HepCdataset$ALP[is.na(HepCdataset$ALP)]<-mean(HepCdataset$ALP,na.rm=TRUE)
> HepCdataset$ALT[is.na(HepCdataset$ALT)]<-mean(HepCdataset$ALT,na.rm=TRUE)
> HepCdataset$AST[is.na(HepCdataset$AST)]<-mean(HepCdataset$AST,na.rm=TRUE)
> HepCdataset$BIL[is.na(HepCdataset$BIL)]<-mean(HepCdataset$BIL,na.rm=TRUE)
> HepCdataset$CHE[is.na(HepCdataset$CHE)]<-mean(HepCdataset$CHE,na.rm=TRUE)
> HepCdataset$CHOL[is.na(HepCdataset$CHOL)]<-mean(HepCdataset$CHOL,na.rm=TRUE)
> HepCdataset$CREA[is.na(HepCdataset$CREA)]<-mean(HepCdataset$CREA,na.rm=TRUE)
> HepCdataset$GGT[is.na(HepCdataset$GGT)]<-mean(HepCdataset$GGT,na.rm=TRUE)
> HepCdataset$PROT[is.na(HepCdataset$PROT)]<-mean(HepCdataset$PROT,na.rm=TRUE)
> head(HepCdataset)
  X      Category Age Sex  ALB  ALP  ALT  AST  BIL   CHE CHOL CREA  GGT PROT
1 1 0=Blood Donor  32   m 38.5 52.5  7.7 22.1  7.5  6.93 3.23  106 12.1 69.0
2 2 0=Blood Donor  32   m 38.5 70.3 18.0 24.7  3.9 11.17 4.80   74 15.6 76.5
3 3 0=Blood Donor  32   m 46.9 74.7 36.2 52.6  6.1  8.84 5.20   86 33.2 79.3
4 4 0=Blood Donor  32   m 43.2 52.0 30.6 22.6 18.9  7.33 4.74   80 33.8 75.7
5 5 0=Blood Donor  32   m 39.2 74.1 32.6 24.8  9.6  9.15 4.32   76 29.9 68.7
6 6 0=Blood Donor  32   m 41.6 43.3 18.5 19.7 12.3  9.92 6.05  111 91.0 74.0
```

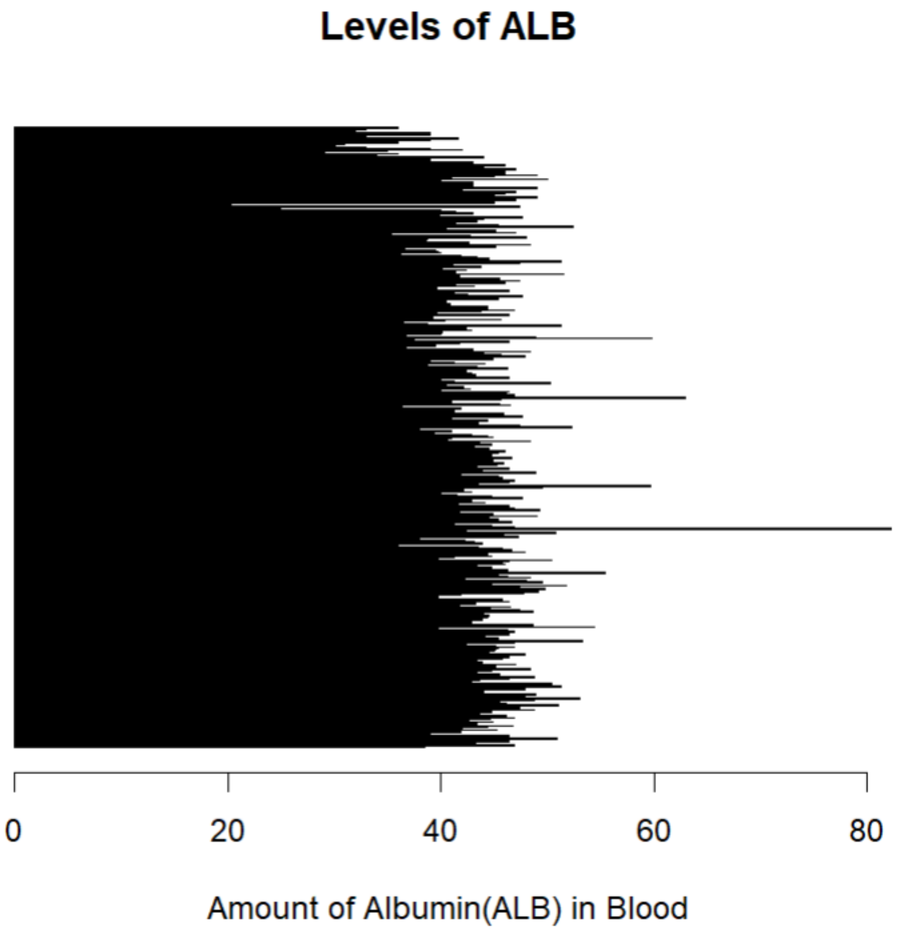
>>Handling Categorical Variables and removing unwanted features



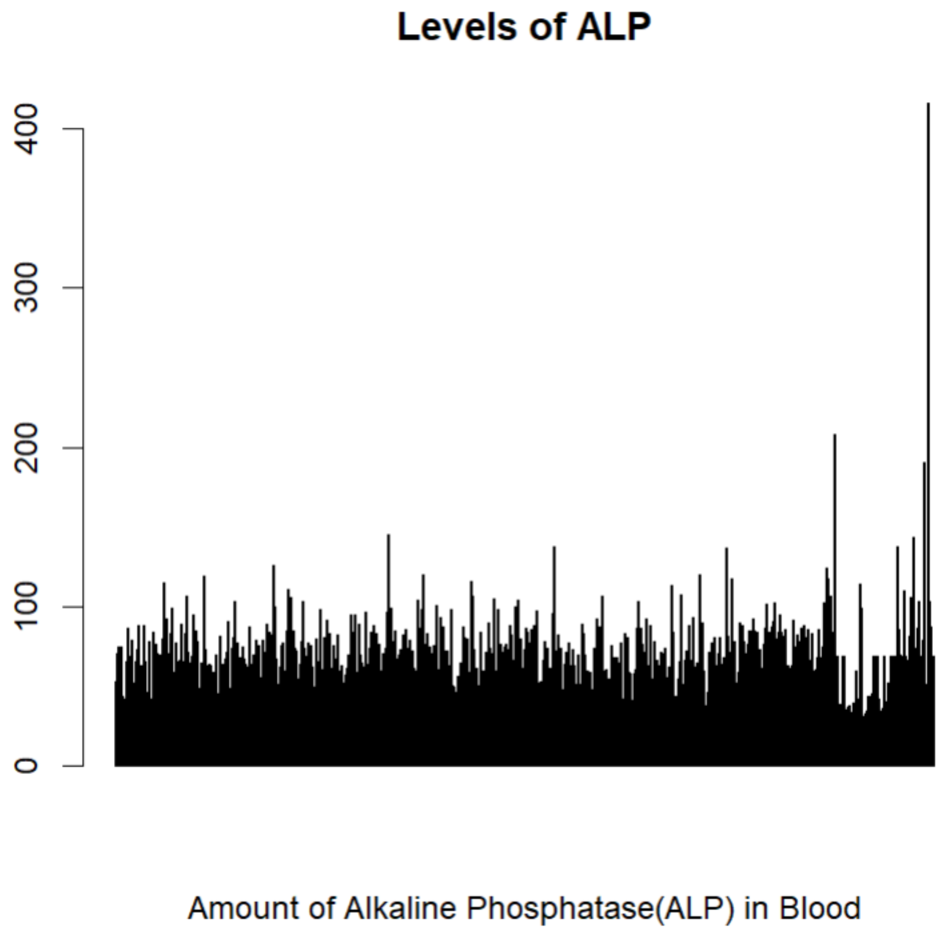
```
> HepCdataset$Sex = factor(HepCdataset$Sex,
+                          levels = c('f','m'),
+                          labels = c(0,1))
>
>
> HepCdataset$Category = factor(HepCdataset$Category,
+                               levels = c('0=Blood Donor',
+                                          '0s=suspect Blood Donor',
+                                          '1=Hepatitis',
+                                          '2=Fibrosis',
+                                          '3=Cirrhosis'),
+                               labels = c('Blood Donor', 'suspect Blood Donor', 'Hepatitis', 'Fibrosis', 'Cirrhosis'))
> HepCdataset=subset(HepCdataset,select=-c(X))
> head(HepCdataset)
     Category Age Sex  ALB  ALP  ALT  AST  BIL   CHE CHOL CREA  GGT PROT
1 Blood Donor  32   1 38.5 52.5  7.7 22.1  7.5  6.93 3.23  106 12.1 69.0
2 Blood Donor  32   1 38.5 70.3 18.0 24.7  3.9 11.17 4.80   74 15.6 76.5
3 Blood Donor  32   1 46.9 74.7 36.2 52.6  6.1  8.84 5.20   86 33.2 79.3
4 Blood Donor  32   1 43.2 52.0 30.6 22.6 18.9  7.33 4.74   80 33.8 75.7
5 Blood Donor  32   1 39.2 74.1 32.6 24.8  9.6  9.15 4.32   76 29.9 68.7
6 Blood Donor  32   1 41.6 43.3 18.5 19.7 12.3  9.92 6.05  111 91.0 74.0
```

Visualization of data:

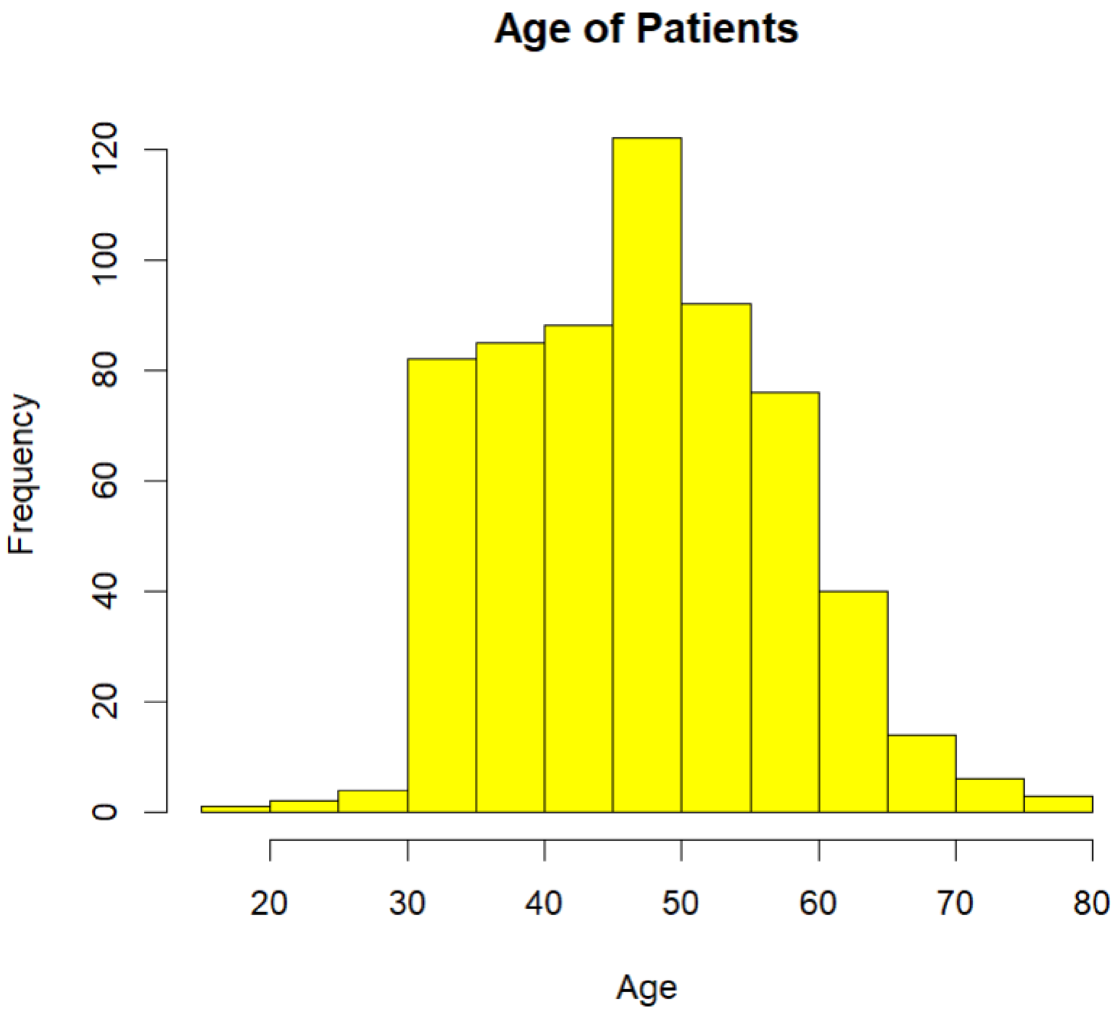
```
# Horizontal Bar Plot for
# Amount of Albumin(ALB) in Blood
barplot(HepCdataset$ALB,
        main = 'Levels of ALB',
        xlab = 'Amount of Albumin(ALB) in Blood', horiz = TRUE)
```



