Indian Liver Patient - CYO Project

Melissa Mayer

2/26/2022

Introduction

This project utilizes the Indian Liver Patient dataset which consists of medical records collected from North East Andhra Pradesh, India (https://www.kaggle.com/uciml/indian-liver-patient-records).

The motivation for the project is the increasing number of people with liver disease due to excessive alcohol consumption, environmental toxins, contaminated food and drugs. The dataset was created for the purpose of finding a predictive algorithm that would aid doctors in identifying patients with liver disease. That dataset consists of 583 patient records (416 liver patients and 167 non-liver patients) and 10 variables that potentially impact whether someone has liver disease. During this analysis, the 10 variables were examined in both the liver and non-liver groups to look for any key differences in results. Machine learning techniques were applied to develop two algorithms which demonstrate the impact of the different variables in predicting liver disease. The data was partitioned into a 20% test set for the purposes of testing the algorithms. The final models that are used in this analysis are LDA and Random Forest.

Background Information

According to Wikipedia, Andhra Pradesh is a state in the southeast coastal region of India. It is the seventh largest state by area and tenth largest in population. The northeast region borders the Bay of Bengal. The state's main industries are agriculture and livestock. Seventy percent of the population is rural. It has a female to male ratio of 996 to 1000 which is higher than the national average of 926 per 1000. The Indian Liver Patient dataset consists of 441 male patients and 142 female patients which is not representative of the gender composition of the state. Most of the population is Hindu with a minority of Muslims. In addition, the literacy rate in 2016 was 67.4% expected to rise to over 90% by 2021. Greater literacy could be an important factor in improving health outcomes.

According to the WHO, liver disease is the tenth leading cause of death in India and may affect 1 in 5 Indians. In 2017, liver cirrhosis was the 14th leading cause of death in the world and expected to increase by 2020. Once a patient is diagnosed with liver disease, it is unlikely to be reversed. Treatment can be very costly, and some patients may not be able to afford it. Patients may only survive a few years without a liver transplant. In India, the demand for organs greatly exceeds the supply. Fatty liver disease, prevalent in diabetics, as well as alcohol abuse, have become the leading causes of cirrhosis whereas it was previously Hepatitis B and C. Furthermore, the age of alcoholic liver disease has become lower. Most of the patients range in age from 40's to 60's, although the most common age of developing alcoholic liver disease in India has decreased to age 30-40. In the West, the average age range is 45-55. The mean age of our dataset is 44.7 and ranged from age 4-90. Due to the aforementioned factors, it is important to investigate this problem. Furthermore, lifestyle modification can play a big role in the prevention of liver disease, so identification of the patients can lead to intervention to prevent the worsening of the disease. (https://timesofindia.indiatimes.com/life-style/health-fitness/health-news/is-liver-disease-the-next-major-lifestyle-disease-of-india-after-diabetes-and-bp/articleshow/58122706.cms)

We are going to install any missing packages that are needed for the project: tidyverse, readxl, caret, and randomForest

```
if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
if(!require(readxl)) install.packages("readxl", repos = "http://cran.us.r-project.org")
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-project.org")
if(!require(randomForest)) install.packages("randomForest", repos = "http://cran.us.r-project.org")
```

Next, we will load the libraries that will be used in model development

```
library(tidyverse)
library(readxl)
library(caret)
library(randomForest)
```

Import the dataset "indian_liver_patient.csv" which comes from this website: https://www.kaggle.com/uciml/indian-liver-patient-records

```
dat <- read_csv("cyo-project/indian_liver_patient.csv", show_col_types = FALSE)
View(dat)
as_tibble(dat)
```

```
## # A tibble: 583 x 11
##      Age Gender Total_Bilirubin Direct_Bilirubin Alkaline_Phosphotase
##    <dbl> <chr>            <dbl>            <dbl>                <dbl>
##  1    65 Female             0.7              0.1                  187
##  2    62 Male              10.9              5.5                  699
##  3    62 Male               7.3              4.1                  490
##  4    58 Male               1                0.4                  182
##  5    72 Male               3.9              2                    195
##  6    46 Male               1.8              0.7                  208
##  7    26 Female             0.9              0.2                  154
##  8    29 Female             0.9              0.3                  202
##  9    17 Male               0.9              0.3                  202
## 10    55 Male               0.7              0.2                  290
## # ... with 573 more rows, and 6 more variables: Alamine_Aminotransferase <dbl>,
## #   Aspartate_Aminotransferase <dbl>, Total_Protiens <dbl>, Albumin <dbl>,
## #   Albumin_and_Globulin_Ratio <dbl>, Dataset <dbl>
```

This file contains the standard ranges for the lab tests that are part of the dataset

```
dat2 <- read_excel("CYO-Project/Indian_Liver_Patient_Tests.xlsx")
as_tibble(dat2)
```

```
## # A tibble: 9 x 2
```

```
##   `Liver Function Test`      `Standard Range`
##   <chr>                      <chr>
## 1 Albumin                    3.8 - 4.8 g/dL
## 2 Total Protein              6.0 - 8.5 g/dL
## 3 Globulin                   1.5 - 4.5 g/dL
## 4 A/G Ratio                  1.2 - 2.2
## 5 Total Bilirubin            0.0 - 1.2 mg/dL
## 6 Direct Bilirubin           0.3 mg/dL
## 7 Alakaline Phosphatase      44 - 121 IU/L
## 8 Aspartate Aminotransferase 0 - 40 IU/L
## 9 Alanine Aminotransferase   0 - 32 IU/L
```

There are 2 misspelled column names:

"Alamine_Aminotransferase" and "Total_Protiens"

