diagnosis

Medea

5/12/2021

Contents

1 Introduction

The topic of this study is trying to predict fertility : normal or altered. We have used data from https: //archive.ics.uci.edu/ml/datasets/Fertility, we downloaded data, added header & save as csv. The modified version now located on the github. This is location from where will access it through project.

```
if(!require(tidyverse)) install.packages("tidyverse",
                                         repos = "http://cran.us.r-project.org")
if(!require(caret)) install.packages("caret",
                                     repos = "http://cran.us.r-project.org")
if(!require(data.table)) install.packages("data.table",
                                          repos = "http://cran.us.r-project.org")
if(!require(rvest)) install.packages("rvest",
                                     repos = "http://cran.us.r-project.org")
if(!require(rpart)) install.packages("rpart",
                                     repos = "http://cran.us.r-project.org")
if(!require(rpart.plot)) install.packages("rpart.plot",
                                          repos = "http://cran.us.r-project.org")


library(caret)
library(data.table)
library(tidyverse)
library(rvest)
library(rpart.plot)

url_csv <- paste0("https://raw.githubusercontent.com/medeag/fertility-capstone/",
                  "main/dataset/fertility_diagnosis.csv")
diagnosis <- read.csv(url_csv, header = TRUE)
```

We have comfired how many samples do we have in datasheet and what are column names.

```
dim(diagnosis)
```

```
## [1] 100  10
```

```
names(diagnosis)
```

```
##  [1] "Season"              "Age"                 "Childish_Diseases"
##  [4] "Trauma"              "Surgeon"             "Fevers_Last_Year"
##  [7] "Alcohol_Consumption" "Smoking"             "Sitting_Per_Day"
## [10] "Output"
```

So we have 100 samples and 10 column. Here we are copying definition of columns from the archive.ics.uci.edu

1. Season : in which the analysis was performed. 1) winter, 2) spring, 3) Summer, 4) fall. (-1, -0.33, 0.33, 1)
2. Age : at the time of analysis. 18-36 (0, 1)
3. Diseases: Childish diseases (ie , chicken pox, measles, mumps, polio) 1) yes, 2) no. (0, 1)
4. Trauma: Accident or serious trauma 1) yes, 2) no. (0, 1)
5. Surgeon: Surgical intervention 1) yes, 2) no. (0, 1)
6. Fevers_Last_Year: High fevers in the last year 1) less than three months ago, 2) more than three months ago, 3) no. (-1, 0, 1)
7. Alcohol_Consumption: Frequency of alcohol consumption 1) several times a day, 2) every day, 3) several times a week, 4) once a week, 5) hardly ever or never (0, 1)
8. Smoking: Smoking habit 1) never, 2) occasional 3) daily. (-1, 0, 1)
9. Sitting_Per_Day: Number of hours spent sitting per day ene-16 (0, 1)
10. Output: Diagnosis normal (N), altered (O)

Before starting any analysis let's review how many altered & normal cases we have.