```
# Vertical Bar Plot for
# Amount of Alkaline Phosphatase(ALP) in Blood
barplot(HepCdataset$ALP, main = 'Levels of ALP',
        xlab = 'Amount of Alkaline Phosphatase(ALP) in Blood', horiz = FALSE)
```

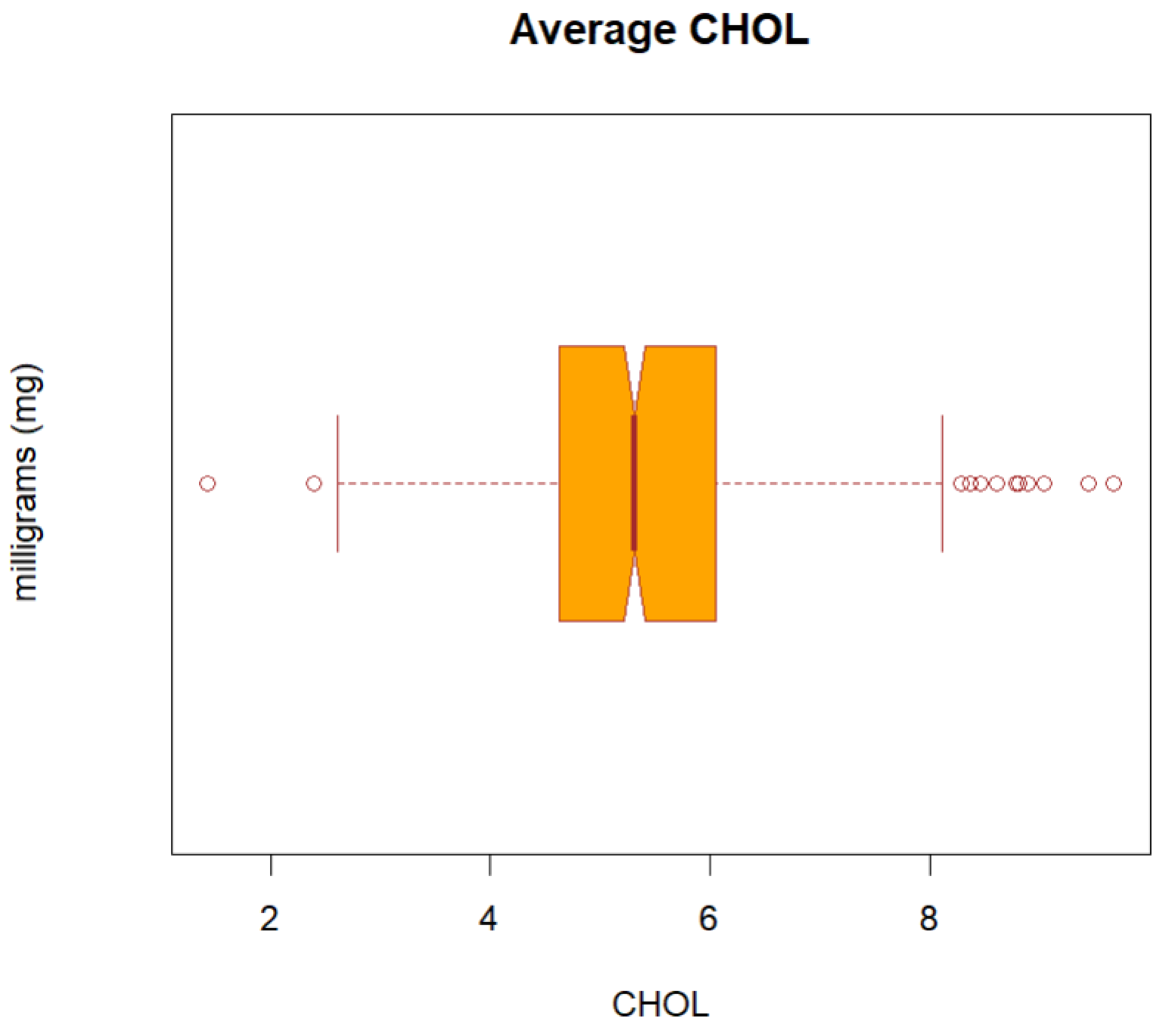