Change to "Alanine_Aminotransferase" and "Total_Proteins"

```
dat <- dat %>%
  set_names(c("Age", "Gender", "Total_Bilirubin",
                               "Direct_Bilirubin", "Alkaline_Phosphatase",
                               "Alanine_Aminotransferase",
                               "Aspartate_Aminotransferase", "Total_Proteins",
                               "Albumin", "Albumin_and_Globulin_Ratio",
                               "Dataset"))
```

Change the Gender variable from a Character to a Factor for better interpretation of the data

```
dat <- dat %>% mutate(Gender = as.factor(Gender))
```

Substitute any missing values in the Albumin_and_Globulin_Ratio column with the median value for the variable (there are a total of 4 NA's)

```
dat <- dat %>%
  mutate(Albumin_and_Globulin_Ratio = ifelse(is.na(Albumin_and_Globulin_Ratio),
                                             median(Albumin_and_Globulin_Ratio,
                                                    na.rm = TRUE),
                                             Albumin_and_Globulin_Ratio))
summary(dat)
```

```
##       Age           Gender    Total_Bilirubin  Direct_Bilirubin
##  Min.   : 4.00   Female:142   Min.   : 0.400   Min.   : 0.100
##  1st Qu.:33.00   Male  :441   1st Qu.: 0.800   1st Qu.: 0.200
##  Median :45.00                Median : 1.000   Median : 0.300
##  Mean   :44.75                Mean   : 3.299   Mean   : 1.486
##  3rd Qu.:58.00                3rd Qu.: 2.600   3rd Qu.: 1.300
```

```
##  Max.   :90.00                Max.   :75.000   Max.   :19.700
##  Alkaline_Phosphatase Alanine_Aminotransferase Aspartate_Aminotransferase
##  Min.   :  63.0       Min.   :  10.00          Min.   :  10.0
##  1st Qu.: 175.5       1st Qu.:  23.00          1st Qu.:  25.0
##  Median : 208.0       Median :  35.00          Median :  42.0
##  Mean   : 290.6       Mean   :  80.71          Mean   : 109.9
##  3rd Qu.: 298.0       3rd Qu.:  60.50          3rd Qu.:  87.0
##  Max.   :2110.0       Max.   :2000.00          Max.   :4929.0
##  Total_Proteins     Albumin      Albumin_and_Globulin_Ratio    Dataset
##  Min.   :2.700   Min.   :0.900   Min.   :0.3000             Min.   :1.000
##  1st Qu.:5.800   1st Qu.:2.600   1st Qu.:0.7000             1st Qu.:1.000
##  Median :6.600   Median :3.100   Median :0.9300             Median :1.000
##  Mean   :6.483   Mean   :3.142   Mean   :0.9469             Mean   :1.286
##  3rd Qu.:7.200   3rd Qu.:3.800   3rd Qu.:1.1000             3rd Qu.:2.000
##  Max.   :9.600   Max.   :5.500   Max.   :2.8000             Max.   :2.000
```

Methods/Analysis

As stated above, the dataset consists of 10 variables in addition to 1 outcome variable labeled "Dataset" which classifies the cases as liver (1) or non-liver (2). The variables are as follows: Age of the patient, Gender of the patient, Total Bilirubin, Direct Bilirubin, Alkaline Phosphatase, Alanine Aminotransferase (misspelled as Alamine), Aspartate Aminotransferase, Total Proteins (misspelled as Protiens), Albumin, and Albumin and Globulin Ratio. The Age variable lists anyone over age 89 as age 90. I obtained reference ranges for these lab tests from my medical provider, Mount Sinai Medical Center in New York City (except for Direct Bilirubin). It should be noted that other laboratories may use slightly different ranges. *According to the Mayo Clinic, 0.3 mg/dL is considered normal for Direct Bilirubin. Higher levels may indicate that your body is not clearing bilirubin properly. https://www.mayoclinic.org/tests-procedures/bilirubin/about/pac-20393041

Bilirubin is tested to check the health of the liver. Higher than normal levels might indicate liver or bile duct problems. A liver function test checks the enzymes and protein levels of the blood. Alkaline Phosphatase, Aspartate Aminotransferase, and Alanine Aminotransferase are enzymes found in the liver. Elevated levels could be a sign of liver damage. Albumin and Globulin are proteins made by the liver. Total Proteins consists of Albumin and Globulin. https://my.clevelandclinic.org/health/diagnostics/22058-comprehensive-metabolic-panel-cmp

After making the Gender variable into a factor and substituting the median value for the NA's in Albumin_and_Globulin column, the data was summarized. The dataset can be broken up into Liver (Dataset == 1) and Non-liver (Dataset == 2) groups to inspect the differences between the two groups.

Filter the data by the Dataset column (Liver_Pt = 1, Nonliver_Pt = 2)

