Course9-part2-whu

Wanjun Hu

2025-01-25

Section 1. Introduction

This project studies a very small dataset, called diabetes.csv. The dataset is provided at the Kaggle website. Here is the link https://www.kaggle.com/datasets/akshaydattatraykhare/diabetes-dataset. According to the webpage at kaggle, this daatset is obtained from the national Institute of Diabetes and Digestive and Kindney Diseases.

It is a collection of 768 records of women ranging from 21 to 80 years old. All are Pima Indian heritage. The datset contains 9 variables, pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, Age and Outcome. All variables are of integer or numeric types. The outcome is either 0 or 1, while O means not having diabetes, and 1 having diabetes.

Subsection 1.1. Downloading the dataset

Downloading directly from the kaggle website is tricky. It requres login. Instead, we downloaded the csv file and upload to my own github repo https://www.github.com/whu-asurams/diabetes. One can download the whole repo as a zip file. The link is https://github.com/whu-asurams/diabetes/archive/refs/heads/main.zip.

After downloading the zip file, we can extract the diabetes.csv file, which is inside the folder “diabetes-main/”.

The following code will handle the the downloading and unzipping task.

```r
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")

library(tidyverse)
library(caret)
library(dplyr)

options(timeout = 120)

diabetes_file<- "diabetes.zip"

path<- "https://github.com/whu-asurams/diabetes/archive/refs/heads/main.zip"
if(!file.exists(diabetes_file))
  download.file(path, diabetes_file)

csv_file <- "diabetes-main/diabetes.csv"
if(!file.exists(csv_file))
  unzip(diabetes_file, csv_file)
```

Subsection 1.2. Basic information of the dataset

After loading the “daibetes.csv” file into R. One can check quickly the information of the dataset.

The variables are

Pregnancies - number of times a woman has been pregnant before

Glucode - The glucose reading of a blood test

BloodPressure - the blood pressur

SkinThickness - The skid thickness

Insulin - The insulin reading of blood work

BMI - BMI number

DiabetesPedigreeFunction - A number calculated based on the heritage. We rename it to DPF for formatting purpose.

Age - Patentien’s age when she was recorded.

Outcome - whether or not a patient has diabetes.

The structure of the dataset is provided below

```
## 'data.frame':    768 obs. of  9 variables:
##  $ Pregnancies  : int  6 1 8 1 0 5 3 10 2 8 ...
##  $ Glucose      : int  148 85 183 89 137 116 78 115 197 125 ...
##  $ BloodPressure: int  72 66 64 66 40 74 50 0 70 96 ...
##  $ SkinThickness: int  35 29 0 23 35 0 32 0 45 0 ...
##  $ Insulin      : int  0 0 0 94 168 0 88 0 543 0 ...
##  $ BMI          : num  33.6 26.6 23.3 28.1 43.1 25.6 31 35.3 30.5 0 ...
##  $ DPF          : num  0.627 0.351 0.672 0.167 2.288 ...
##  $ Age          : int  50 31 32 21 33 30 26 29 53 54 ...
##  $ Outcome      : int  1 0 1 0 1 0 1 0 1 1 ...
```

The data range of each variable is listed below.

```
## [1] "Pregnancies              : integer, range = (0, 17)"
## [2] "Glucose                  : integer, range = (0, 199)"
## [3] "BloodPressure            : integer, range = (0, 122)"
## [4] "SkinThickness            : integer, range = (0, 99)"
## [5] "Insulin                  : integer, range = (0, 846)"
## [6] "BMI                      : numeric, range = (0.000, 67.100)"
## [7] "DPF                      : numeric, range = (0.078, 2.420)"
## [8] "Age                      : integer, range = (21, 81)"
```