```
hist(HepCdataset$Age, main ="Age of Patients",
    xlab ="Age",
    xlim = c(15,80), col ="yellow",
    freq = TRUE)
```



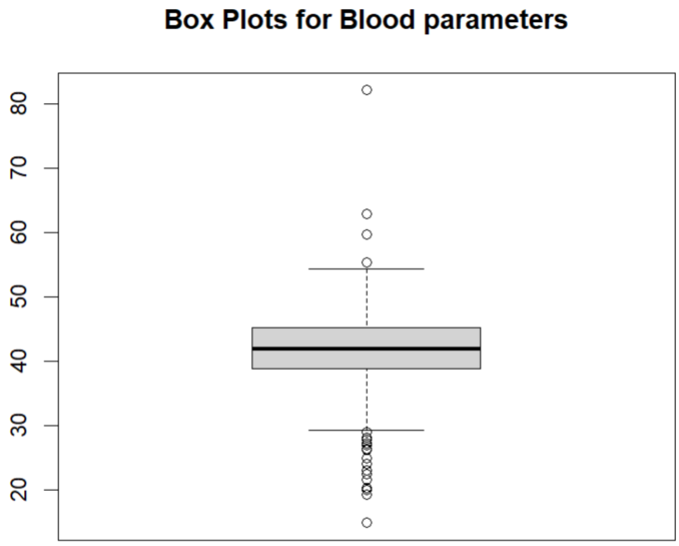
From this bar graph we can see that most of the patients are between age 45-50