Summarize each grouping of data to observe any differences

```
Liver_Pt <- dat %>% filter(Dataset == 1)
summary(Liver_Pt)
```

```
##       Age           Gender    Total_Bilirubin  Direct_Bilirubin
##  Min.   : 7.00   Female: 92   Min.   : 0.400   Min.   : 0.100
##  1st Qu.:34.00   Male  :324   1st Qu.: 0.800   1st Qu.: 0.200
##  Median :46.00                Median : 1.400   Median : 0.500
##  Mean   :46.15                Mean   : 4.164   Mean   : 1.924
```

```
##  3rd Qu.:58.00               3rd Qu.: 3.625   3rd Qu.: 1.800
##  Max.   :90.00               Max.   :75.000   Max.   :19.700
##  Alkaline_Phosphatase Alanine_Aminotransferase Aspartate_Aminotransferase
##  Min.   :  63.0       Min.   :  12.00          Min.   :  11.00
##  1st Qu.: 186.0       1st Qu.:  25.00          1st Qu.:  29.75
##  Median : 229.0       Median :  41.00          Median :  52.50
##  Mean   : 319.0       Mean   :  99.61          Mean   : 137.70
##  3rd Qu.: 315.2       3rd Qu.:  76.50          3rd Qu.: 108.75
##  Max.   :2110.0       Max.   :2000.00          Max.   :4929.00
##  Total_Proteins     Albumin      Albumin_and_Globulin_Ratio    Dataset
##  Min.   :2.700   Min.   :0.900   Min.   :0.3000             Min.   :1
##  1st Qu.:5.700   1st Qu.:2.500   1st Qu.:0.7000             1st Qu.:1
##  Median :6.550   Median :3.000   Median :0.9000             Median :1
##  Mean   :6.459   Mean   :3.061   Mean   :0.9143             Mean   :1
##  3rd Qu.:7.200   3rd Qu.:3.625   3rd Qu.:1.1000             3rd Qu.:1
##  Max.   :9.600   Max.   :5.500   Max.   :2.8000             Max.   :1
```

```
Nonliver_Pt <- dat %>% filter(Dataset == 2)
summary(Nonliver_Pt)
```

```
##       Age           Gender    Total_Bilirubin Direct_Bilirubin
##  Min.   : 4.00   Female: 50   Min.   :0.500   Min.   :0.1000
##  1st Qu.:28.00   Male  :117   1st Qu.:0.700   1st Qu.:0.2000
##  Median :40.00                Median :0.800   Median :0.2000
##  Mean   :41.24                Mean   :1.143   Mean   :0.3964
##  3rd Qu.:55.00                3rd Qu.:1.100   3rd Qu.:0.3500
##  Max.   :85.00                Max.   :7.300   Max.   :3.6000
##  Alkaline_Phosphatase Alanine_Aminotransferase Aspartate_Aminotransferase
##  Min.   :  90.0       Min.   : 10.00           Min.   : 10.00
##  1st Qu.: 161.5       1st Qu.: 20.00           1st Qu.: 21.00
##  Median : 186.0       Median : 27.00           Median : 29.00
##  Mean   : 219.8       Mean   : 33.65           Mean   : 40.69
##  3rd Qu.: 213.0       3rd Qu.: 37.50           3rd Qu.: 43.50
##  Max.   :1580.0       Max.   :181.00           Max.   :285.00
##  Total_Proteins     Albumin      Albumin_and_Globulin_Ratio    Dataset
##  Min.   :3.700   Min.   :1.400   Min.   :0.370              Min.   :2
##  1st Qu.:5.900   1st Qu.:2.900   1st Qu.:0.900              1st Qu.:2
##  Median :6.600   Median :3.400   Median :1.000              Median :2
##  Mean   :6.543   Mean   :3.344   Mean   :1.028              Mean   :2
##  3rd Qu.:7.300   3rd Qu.:4.000   3rd Qu.:1.200              3rd Qu.:2
##  Max.   :9.200   Max.   :5.000   Max.   :1.900              Max.   :2
```

Before developing our two models, we will first explore the data using machine learning techniques. The first attempt at applying machine learning is to randomly guess whether a patient will be in the liver or non-liver group. Not surprisingly, this yields an accuracy of around 50% (0.5128205).

Predict Liver Disease by Guessing

```
y <- dat$Dataset
set.seed(1, sample.kind = "Rounding")
```

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

Partition the data into a 20% test set based on the Dataset column

```
test_index <- createDataPartition(y, times = 1, p = 0.2, list = FALSE)
test_set <- dat[test_index, ]
train_set <- dat[-test_index, ]
```

Guess by sampling from the two options of Dataset = 1 or Dataset = 2

```
guess <- sample(c(1,2), nrow(test_set), replace = TRUE)
```

Get the accuracy of the prediction and create a table to show the results

```
accuracy1 <- mean(guess == test_set$Dataset)
accuracy1
```

```
## [1] 0.5128205
```

```
accuracy_results <- tibble(Method = "Guessing", Accuracy = accuracy1)
```

Next, we predict the likelihood of having liver disease depending on Gender (Male = 0.730659, Female = 0.6239316). Although males are more likely to have liver disease than females, both groups are more likely than not to have liver disease. Gender alone will not be a likely determinant.

Obtain the likelihood of having Liver Disease in Males and Females from the Training Set

```
train_set %>%
  group_by(Gender) %>%
  summarize(Dataset = mean(Dataset == 1)) %>%
  filter(Gender == "Male") %>%
  pull(Dataset)
```

```
## [1] 0.730659
```

```
train_set %>%
  group_by(Gender) %>%
  summarize(Dataset = mean(Dataset == 1)) %>%
  filter(Gender == "Female") %>%
  pull(Dataset)
```

```
## [1] 0.6239316
```

The next attempt at applying machine learning techniques is to try to compute overall accuracy by taking one variable, Albumin, computing the mean and standard deviation and then determining a cut-off point which potentially makes someone more likely to have liver disease. In this scenario, we use the value 4.9 which is the mean Albumin level (3.34) for non-liver patients + 2 SD's (2 x 0.784). Anyone above this value is predicted to not be a liver patient. The accuracy obtained is 0.7118353.