Subsection 1.3. Split the dataset to training and test parts

We split the dataset into a training part and a test part. The training part is 80% of the dataset, while the test is 20%.

```
## [1] "The summary of train_part"
## 'data.frame':    613 obs. of  9 variables:
##  $ Pregnancies  : int  6 1 8 1 5 3 10 2 8 4 ...
##  $ Glucose      : int  148 85 183 89 116 78 115 197 125 110 ...
##  $ BloodPressure: int  72 66 64 66 74 50 0 70 96 92 ...
##  $ SkinThickness: int  35 29 0 23 0 32 0 45 0 0 ...
##  $ Insulin      : int  0 0 0 94 0 88 0 543 0 0 ...
##  $ BMI          : num  33.6 26.6 23.3 28.1 25.6 31 35.3 30.5 0 37.6 ...
##  $ DPF          : num  0.627 0.351 0.672 0.167 0.201 0.248 0.134 0.158 0.232 0.191 ...
##  $ Age          : int  50 31 32 21 30 26 29 53 54 30 ...
##  $ Outcome      : int  1 0 1 0 0 1 0 1 1 0 ...

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

## [1] "The summary of test_part"

## 'data.frame':    155 obs. of  9 variables:
##  $ Pregnancies  : int  0 5 7 3 3 9 7 9 8 5 ...
##  $ Glucose      : int  137 166 196 158 88 102 133 171 133 44 ...
##  $ BloodPressure: int  40 72 90 76 58 76 84 110 72 62 ...
##  $ SkinThickness: int  35 19 0 36 11 37 0 24 0 0 ...
##  $ Insulin      : int  168 175 0 245 54 0 0 240 0 0 ...
##  $ BMI          : num  43.1 25.8 39.8 31.6 24.8 32.9 40.2 45.4 32.9 25 ...
##  $ DPF          : num  2.288 0.587 0.451 0.851 0.267 ...
##  $ Age          : int  33 51 41 28 22 46 37 54 39 36 ...
##  $ Outcome      : int  1 1 1 1 0 1 0 1 1 0 ...
```

In the following, we will analyze the dataset and select three combinations of predictors. Then, we will use each combination of predictors to feed a sequence of models. An ensemble analysis is also provided at the end.

Section 2. Methods/Analysis

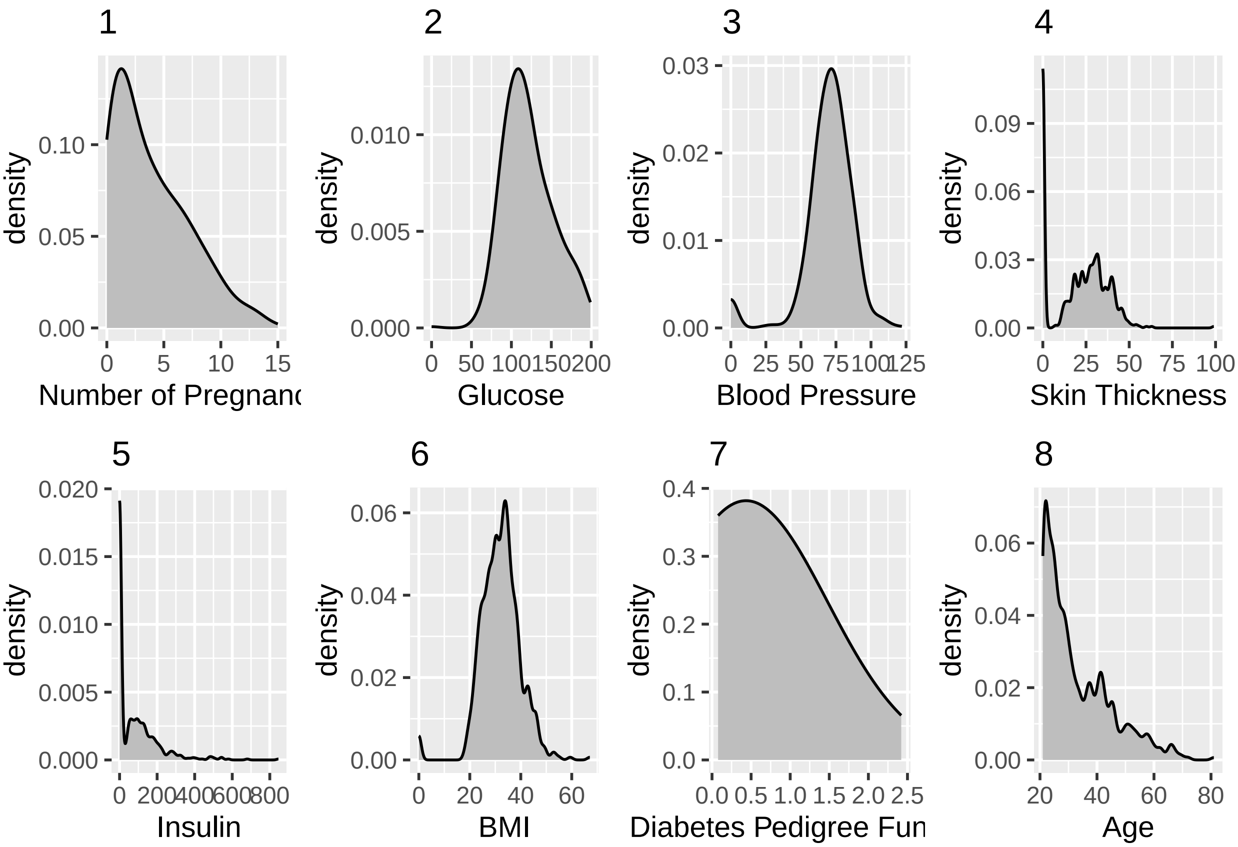
a methods/analysis section that explains the process and techniques used, including data cleaning, data exploration and visualization, insights gained, and your modeling approaches (you must use at least two different models or algorithms);