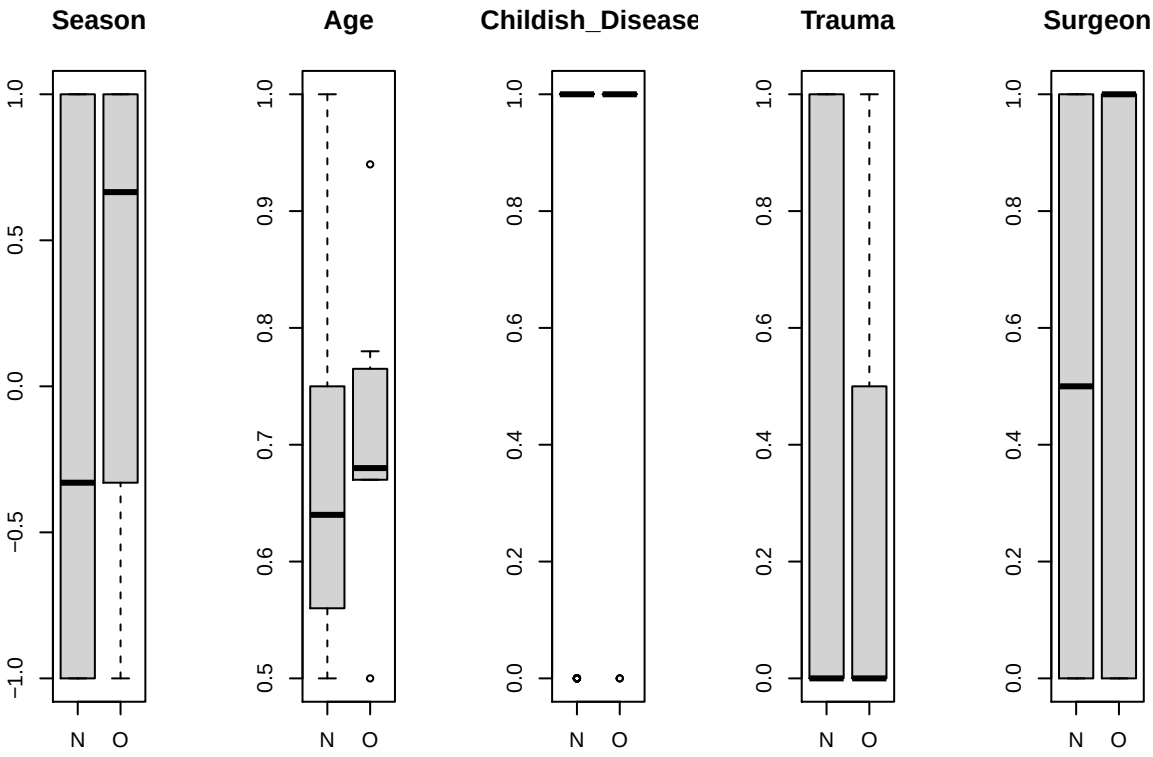
```
table(diagnosis$Output)
```

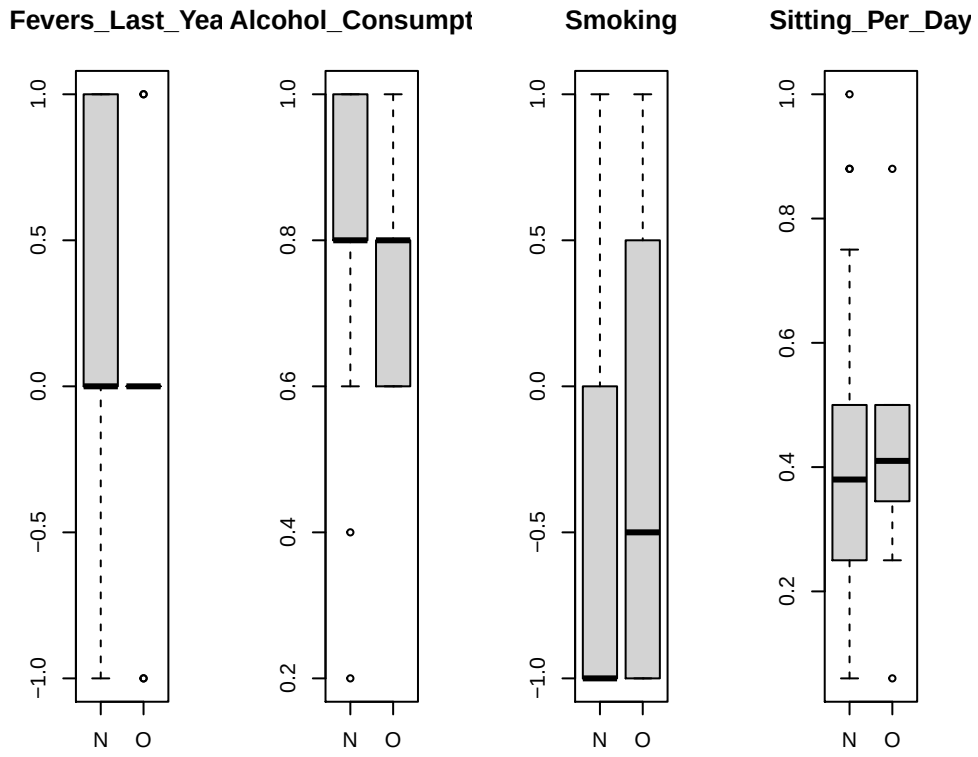
```
##
##  N  O
## 88 12
```

so we have 88 normal & 12 altered cases.