Create variables x and y

```
x <- dat$Albumin
y <- dat$Dataset
```

Partition the data into a 20% test set based on the Dataset column

```
set.seed(1, sample.kind = "Rounding")
```

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

```
test_index <- createDataPartition(y, times = 1, p = 0.2, list = FALSE)
test_set <- dat[test_index, ]
train_set <- dat[-test_index, ]
```

Summarize the data for the Albumin variable for both Liver and Non-liver patients

```
dat %>% group_by(Dataset) %>% summarize(mean(Albumin), sd(Albumin))
```

```
## # A tibble: 2 x 3
##   Dataset `mean(Albumin)` `sd(Albumin)`
##     <dbl>           <dbl>         <dbl>
## 1       1            3.06         0.787
## 2       2            3.34         0.784
```

Determine a cutoff point of Albumin level to predict Liver Disease

```
y_hat <- ifelse(x > 4.9, 2, 1)
```

Obtain the accuracy of the prediction and add it to the results table

```
accuracy2 <- mean(y == y_hat)
accuracy2
```

```
## [1] 0.7118353
```

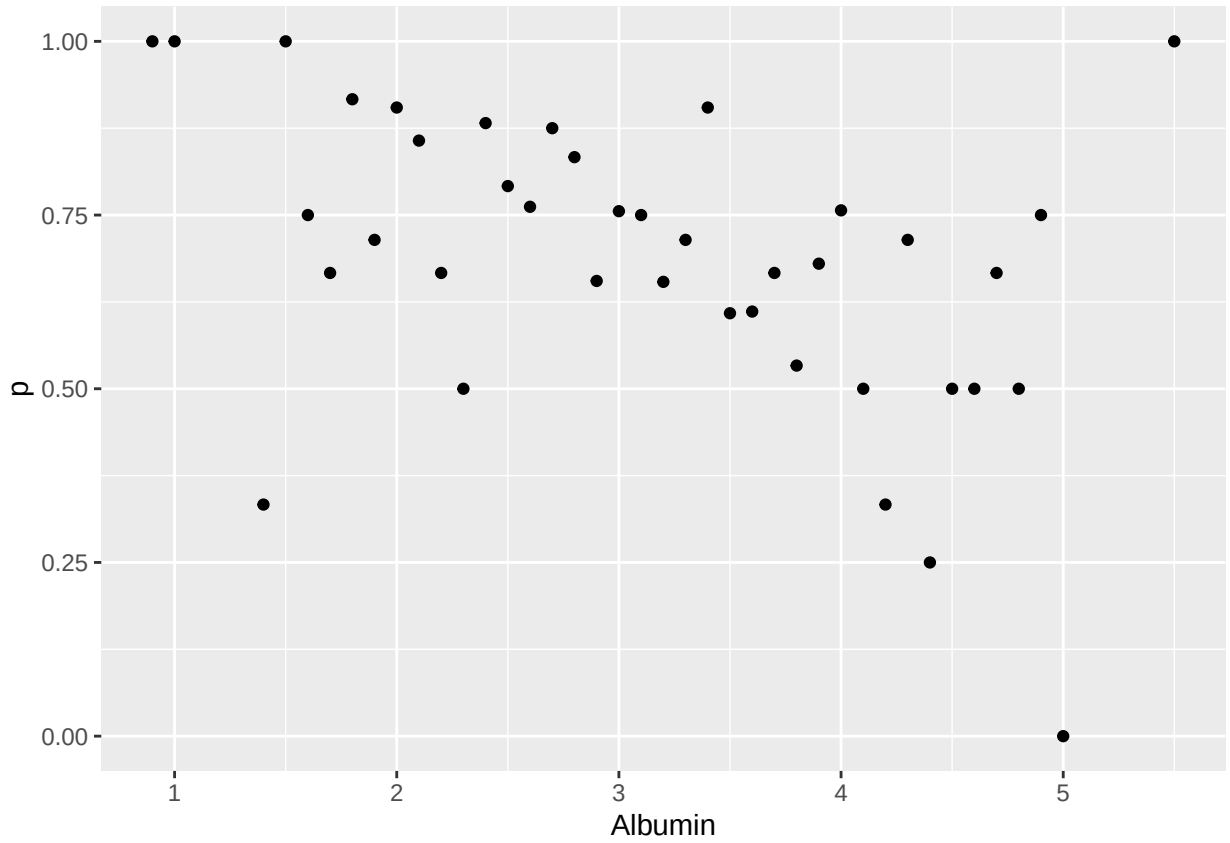
```
accuracy_results <- bind_rows(accuracy_results,
                              tibble(Method = "Albumin Cutoff",
                                     Accuracy = accuracy2))
accuracy_results %>% knitr::kable()
```

Method	Accuracy
Guessing	0.5128205
Albumin Cutoff	0.7118353

Next, we plot the conditional probability of having liver disease based on Albumin level. In order to reduce the variability, we then create quantiles so each group has the same number of points. It should be noted that in both the Liver and Non-liver groups, the mean Albumin level is lower than the standard range (3.8-4.8 g/dL). Lower albumin levels are associated with several conditions: liver disease, inflammation, shock, malnutrition, nephritic syndrome, Crohn's disease and celiac disease, so a lower than normal albumin level is not necessarily indicative of liver disease.