The dataset is clean and well-prepared. No further cleaning is necessary.

Each of the variables is independent. Some variables such as blood pressure demonstarte a less relevancy to the outcome. We will then to find some combinations of the 8 variables, that provides the highest prediction accuracy. We train 7 models on each of the combinations.

Subsection 2.1. Inspect each predicator (variable)

The density plot of each variable is provided below. One can quickly observe that the variables Glucose, BloodPressure and BMI are closed to normal distribution. Others are skewed to the right, or positively skewed.



Subsection 2.2. Correlations

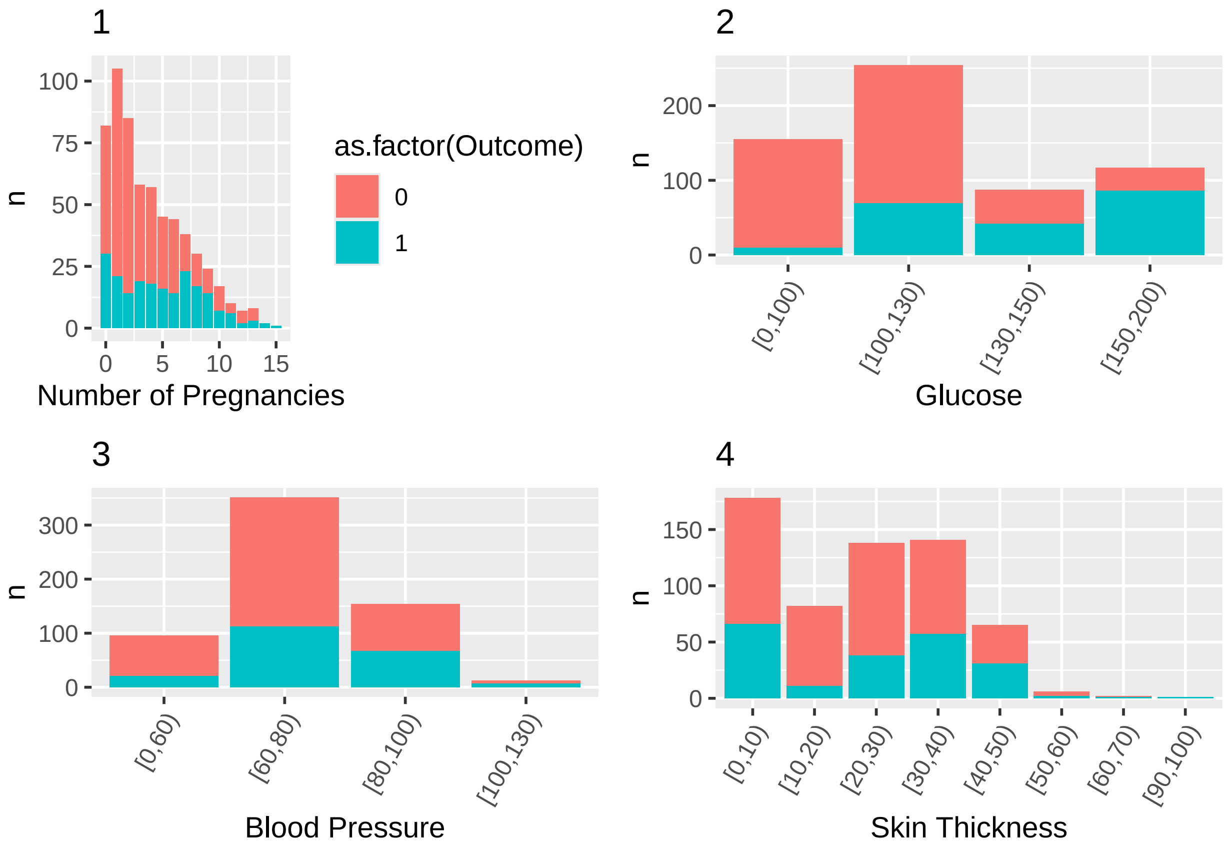
Running correlation test on each variblae of “Pregnancies”, “Glucose”, “BloodPressure”, “SkinThickness”, “Insulin”, “BMI”, “DPF”, “Age”, against the Outcome varibale, we obtain the following summary.

```
## [1] "Pregnancies    : t=5.14, df=611, p-value=3.6e-07 <= 0.05, cor=0.2037"
## [2] "Glucose        : t=14.44, df=611, p-value=7.2e-41 <= 0.05, cor=0.5044"
## [3] "BloodPressure  : t=2.57, df=611, p-value=1.0e-02 <= 0.05, cor=0.1033"
## [4] "SkinThickness  : t=2.09, df=611, p-value=3.7e-02 <= 0.05, cor=0.0841"
## [5] "Insulin        : t=3.89, df=611, p-value=1.1e-04 <= 0.05, cor=0.1554"
## [6] "BMI            : t=7.95, df=611, p-value=9.3e-15 <= 0.05, cor=0.3061"
## [7] "DPF            : t=5.01, df=611, p-value=7.1e-07 <= 0.05, cor=0.1986"
## [8] "Age            : t=5.97, df=611, p-value=4.0e-09 <= 0.05, cor=0.2347"
```