We use boxplot to see how each column related to normal vs altered case

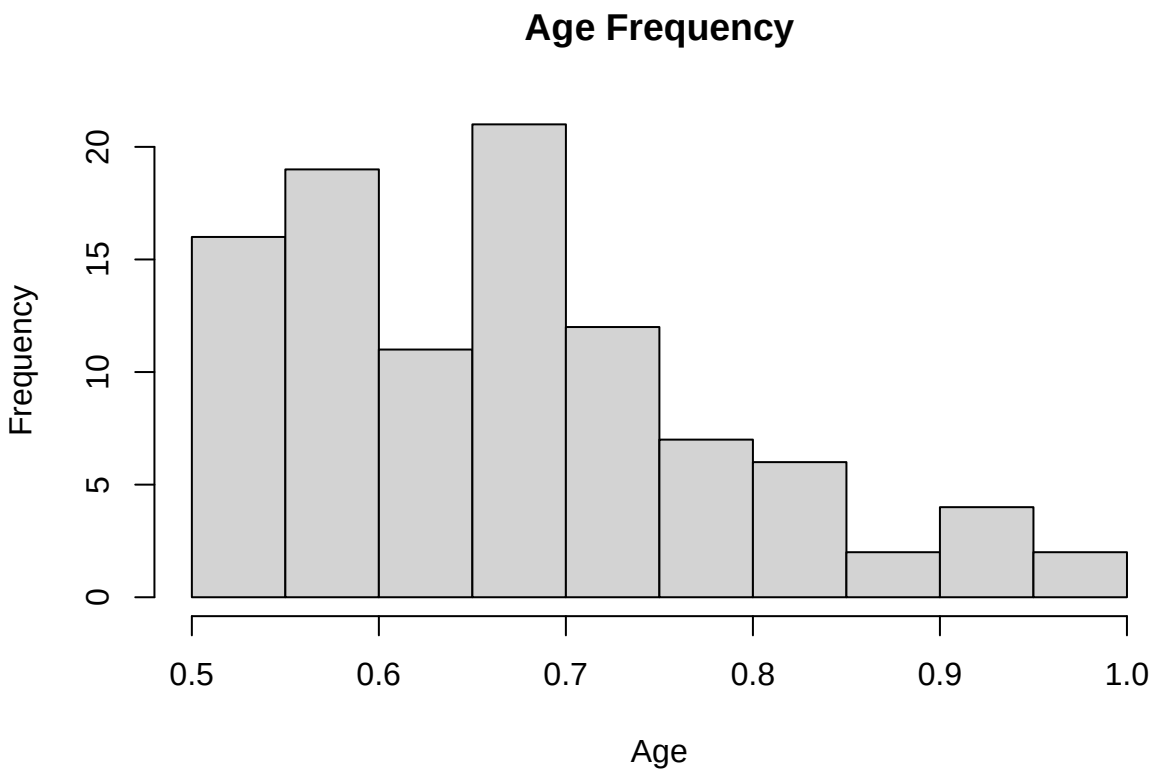
```
# boxplot for each column
par(mfrow=c(1,5))
for(i in 1:9){
    boxplot(split(diagnosis[,i],diagnosis$Output), main=names(diagnosis)[i])
}
```