```
boxplot(HepCdataset$CHOL, main = "Average CHOL",
        xlab = "CHOL", ylab = "milligrams (mg)",
        col = "orange", border = "brown",
        horizontal = TRUE, notch = TRUE)
```

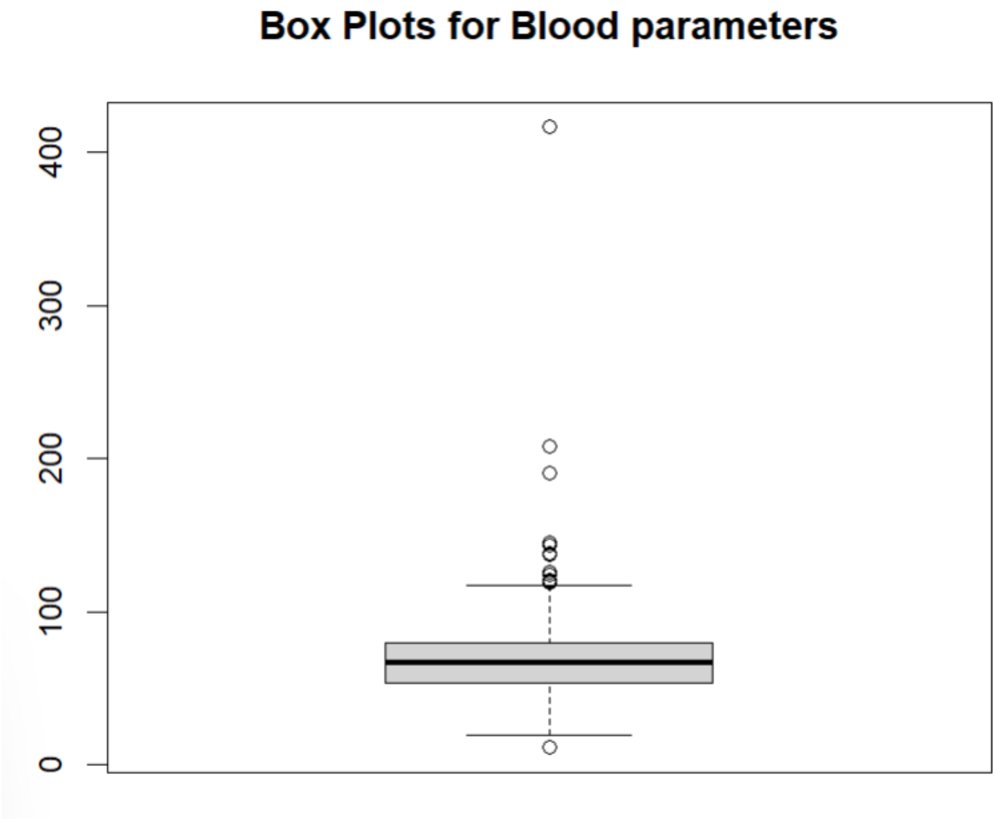


We can observe that there are countable amount of outliers

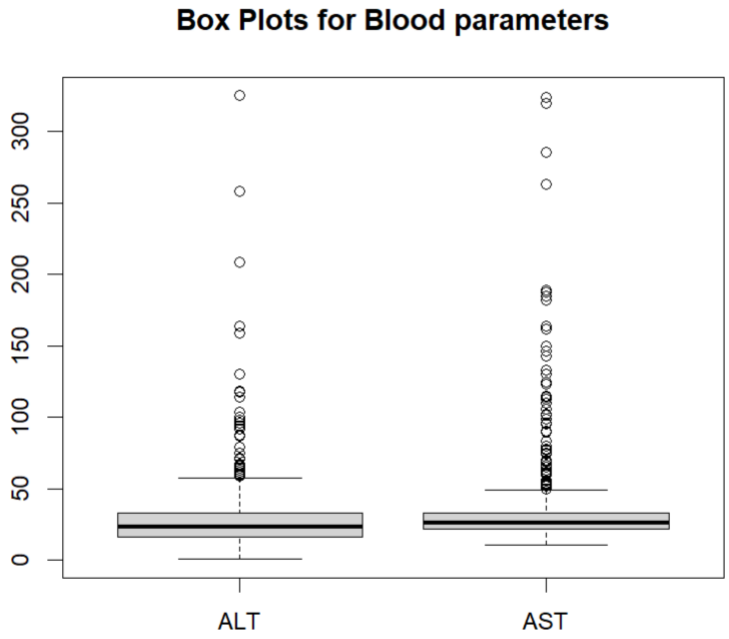
```
boxplot(boxplot(HepCdataset[, 4],
        main ='Box Plots for Blood parameters')[, 4],
        main ='Box Plots for Blood parameters')
```



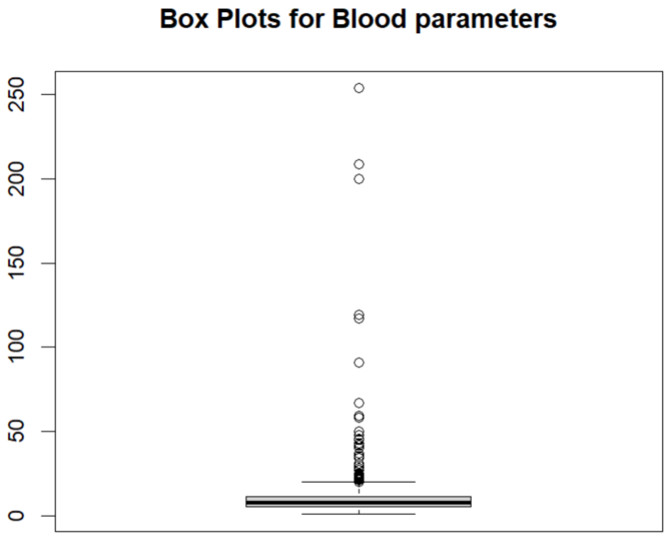
```
boxplot(boxplot(HepCdataset[, 5],
        main ='Box Plots for Blood parameters')[, 5],
        main ='Box Plots for Blood parameters')
```



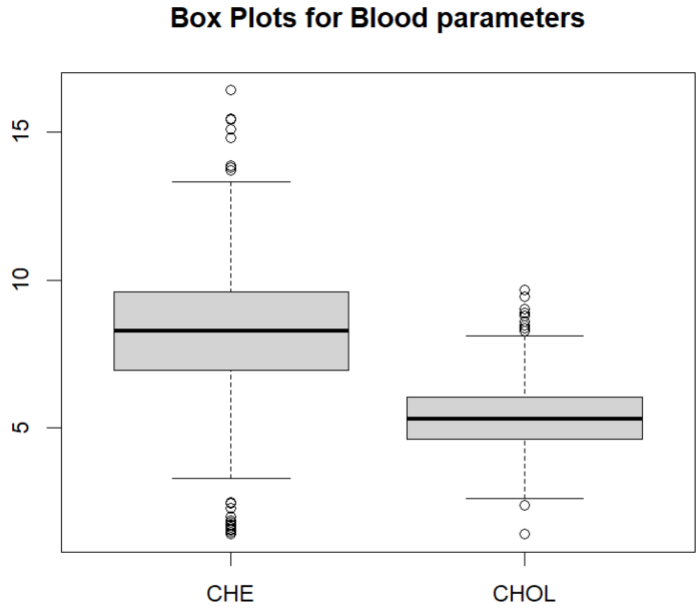
```
boxplot(boxplot(HepCdataset[, 6:7],
        main ='Box Plots for Blood parameters')[, 6:7],
        main ='Box Plots for Blood parameters')
```



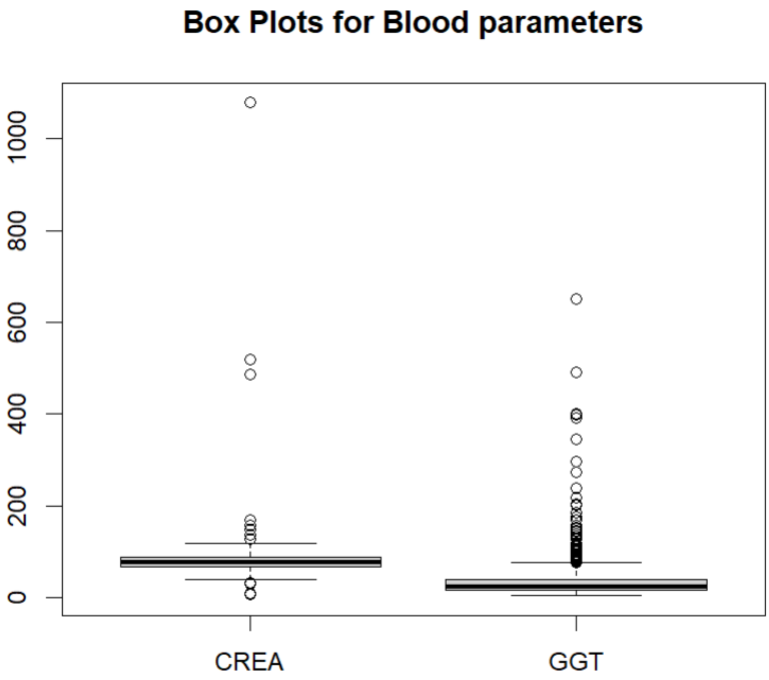
```
boxplot(boxplot(HepCdataset[, 8],