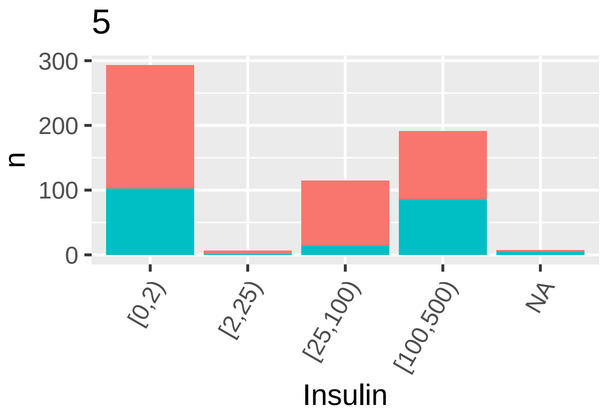
All variables have positive correlation with the outcome. The p-values are also less than 0.05, which means the positive correlations are statistically significant. In particular, the variables Glucose, BMI„ Pregnancies, Age demonstrate a correlation of more than 0.2, while the SkinThickness has a much smaller correlation.

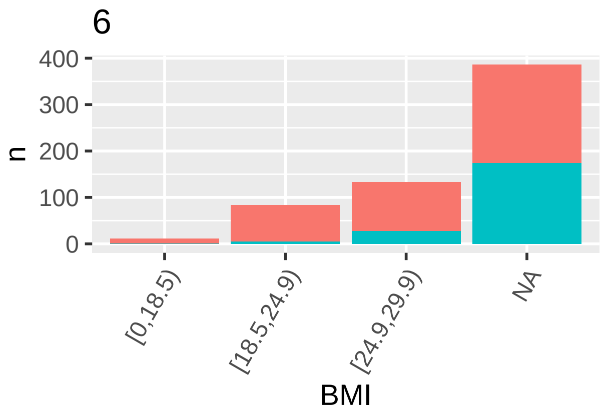
Subsection 2.3. Bar charts against Outcome