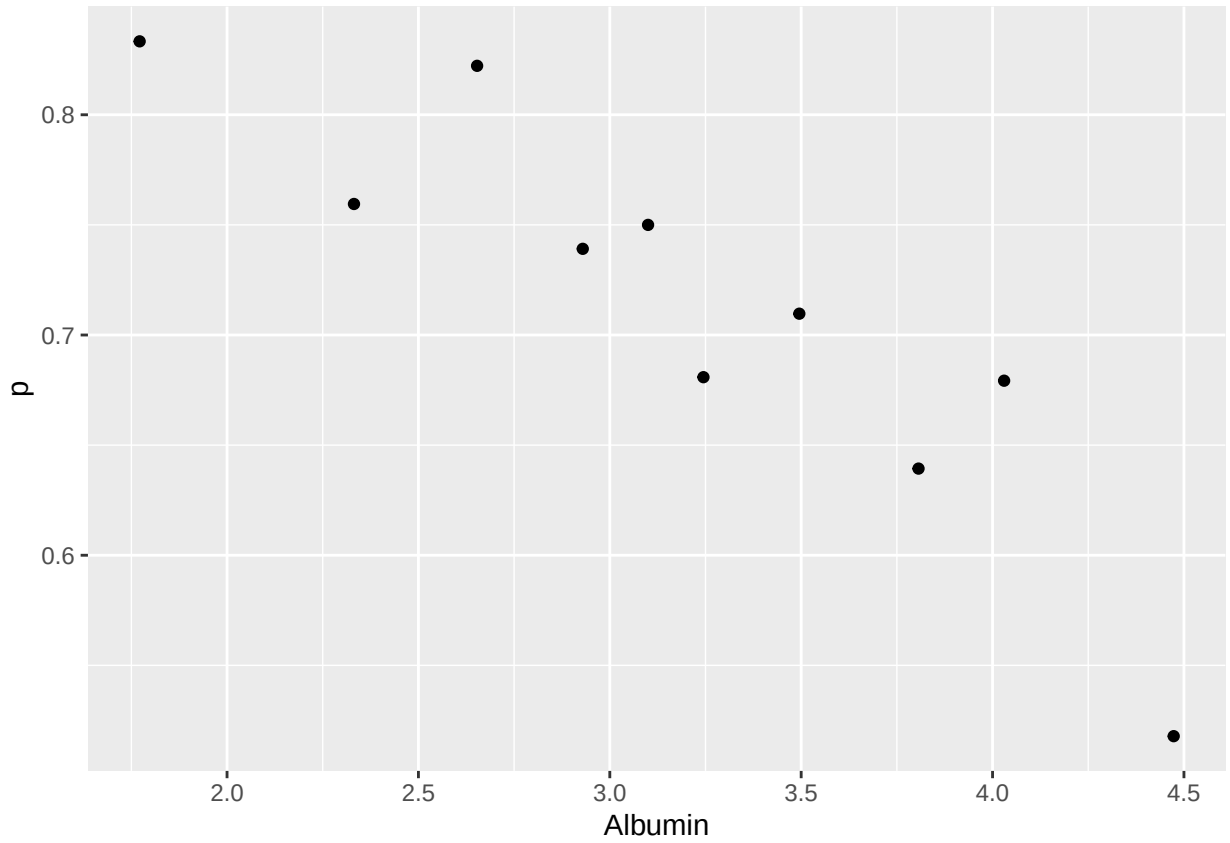
Make a plot of Albumin levels and the conditional probability of having Liver Disease (Dataset = 1)

```
dat %>% group_by(Albumin) %>% summarize(p = mean(Dataset == 1)) %>%
qplot(Albumin, p, data =.)
```



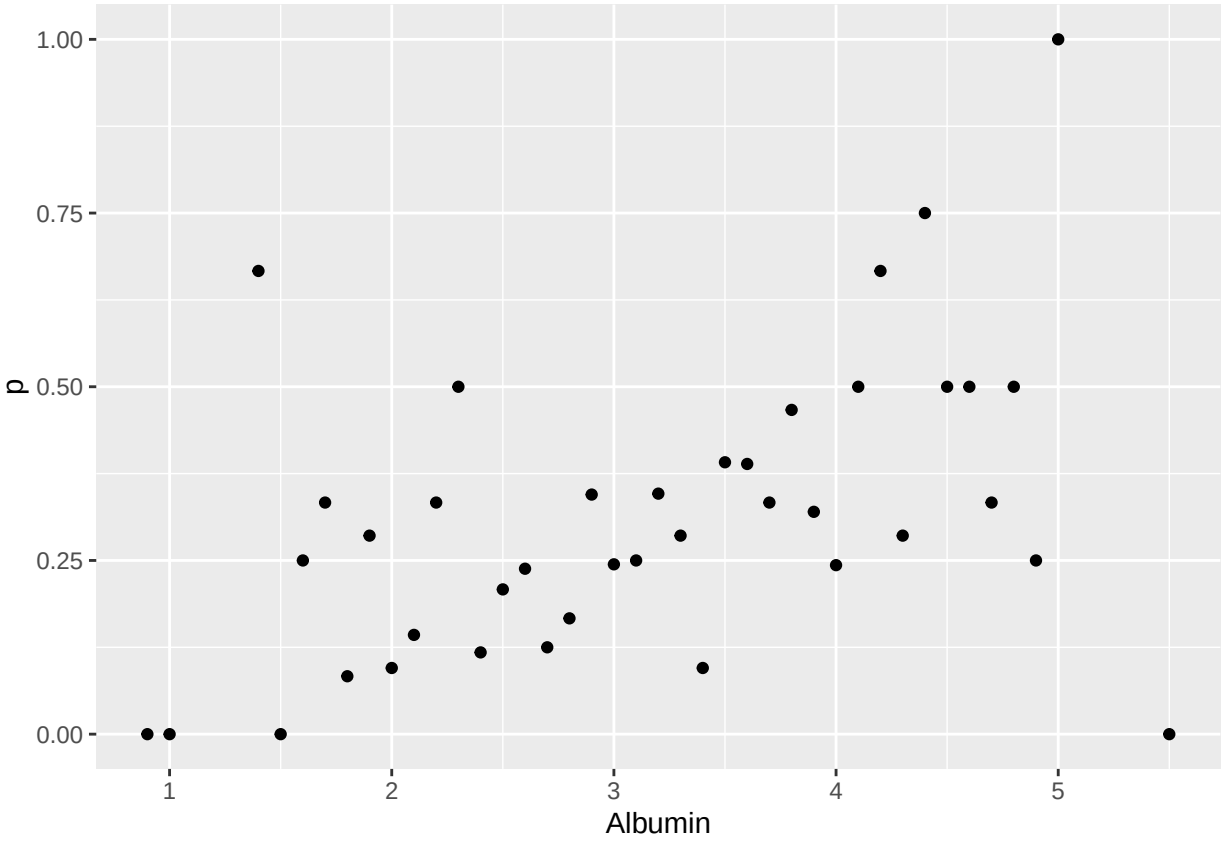
Create quantiles so there are equal # of points for each level of Albumin

```
ps <- seq(0, 1, 0.1)
dat %>%
mutate(g = cut(Albumin, quantile(Albumin, ps), include.lowest = TRUE)) %>%
group_by(g) %>%
summarize(p = mean(Dataset == 1), Albumin = mean(Albumin)) %>%
qplot(Albumin, p, data =.)
```