        main ='Box Plots for Blood parameters')[, 8],
        main ='Box Plots for Blood parameters')
```



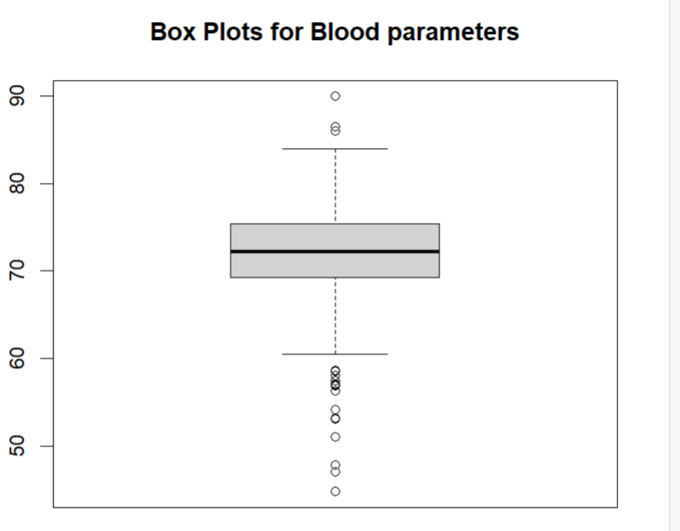
```
boxplot(boxplot(HepCdataset[, 9:10],
        main ='Box Plots for Blood parameters')[, 9:10],
        main ='Box Plots for Blood parameters')
```



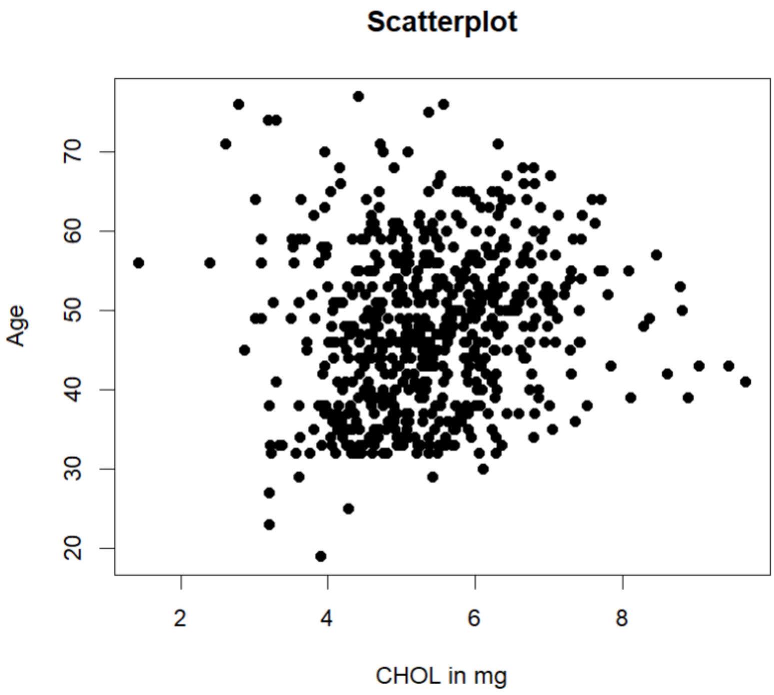
```
boxplot(boxplot(HepCdataset[,11:12],
        main ='Box Plots for Blood parameters')[,11:12],
        main ='Box Plots for Blood parameters')
```



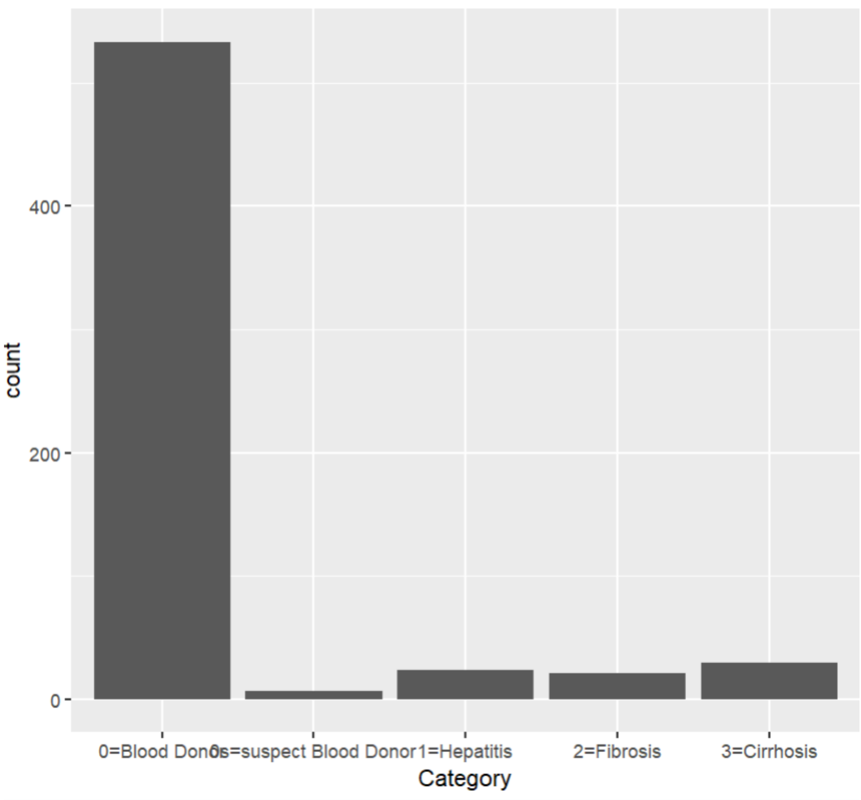
```
boxplot(boxplot(HepCdataset[, 13],
        main ='Box Plots for Blood parameters')[, 13],
        main ='Box Plots for Blood parameters')
```



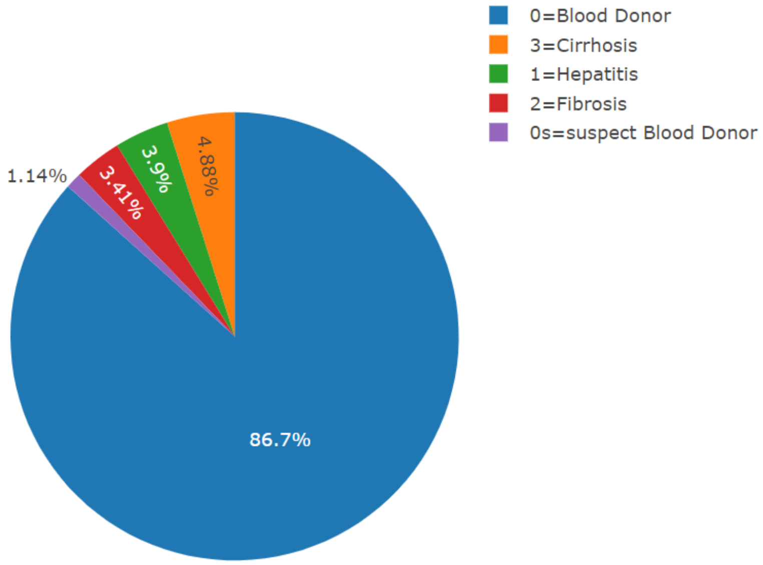
```
# Scatter plot for Ozone Concentration per month
plot(HepCdataset$CHOL, HepCdataset$Age,
     main ="Scatterplot",
     xlab ="CHOL in mg",
     ylab =" Age ", pch = 19)
```



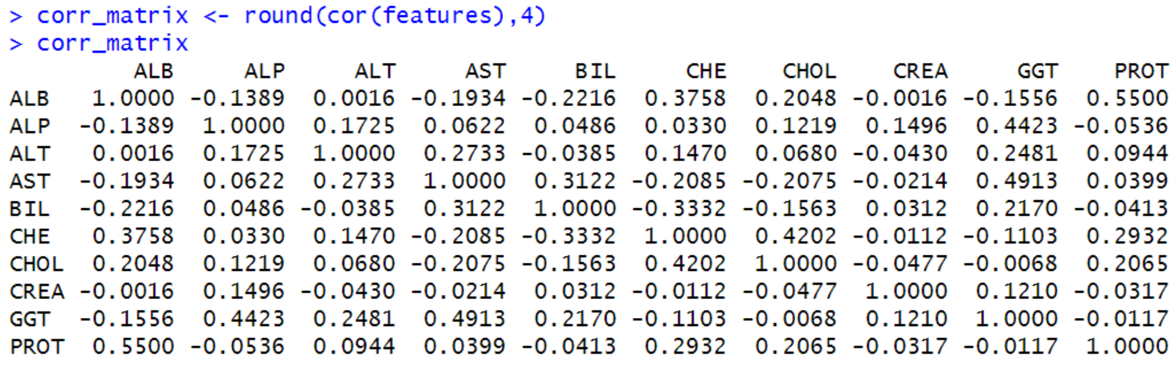
```
library(ggplot2)