Season
Age
Childish_Disease
Trauma
Surgeon
N
O

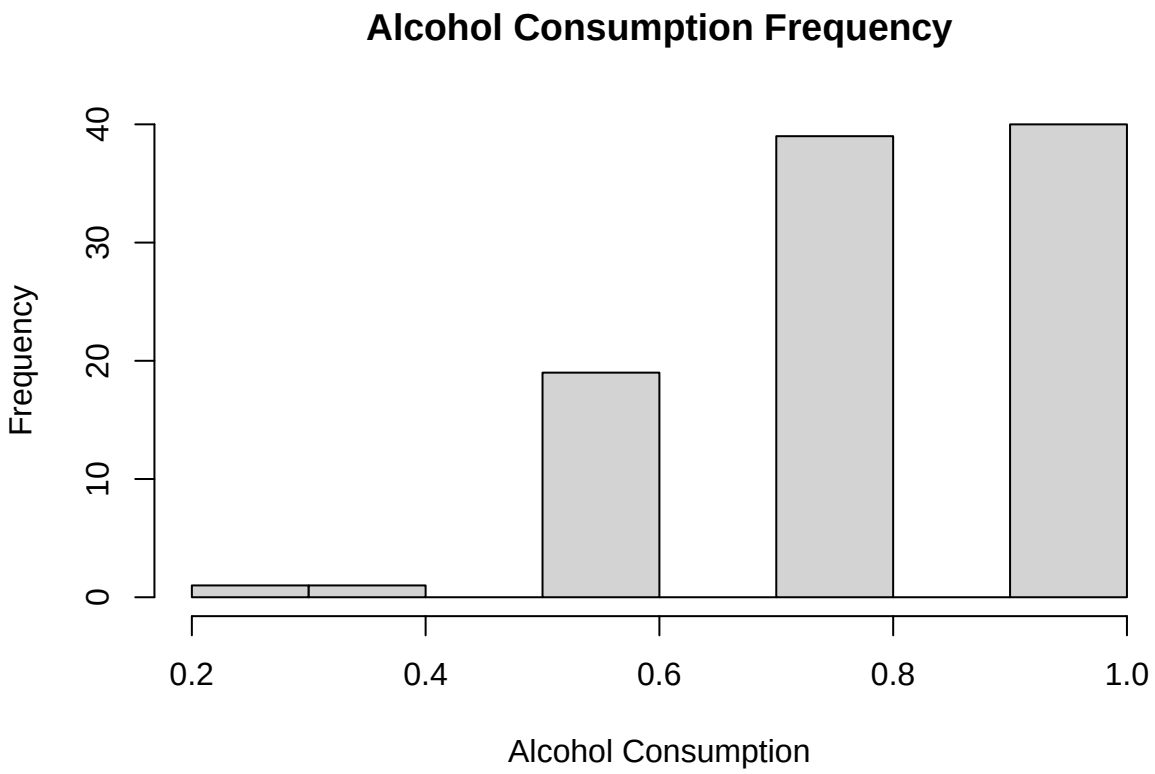


Interesting take away from this graphics that altered cases looks like more in upper age limit than normal ones, also alcohol parameter is distinguishes records. Also alcohol consumption is different between between normal and altred

```
hist(diagnosis$Age, main= "Age Frequency" , xlab = "Age")
```




```
hist(diagnosis$Alcohol_Consumption, main ="Alcohol Consumption Frequency",
     xlab = "Alcohol Consumption")
```



2 Analysis

Our goal is to see if we can predict results using different kind of methods. For simplicity sake we will use following three method:

1. Linear Discriminant Analysis (LDA)
2. K-NN
3. Decision Trees (Also we will draw decision tree here)
4. Random Forest (also we will list most important predictors)

We will use train method from caret package.

But first we need to divide dataset into the train & test sets, since dataset is rather small we will devide data set into two parts: 90% for training and 10% for test.