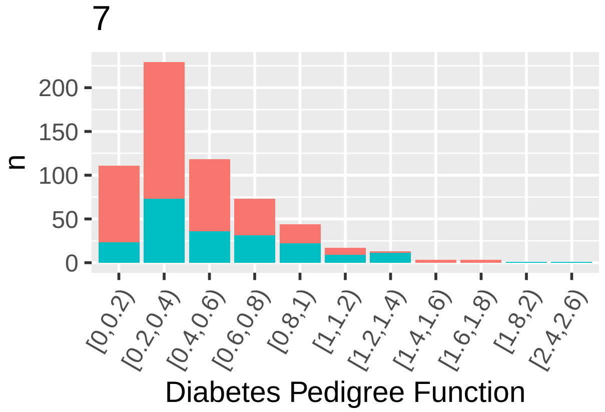
The bar chart of each variable against outcome is also provided. One can quickly observe that the SkinThickness shows no patterns. The BloodPressure and Insulin shows a little bit pattern.

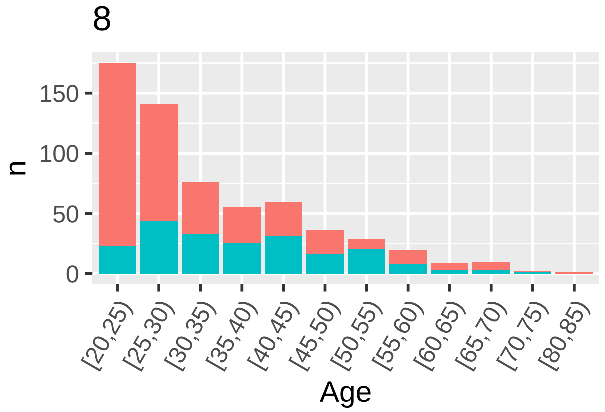


See the other four









Subsection 2.4. Select predictors

Based on above analysis, we will select the following combinations of predictors

1. Combination 1: all 8 varibles
2. Combination 2: Pregnancies, Glucose, Insulin, BMI, DPF, and Age (all have correlations>0.15)
3. Combination 3: Pregnancies, Glucose, BMI, and Age (all have correlations> 0.2)

Section 3. Results

We will prepare three dataframe based on above analysis, and 7 models to train.

Subsection 3.1. Combinations of predictors

According to above analysis, we choose the three combinations. They are selected using the following code.

```
# For train_part

x1<- train_part[,-9]
x2<- train_part%>%
  select(Pregnancies, Glucose, Insulin, BMI, DPF, Age)
```

```
x3<- train_part%>%
  select(Pregnancies, Glucose, BMI, Age)

y<-factor(train_part$Outcome)

#For test_part
xt1<- test_part[,-9]
xt2<- test_part%>%
  select(Pregnancies, Glucose, Insulin, BMI, DPF, Age)
xt3<- test_part%>%
  select(Pregnancies, Glucose, BMI, Age)

yt<- factor(test_part$Outcome)
```

The models we will use are "glm", "lda", "naive_bayes", "knn", "gamLoess", "qda", "rf".