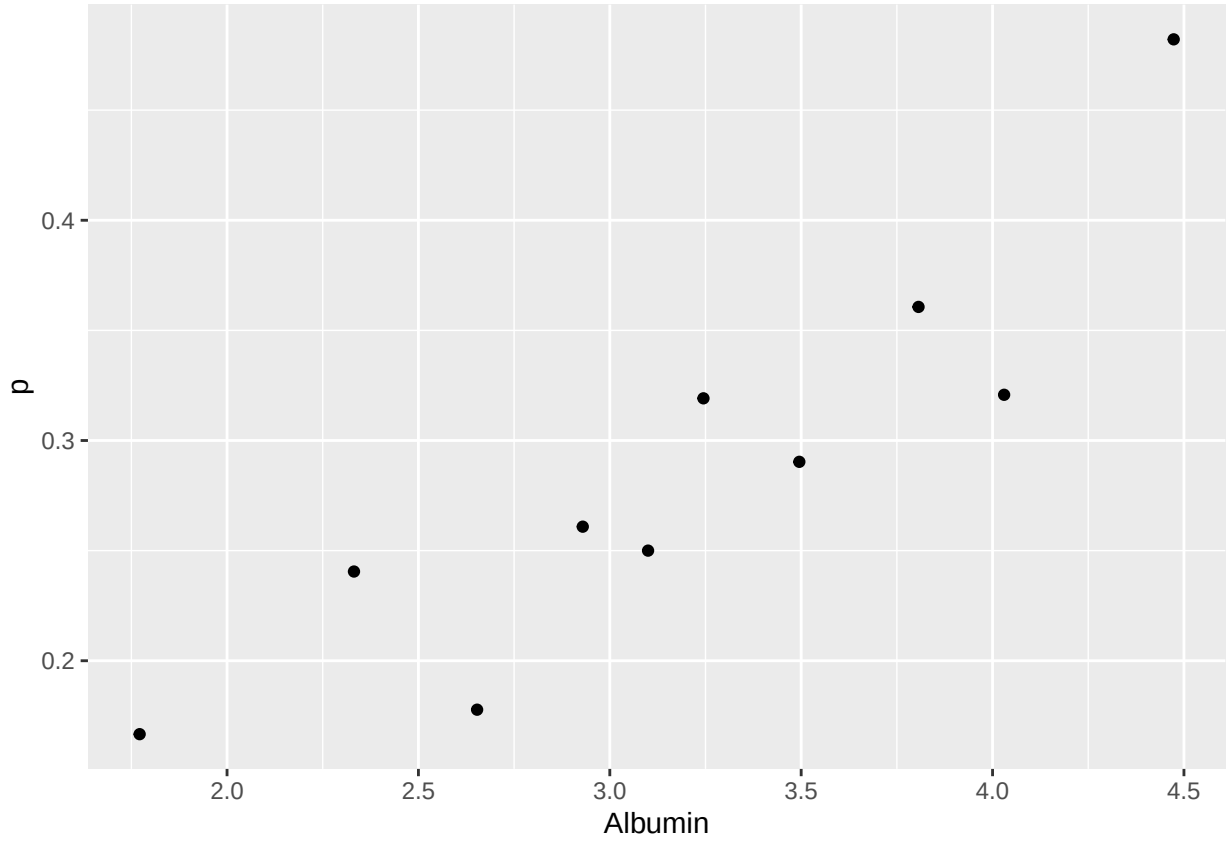
Make a plot of Albumin levels and the conditional probability of not having Liver Disease (Dataset = 2)

```
dat %>% group_by(Albumin) %>% summarize(p = mean(Dataset == 2)) %>%
qplot(Albumin, p, data =.)
```



Create quantiles so there are equal # of points for each level of Albumin

```
ps <- seq(0, 1, 0.1)
dat %>%
mutate(g = cut(Albumin, quantile(Albumin, ps), include.lowest = TRUE)) %>%
group_by(g) %>%
summarize(p = mean(Dataset == 2), Albumin = mean(Albumin)) %>%
qplot(Albumin, p, data =.)
```



Another commonality between the patient groups is higher than normal levels of Alkaline Phosphatase (mean(Liver_Pt) = 319.0, mean(Nonliver_Pt) = 219.8). The standard range is 44-121 IU/L. The presence of high Alkaline Phosphatase can be associated with liver disorders, bone disorders, cancer, kidney disorders, use of birth control pills, and hyperthyroidism, among many other causes. (https://labs.selfdecode.com/blog/alkaline-phosphatase/)