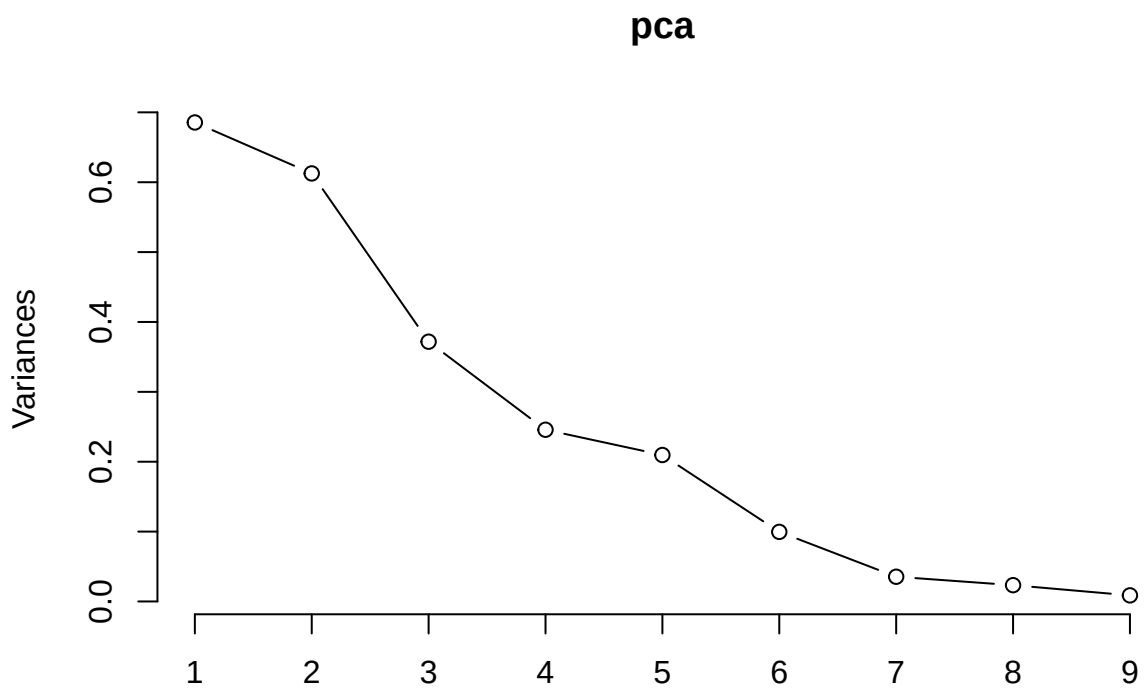
```
set.seed(1,sample.kind = "Rounding")
y<-diagnosis$Output
x <- diagnosis[-10]
test_index <- createDataPartition(y, times = 1, p = 0.1, list = FALSE)
test_set_x <- x[test_index, ]
test_set_y <- y[test_index]
train_set_x <- x[-test_index,]
train_set_y <- y[-test_index]
```

but before moving forward we need to answer question: we have 9 columns which affects output result, can be variance explained with fewer ones? Which one are the important ones? Let's perform principle analysis to check.

```
tmp <- train_set_x
pca <- prcomp(tmp)
summary(pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4     PC5     PC6     PC7
## Standard deviation     0.8279 0.7827 0.6097 0.4957 0.45785 0.31557 0.18817
## Proportion of Variance 0.2990 0.2672 0.1622 0.1072 0.09145 0.04344 0.01545
## Cumulative Proportion  0.2990 0.5663 0.7285 0.8357 0.92713 0.97057 0.98602
##                           PC8     PC9
## Standard deviation     0.15271 0.09343
## Proportion of Variance 0.01017 0.00381
## Cumulative Proportion  0.99619 1.00000
```

```
plot(pca, type = "l")
```



So 97% of the data we have can be described with 6 columns. That's doesn't sounds like much improvement.