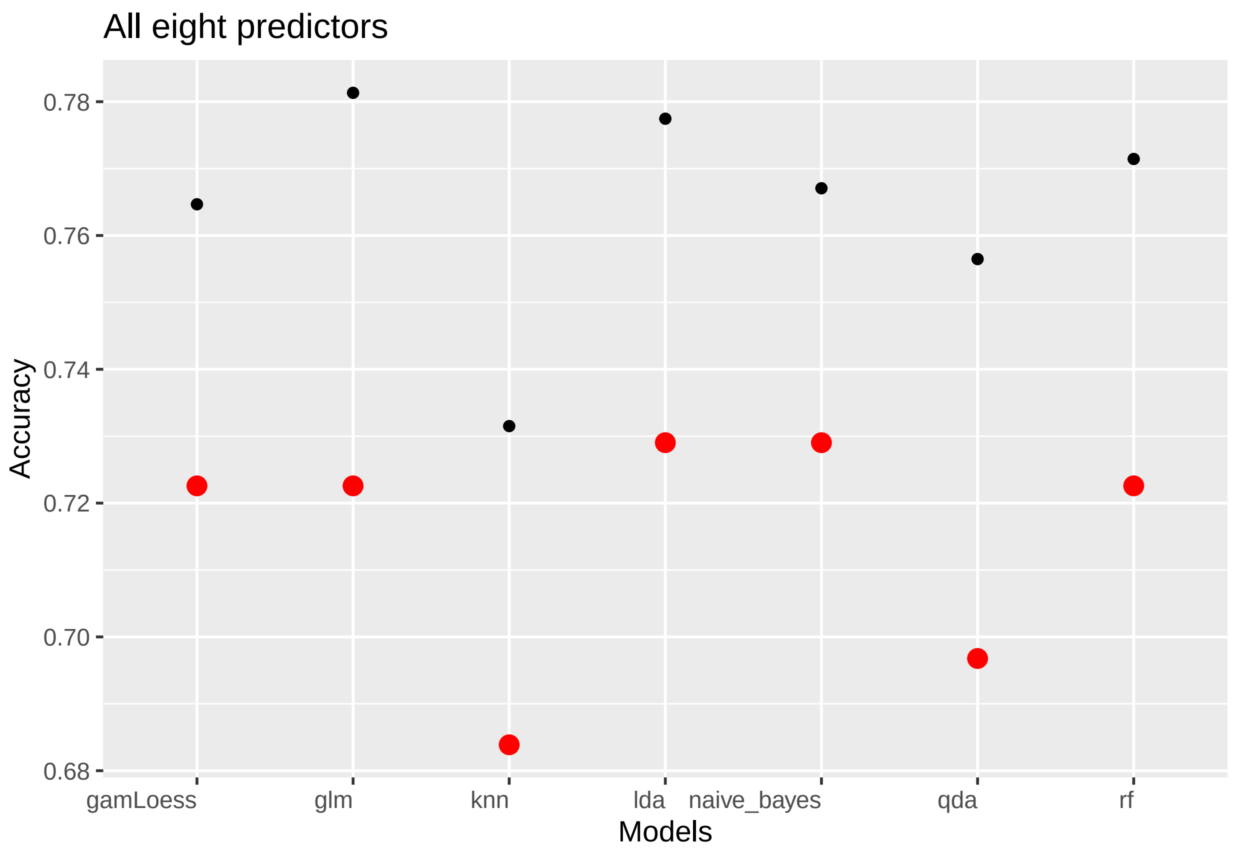
Subsection 3.2. Train and predict

Now, we will train and predict on each combination.

Combination 1: on all eight variables

The training accuracy and test predictions are plotted below. The models of "lda" and "naive bayes" provide the highest test prediction accuracy, which are around 0.73, while the knn model provides the lowest test prediction accuracy of around 0.684.

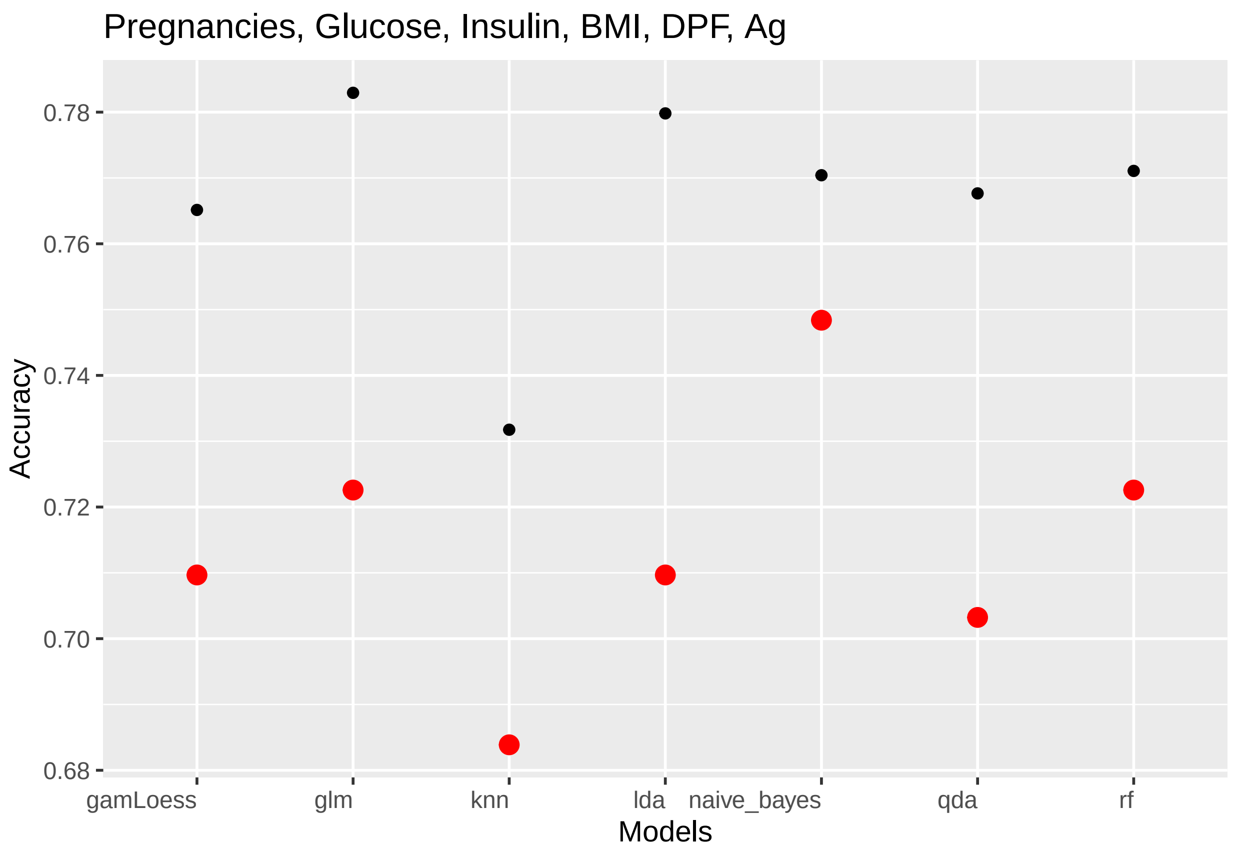
```
train_pred(x1,xt1, y, yt, "All eight predictors")
```