ggplot(data=dataset, aes(x = Category)) +  geom_bar()
```



```
library(plotly)

plot_ly(data = dataset, labels = ~Category, type = "pie")
```



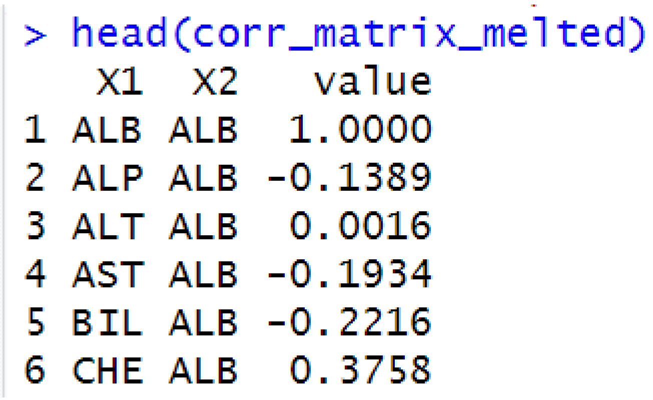
>>Correlation Matrix



```
> corr_matrix <- round(cor(features),4)
> corr_matrix
         ALB     ALP     ALT     AST     BIL     CHE    CHOL    CREA     GGT    PROT
ALB   1.0000 -0.1389  0.0016 -0.1934 -0.2216  0.3758  0.2048 -0.0016 -0.1556  0.5500
ALP  -0.1389  1.0000  0.1725  0.0622  0.0486  0.0330  0.1219  0.1496  0.4423 -0.0536
ALT   0.0016  0.1725  1.0000  0.2733 -0.0385  0.1470  0.0680 -0.0430  0.2481  0.0944
AST  -0.1934  0.0622  0.2733  1.0000  0.3122 -0.2085 -0.2075 -0.0214  0.4913  0.0399
BIL  -0.2216  0.0486 -0.0385  0.3122  1.0000 -0.3332 -0.1563  0.0312  0.2170 -0.0413
CHE   0.3758  0.0330  0.1470 -0.2085 -0.3332  1.0000  0.4202 -0.0112 -0.1103  0.2932
CHOL  0.2048  0.1219  0.0680 -0.2075 -0.1563  0.4202  1.0000 -0.0477 -0.0068  0.2065
CREA -0.0016  0.1496 -0.0430 -0.0214  0.0312 -0.0112 -0.0477  1.0000  0.1210 -0.0317
GGT  -0.1556  0.4423  0.2481  0.4913  0.2170 -0.1103 -0.0068  0.1210  1.0000 -0.0117
PROT  0.5500 -0.0536  0.0944  0.0399 -0.0413  0.2932  0.2065 -0.0317 -0.0117  1.0000
```

corr_matrix_melted <- melt(corr_matrix)

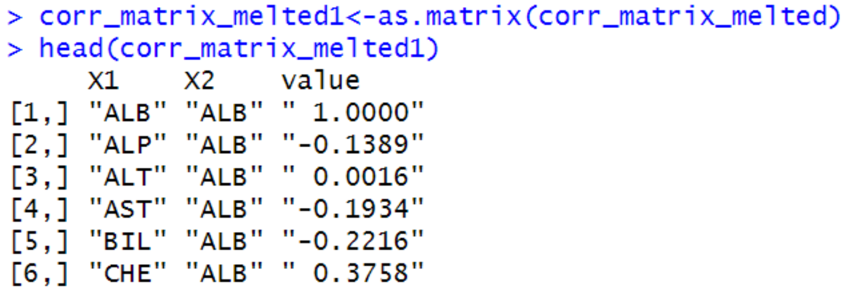
head(corr_matrix_melted)



```
> head(corr_matrix_melted)
   X1  X2   value
1 ALB ALB  1.0000
2 ALP ALB -0.1389
3 ALT ALB  0.0016
4 AST ALB -0.1934
5 BIL ALB -0.2216
6 CHE ALB  0.3758
```

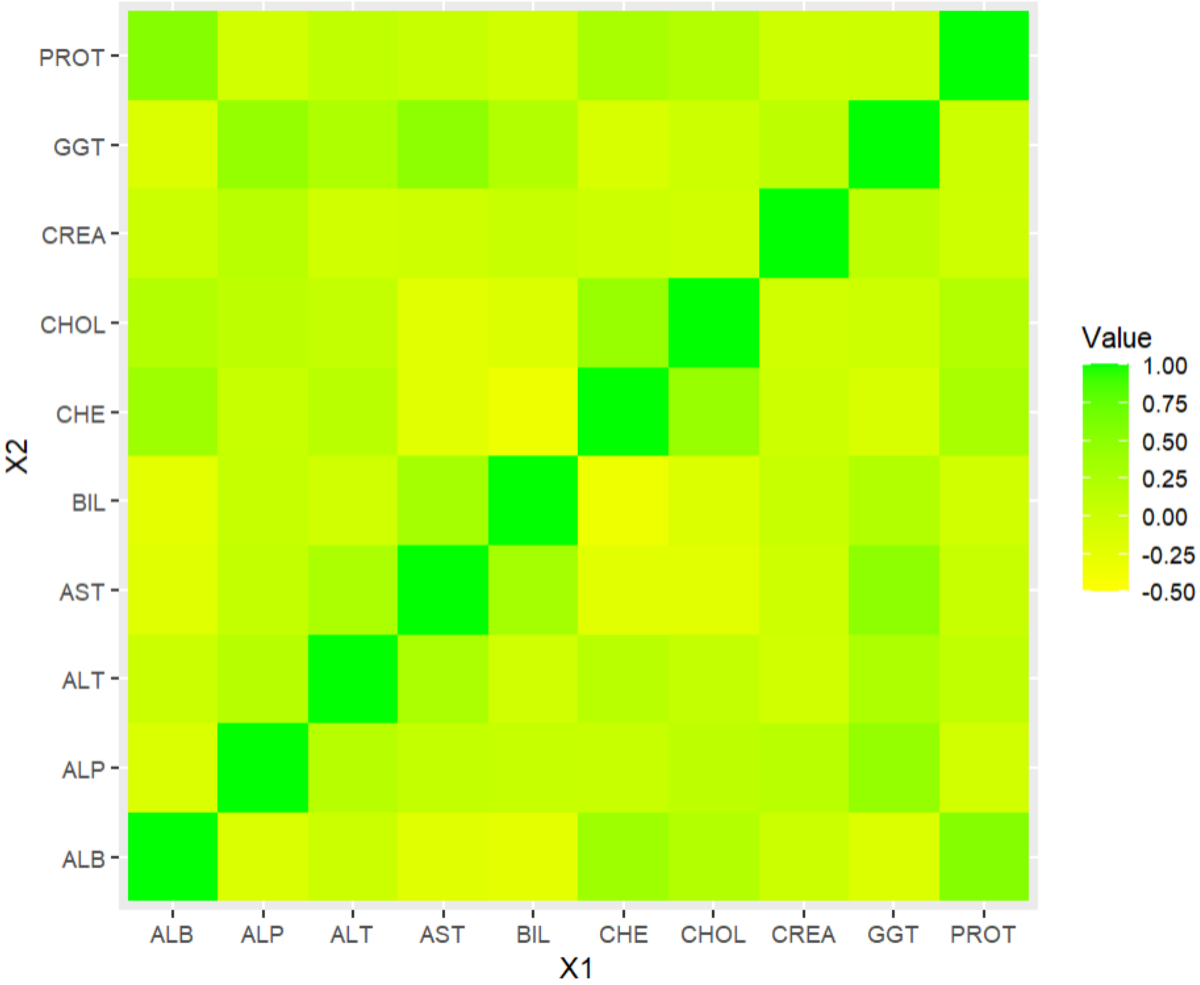
corr_matrix_melted1<-as.matrix(corr_matrix_melted)

head(corr_matrix_melted1)