Calculate the mean levels of Alkaline Phosphatase for Liver and Non-liver Patients

```
mean(Liver_Pt$Alkaline_Phosphatase)
```

```
## [1] 319.0072
```

```
mean(Nonliver_Pt$Alkaline_Phosphatase)
```

```
## [1] 219.7545
```

Finally, we will discuss the 2 machine learning models that were developed for purposes of showing how this data can predict liver disease in patients. After exploring several machine learning techniques with different groupings of variables, the LDA method was found to yield a reasonable level of accuracy using all of the predictors in the dataset. The Random Forest method further improved upon the LDA model. In this model, 6 out of the 10 predictors (Total Bilirubin, Direct Bilirubin, Alkaline Phosphatase, Alanine Aminotransferase, Aspartate Aminotransferase, and Total Proteins) are used. These were chosen to be part of the model after trying all of the variables and then eliminating those which didn't add to the overall accuracy of 0.7627119. This will be elaborated on further in the Results section.

Results

Change the Dataset variable from (1,2) to (1,0): 1 = Liver Disease, 0 = No Liver Disease

```
dat$Dataset<-factor(dat$Dataset, levels = c(1,2), labels=c("1", "0"))
y <- dat$Dataset
set.seed(1, sample.kind = "Rounding")
```

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

Partition the data into a 20% test set based on the Dataset column

```
test_index <- createDataPartition(y, times = 1, p = 0.2, list = FALSE)
test_set <- dat[test_index, ]
train_set <- dat[-test_index, ]
```

Train an algorithm using the "LDA" method using all 10 variables as predictors

```
train_lda <- train(Dataset ~ ., method = "lda", data = train_set)
```

```
## Warning in (function (kind = NULL, normal.kind = NULL, sample.kind = NULL) :
## non-uniform 'Rounding' sampler used
```

```
lda_preds <- predict(train_lda, test_set)
```

Obtain the accuracy of the prediction and add it to the results table

```
accuracy3 <- mean(lda_preds == test_set$Dataset)
accuracy3
```

```
## [1] 0.720339
```

```
accuracy_results <- bind_rows(accuracy_results, tibble(Method = "LDA", Accuracy = accuracy3))
accuracy_results %>% knitr::kable()
```

Method	Accuracy
Guessing	0.5128205
Albumin Cutoff	0.7118353
LDA	0.7203390

View the Final Model

```
train_lda$finalModel
```

```
## Call:
## lda(x, grouping = y)
##
## Prior probabilities of groups:
##         1         0
## 0.7139785 0.2860215
##
## Group means:
##        Age GenderMale Total_Bilirubin Direct_Bilirubin Alkaline_Phosphatase
## 1 46.46084  0.7861446        4.096988        1.8819277             310.8825
## 0 40.25564  0.7218045        1.142857        0.3969925             220.9624
##   Alanine_Aminotransferase Aspartate_Aminotransferase Total_Proteins  Albumin
## 1                100.80120                  139.18072       6.455422 3.061446
## 0                 33.94737                   39.94737       6.536090 3.368421
##   Albumin_and_Globulin_Ratio
## 1                  0.9198795
## 0                  1.0450376
##
## Coefficients of linear discriminants:
##                                       LD1
## Age                         -0.0291692368
## GenderMale                  -0.2029101447
## Total_Bilirubin             -0.0028159624
## Direct_Bilirubin            -0.1494983601
## Alkaline_Phosphatase        -0.0013396355
## Alanine_Aminotransferase    -0.0026410176
## Aspartate_Aminotransferase   0.0005233221
## Total_Proteins              -0.5271260149
## Albumin                      0.8920099456
## Albumin_and_Globulin_Ratio -0.3542496495
```

Change the Dataset column from (1,0) to (Yes, No) where "Yes" is equivalent to having liver disease

```
dat$Dataset<-factor(dat$Dataset, levels = c(1,0), labels=c("Yes", "No"))
y <- dat$Dataset
set.seed(1, sample.kind = "Rounding")
```

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

Partition the data into a 20% test set based on the Dataset column

```
test_index <- createDataPartition(y, times = 1, p = 0.2, list = FALSE)
test_set <- dat[test_index, ]
train_set <- dat[-test_index, ]
```

Train a Random Forest on the data with 6 predictors and 50 trees

```
train_rf <- randomForest(Dataset ~ Total_Bilirubin + Direct_Bilirubin +
                           Alkaline_Phosphatase + Alanine_Aminotransferase +
                           Aspartate_Aminotransferase + Total_Proteins,
                         data = train_set, ntree = 50, importance = TRUE)
```

List the variables in order of importance in contributing to the model

```
varImp(train_rf)
```

```
##                                Yes        No
## Total_Bilirubin            1.3707421 1.3707421
## Direct_Bilirubin           2.8645895 2.8645895
## Alkaline_Phosphatase       1.9452695 1.9452695
## Alanine_Aminotransferase   2.1059935 2.1059935
## Aspartate_Aminotransferase 1.5006187 1.5006187
## Total_Proteins             0.9260783 0.9260783
```

Compute the accuracy of the model and add it to the results table

```
pred <- predict(train_rf,newdata=test_set)
accuracy4 <- mean(pred == test_set$Dataset)
accuracy4
```