Combination 2: On Pregnancies, Glucose, Insulin, BMI, DPF, and Age.

Seven models are trained using the second set of data. The model "naive nayes" provides the highest prediction accuracy, which is around 0.75. The knn model provides the lowest prediction accuracy, which is 0.682

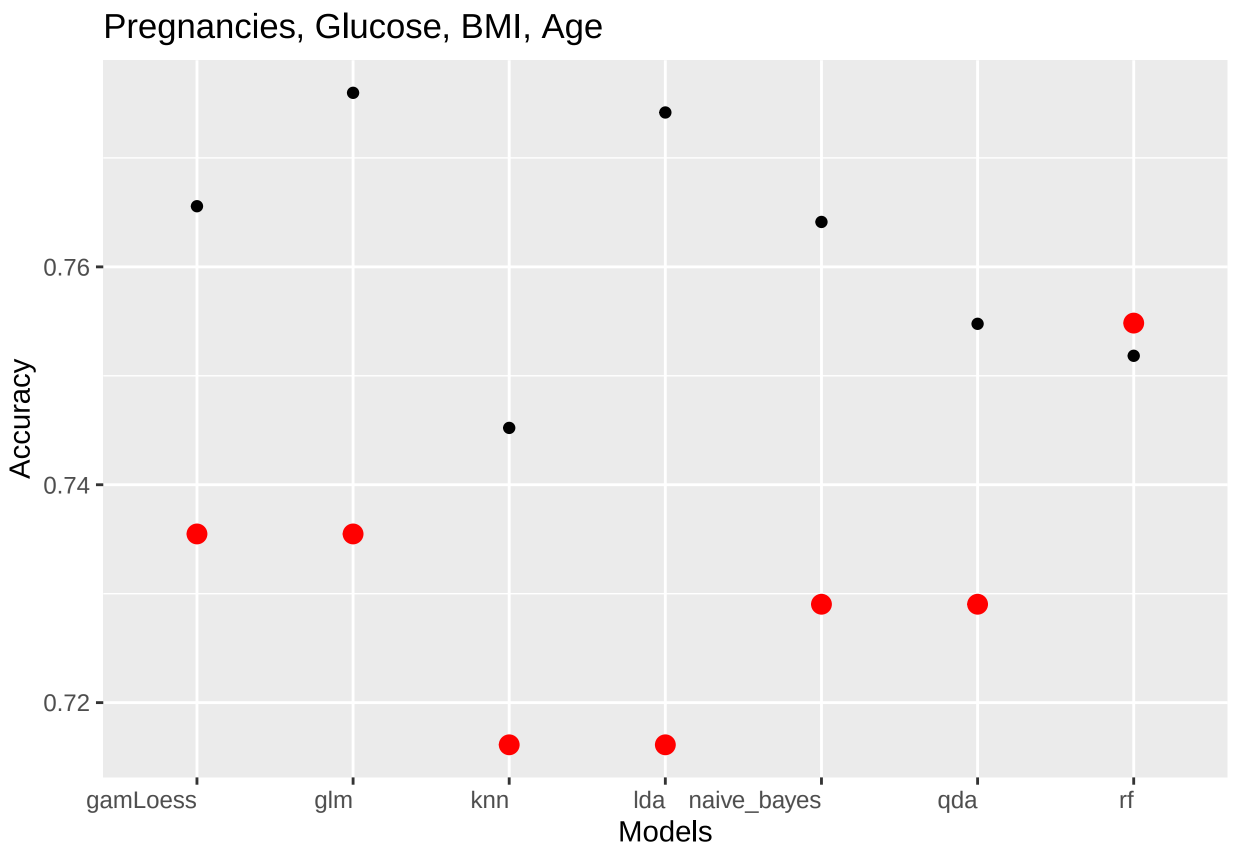
```
train_pred(x2, xt2, y, yt, "Pregnancies, Glucose, Insulin, BMI, DPF, Ag")
```