```
> corr_matrix_melted1<-as.matrix(corr_matrix_melted)
> head(corr_matrix_melted1)
     X1    X2    value
[1,] "ALB" "ALB" " 1.0000"
[2,] "ALP" "ALB" "-0.1389"
[3,] "ALT" "ALB" " 0.0016"
[4,] "AST" "ALB" "-0.1934"
[5,] "BIL" "ALB" "-0.2216"
[6,] "CHE" "ALB" " 0.3758"
```

```
ggplot(corr_matrix_melted, aes(x=X1, y=X2, fill=value)) +
geom_tile()+scale_fill_gradient('Value', limits=c(-0.5, 1), breaks = c(-0.5,-0.25,0, 0.25, 0.5,
0.75, 1),  low = "yellow", high = "green")
```

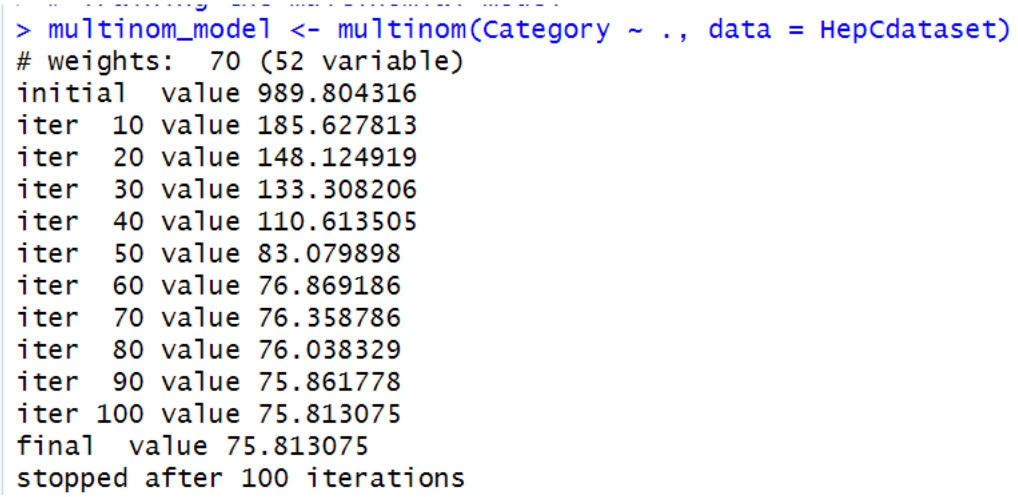


>>Applying Multinominal Logistic regression

```
# Split the data into training and test set
set.seed(123)
training.samples <- HepCdataset$Category %>%
  createDataPartition(p = 0.8, list = FALSE)
train.data  <- HepCdataset[training.samples, ]
test.data <- HepCdataset[-training.samples, ]

library(nnet)
# Setting the reference
train.data$Category <- relevel(train.data$Category,ref = "Blood Donor")
```

# Training the multinomial model

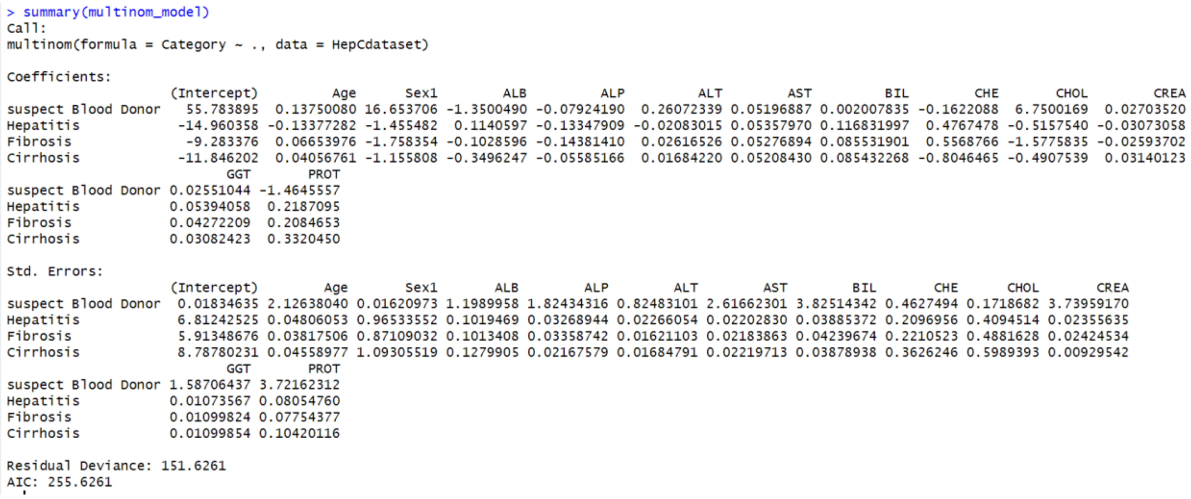
multinom_model <- multinom(Category ~ ., data = HepCdataset)



```
> multinom_model <- multinom(Category ~ ., data = HepCdataset)
# weights:  70 (52 variable)
initial  value 989.804316 
iter  10 value 185.627813
iter  20 value 148.124919
iter  30 value 133.308206
iter  40 value 110.613505
iter  50 value 83.079898
iter  60 value 76.869186
iter  70 value 76.358786
iter  80 value 76.038329
iter  90 value 75.861778
iter 100 value 75.813075
final  value 75.813075 
stopped after 100 iterations
```

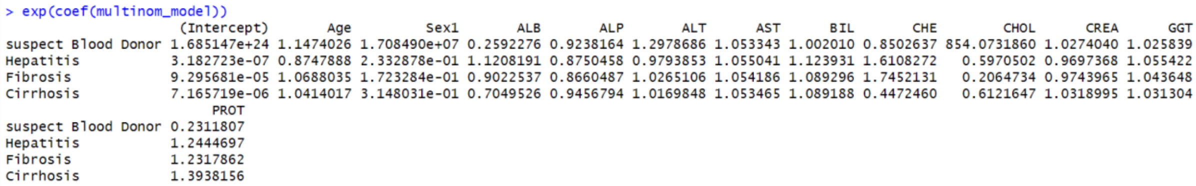
# Checking the model

summary(multinom_model)



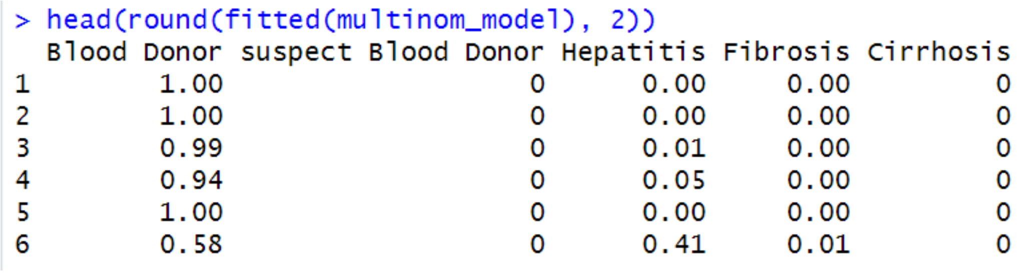
```
> summary(multinom_model)
Call:
multinom(formula = Category ~ ., data = HepCdataset)