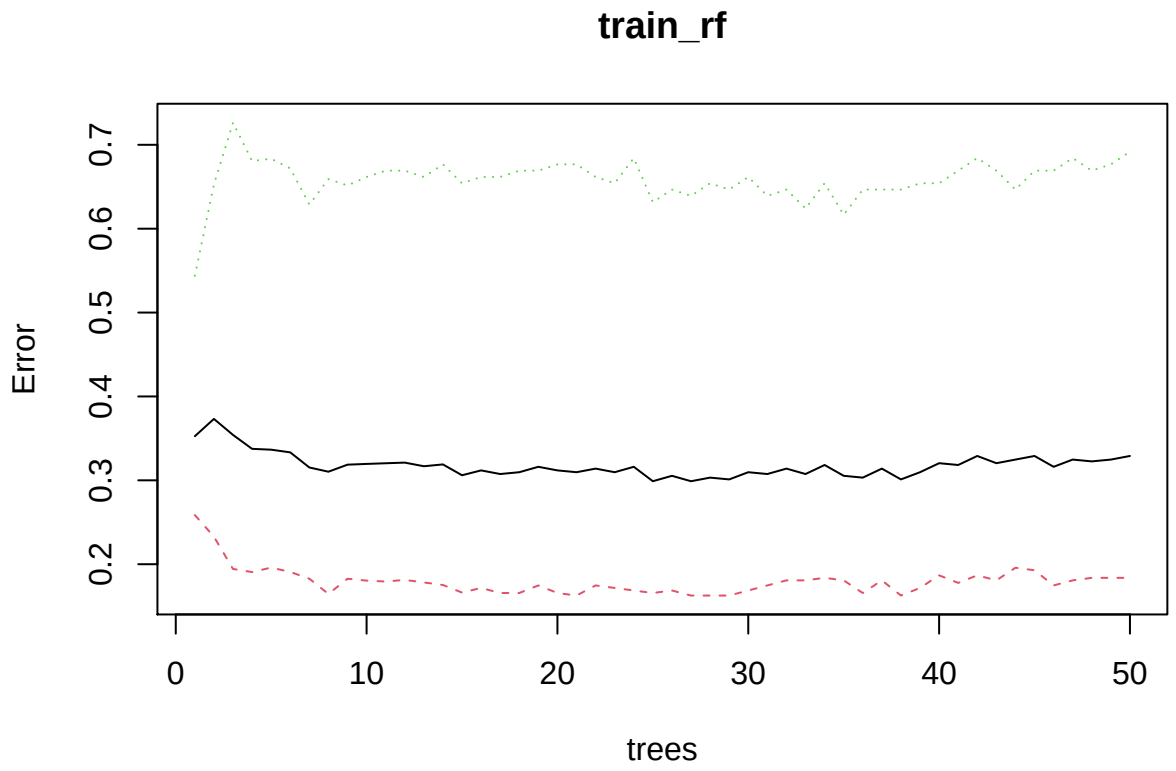
```
## [1] 0.7627119
```

```
accuracy_results <- bind_rows(accuracy_results, tibble(Method = "Random Forest", Accuracy = accuracy4))
accuracy_results %>% knitr::kable()
```

Method	Accuracy
Guessing	0.5128205
Albumin Cutoff	0.7118353
LDA	0.7203390
Random Forest	0.7627119

Plot the model

```
plot(train_rf)
```



1. The first model utilizes the LDA (Linear Discriminant Analysis) method. In this model, all of the variables are used as predictors of the Dataset outcome. An accuracy of 0.720339 is obtained. This is a great improvement over the guessing technique and a small improvement over utilizing a cut-off point for the single variable of Albumin. The LDA method assumes that the correlations between all of the variables are the same. In this method, mathematically the boundary is a line. This is a limitation of the technique which is that it does not capture non-linearity. We do not achieve a very high level of accuracy with this model.
2. The Random Forest model improves upon the LDA model by utilizing 6 predictors. An accuracy of 0.7627119 is obtained with 50 trees. Random forests improve predictability and reduce instability by averaging multiple decision trees. Use of the bootstrap creates randomness among many trees, and the aggregation of the trees is the forest. We use variable importance with the Random Forest model to see how often a predictor is used in the individual trees. The plot of the Random Forest model shows that the accuracy stabilizes around 10 trees.

Conclusion

The Indian Liver Dataset contains a lot of useful information that may potentially aid doctors in identifying patients with liver disease which could lead to earlier intervention and prevent the disease from progressing in patients who have it. Although the models demonstrated some significant predictive power of the variables in predicting liver disease, the accuracy obtained was not very high for either model which indicates some significant limitation of the data and techniques employed.

First of all, the modeling techniques have some drawbacks. The LDA method assumes that all correlations between variables are the same and does not take into account the potential nonlinearity of the data. We lose some flexibility with this modeling approach. A potential drawback of the Random Forest technique is that we lose interpretability which can be aided by using variable importance. Other modeling techniques such as k-nearest neighbors could be used to try to improve upon the accuracy. This was attempted during the project but did not yield a better result.

Another major limitation is that abnormal lab results could mean that a person potentially has other illnesses. The results don't solely point to liver disease. This could be a reason why we observe many patients in the Non-Liver group having abnormal levels of some substances in their blood. They may have another condition that is causing this result. Also surprising is the high prevalence of liver patients in this dataset (more than 70%) given the prevalence of the illness in the population (approximately 20%). This led me to assume that patients who go to this clinic are already suspected of having liver disease, although this information was not provided with the data.

The research indicates that abusing alcohol, having diabetes, or Hepatitis B or C are likely precursors to having liver disease. It seems that this data alone is insufficient. It would be important to get more information on the patients' medical and social histories. Having this complete picture of the patient would be most helpful in early identification of liver disease. It would also be interesting to see the data from a different population such as the U.S. or another Western country to compare the results. This problem merits more research to improve detection of this illness, which should incorporate other types of data, such as alcohol consumption and the presence of other health conditions. Adding this information could improve potential future models.