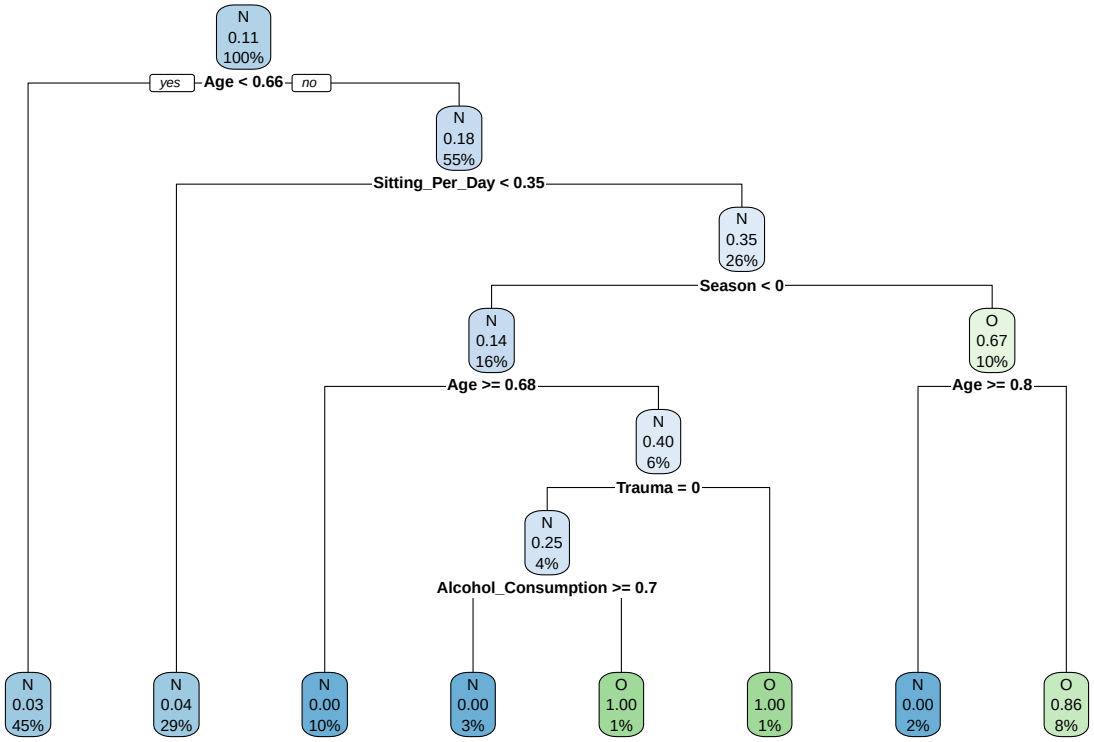
Now it is time to train our model:

```
# LDA
control <- trainControl(method="cv", number=5,  p = .9)

#set.seed(2007,sample.kind = "Rounding")
train_lda <- train(train_set_x,train_set_y,
                   method="lda",
                   trControl=control)

# k-Nearest Neighbors
#set.seed(2007,sample.kind = "Rounding")
train_knn <- train(train_set_x,train_set_y,
                   method="knn",
                   trControl=control)

#Decision Tree
train_rpart <- train(train_set_x,train_set_y,
                     method = "rpart",
                     tuneGrid = data.frame(cp = seq(0, 0.05, len = 25)),
                     control = rpart.control(minsplit = 1, minbucket = 1))
rpart.plot(train_rpart$finalModel)
```




```
#Random Forest
#set.seed(2007,sample.kind = "Rounding")
train_rf <- train(train_set_x,train_set_y,
                  method="rf",
```

```
                  trControl=control)

varImp(train_rf)
```

```
## rf variable importance
##
##                      Overall
## Sitting_Per_Day      100.000
## Age                   79.616
## Season                46.424
## Alcohol_Consumption   32.207
## Fevers_Last_Year      17.204
## Trauma                12.772
## Smoking                7.920
## Surgeon                5.139
## Childish_Diseases      0.000
```

Now it is time to do predictions. Please note we tried to tune decision tree and random forest, to create the best model.

```
# Evaluate LDA model on test data
predictions <- predict(train_lda, test_set_x)
results <-tibble(method='lda',
                 accuracy=(mean(predictions==test_set_y)))


# Evaluate KNN model on test data
predictions <- predict(train_knn, test_set_x)
results <- results %>% add_row(method = "knn",
                               accuracy=(mean(predictions==test_set_y)))


# Evaluate Decision Tree on Test data
predictions <- predict(train_rf, test_set_x)
results <- results %>% add_row(method = "Decision Tree",
                               accuracy=(mean(predictions==test_set_y)))

# Evaluate Random Forest model on test data
predictions <- predict(train_rf, test_set_x)
results <- results %>% add_row(method = "Random Forest",
                               accuracy=(mean(predictions==test_set_y)))
```

3 Results

Below are listed the actual results we got

```
results %>% knitr::kable()
```

method	accuracy
lda	0.8181818
knn	0.8181818
Decision Tree	0.8181818
Random Forest	0.8181818

The results are surprising since all 4 methods come with the same accuracy . So we have no clear winner here.

4 Conclusions

Although we can predict fertility base on sample dataset, we think a big limitation of the study is small sample size, it will be nice if in the future we will be able to find larger sample and with additional features, like weight and genetics. We can also try to combine/assemble several methods to see if we get better result.