Coefficients:
                    (Intercept)         Age      Sex1        ALB         ALP         ALT        AST         BIL        CHE       CHOL        CREA
suspect Blood Donor   55.783895  0.13750080 16.653706 -1.3500490 -0.07924190  0.26072339 0.05196887 0.002007835 -0.1622088  6.7500169  0.02703520
Hepatitis            -14.960358 -0.13377282 -1.455482  0.1140597 -0.13347909 -0.02083015 0.05357970 0.116831997  0.4767478 -0.5157540 -0.03073058
Fibrosis              -9.283376  0.06653976 -1.758354 -0.1028596 -0.14381410  0.02616526 0.05276894 0.085531901  0.5568766 -1.5775835 -0.02593702
Cirrhosis            -11.846202  0.04056761 -1.155808 -0.3496247 -0.05585166  0.01684220 0.05208430 0.085432268 -0.8046465 -0.4907539  0.03140123
                           GGT       PROT
suspect Blood Donor 0.02551044 -1.4645557
Hepatitis           0.05394058  0.2187095
Fibrosis            0.04272209  0.2084653
Cirrhosis           0.03082423  0.3320450

Std. Errors:
                    (Intercept)        Age       Sex1       ALB        ALP        ALT        AST        BIL       CHE      CHOL       CREA
suspect Blood Donor  0.01834635 2.12638040 0.01620973 1.1989958 1.82434316 0.82483101 2.61662301 3.82514342 0.4627494 0.1718682 3.73959170
Hepatitis            6.81242525 0.04806053 0.96533552 0.1019469 0.03268944 0.02266054 0.02202830 0.03885372 0.2096956 0.4094514 0.02355635
Fibrosis             5.91348676 0.03817506 0.87109032 0.1013408 0.03358742 0.01621103 0.02183863 0.04239674 0.2210523 0.4881628 0.02424534
Cirrhosis            8.78780231 0.04558977 1.09305519 0.1279905 0.02167579 0.01684791 0.02219713 0.03878938 0.3626246 0.5989393 0.00929542
                           GGT       PROT
suspect Blood Donor 1.58706437 3.72162312
Hepatitis           0.01073567 0.08054760
Fibrosis            0.01099824 0.07754377
Cirrhosis           0.01099854 0.10420116

Residual Deviance: 151.6261 
AIC: 255.6261 
```

exp(coef(multinom_model))



```
> exp(coef(multinom_model))
                     (Intercept)       Age         Sex1       ALB       ALP       ALT      AST      BIL       CHE        CHOL      CREA      GGT
suspect Blood Donor 1.685147e+24 1.1474026 1.708490e+07 0.2592276 0.9238164 1.2978686 1.053343 1.002010 0.8502637 854.0731860 1.0274040 1.025839
Hepatitis           3.182723e-07 0.8747888 2.332878e-01 1.1208191 0.8750458 0.9793853 1.055041 1.123931 1.6108272   0.5970502 0.9697368 1.055422
Fibrosis            9.295681e-05 1.0688035 1.723284e-01 0.9022537 0.8660487 1.0265106 1.054186 1.089296 1.7452131   0.2064734 0.9743965 1.043648
Cirrhosis           7.165719e-06 1.0414017 3.148031e-01 0.7049526 0.9456794 1.0169848 1.053465 1.089188 0.4472460   0.6121647 1.0318995 1.031304
                         PROT
suspect Blood Donor 0.2311807
Hepatitis           1.2444697
Fibrosis            1.2317862
Cirrhosis           1.3938156
```

head(round(fitted(multinom_model), 2))



```
> head(round(fitted(multinom_model), 2))
  Blood Donor suspect Blood Donor Hepatitis Fibrosis Cirrhosis
1        1.00                   0      0.00     0.00         0
2        1.00                   0      0.00     0.00         0
3        0.99                   0      0.01     0.00         0
4        0.94                   0      0.05     0.00         0
5        1.00                   0      0.00     0.00         0
6        0.58                   0      0.41     0.01         0
```

Predicting the values for train dataset

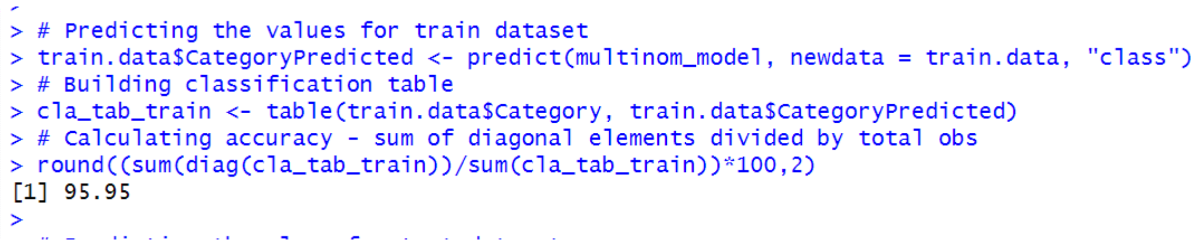
train.data$CategoryPredicted <- predict(multinom_model, newdata = train.data, "class")

Building classification table

cla_tab_train <- table(train.data$Category, train.data$CategoryPredicted)

Calculating accuracy - sum of diagonal elements divided by total obs

round((sum(diag(cla_tab_train))/sum(cla_tab_train))*100,2)



```
> # Predicting the values for train dataset
> train.data$CategoryPredicted <- predict(multinom_model, newdata = train.data, "class")
> # Building classification table
> cla_tab_train <- table(train.data$Category, train.data$CategoryPredicted)
> # Calculating accuracy - sum of diagonal elements divided by total obs
> round((sum(diag(cla_tab_train))/sum(cla_tab_train))*100,2)
[1] 95.95
>
```

Predicting the class for test dataset

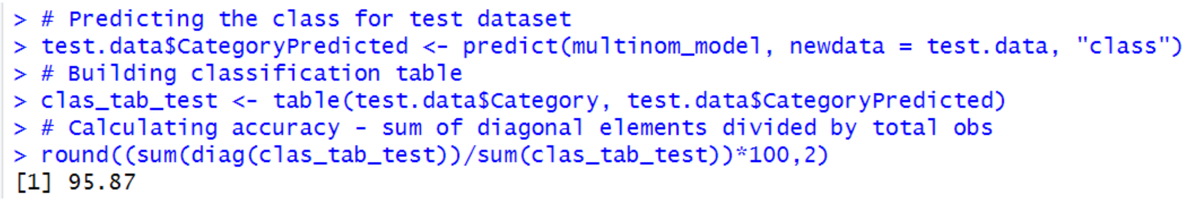
test.data$CategoryPredicted <- predict(multinom_model, newdata = test.data, "class")

Building classification table

clas_tab_test <- table(test.data$Category, test.data$CategoryPredicted)

Calculating accuracy - sum of diagonal elements divided by total obs

round((sum(diag(clas_tab_test))/sum(clas_tab_test))*100,2)



```
> # Predicting the class for test dataset
> test.data$CategoryPredicted <- predict(multinom_model, newdata = test.data, "class")
> # Building classification table
> clas_tab_test <- table(test.data$Category, test.data$CategoryPredicted)
> # Calculating accuracy - sum of diagonal elements divided by total obs
> round((sum(diag(clas_tab_test))/sum(clas_tab_test))*100,2)
[1] 95.87
```

Considered model worked well on both train and test data with 95.95% and 95.87% accuracy respectively.