Combination 3: OnPregnancies, Glucose, BMI, and Age

The third set of data is used for training in each of the seven models. The rf model provides the highest prediction accuray on the test dataset, which is around 0.755, while knn and lda provide the lowest at 0.715.

```r
train_pred(x3, xt3, y, yt, "Pregnancies, Glucose, BMI, Age")
```



In above charts, the black dots are the training accuracies using each model, while the big yellow dots are the average of the test predictions against the outcome.

In general, all models except "rf" are over trained. The over taining is likely due to the the sample size, especially the sample size of test dataset, which is only 155 records, comparing to an usually size of 20,000.

The KNN model shows a lower accuracy, while the "glm" and "lda" models show a high accuracy.

Subsection 3.3. Ensemble of models

We used 7 models in the training and predictions. Each model has a test prediction accuracy around 0.72. We can combine them togther and build a voting model, which uses majority votes, i,e., if 50% of the models say a patient has diabetes, then the new ensemble model predict that she has diabetes.

We use the following function to build a dataframe that list the prediction of each model as a column. Then, we add a new column, called "votes" and another called "y_hat".

```r
# Ensemble model by majority vote
model_ensemble<- function(tp){
  dp<- as.data.frame(tp)
  names(dp)<- models
  dp<- dp%>%
    mutate(votes = rowMeans(dp==1),
          y_hat = ifelse(votes>=0.5, 1, 0))

  mean(dp$y_hat==yt)
}
```

For combination 1: Using all eight predictors

The test prediction accuracy is calculated below.

```r
# prediction of combination 1 using the ensemble model
model_ensemble(test_prediction_1)
```

```
## [1] 0.7225806
```

For combination 2: Using variables Pregnancies, Glucose, Insulin, BMI, DPF, Age

The test prediction accuracy is calculated below.

```r
# prediction of combination 2 using the ensemble model
model_ensemble(test_prediction_2)
```

```
## [1] 0.716129
```

For combination 3: Using variables Pregnancies, Glucose, BMI, Age

The test prediction accuracy is calculated below.

```r
# prediction of combination 3 using the ensemble model
model_ensemble(test_prediction_3)
```

```
## [1] 0.7225806
```

Section 4. Conclusion

To summarize, this project studied the predictors that be effectively used to predict whether a woman has diabete based on some or all of the variables Pregnancies, Glucose, SkinThcikness, BloodPressure, Insulin, BMI, DPF, Age

We choose three different combinations

1. All right variables
2. Pregnancies, Glucose, Insulin, BMI, DPF, Age
3. Pregnancies, Glucose, BMI, Age

A total of 7 models are used in training. An emsemble of the seven is also built based on the majority voting mechanism.

The major limitation of this project is the size of the dataset. Ideally, it is better to have 20,000 records. Ours only has 768 records. However, the method used in this project can be used in future dataset.

In the future, we wish to explore large dataset using the same method developped in this project.

Section 5. Reference

1. https://www.kaggle.com/datasets/akshaydattatraykhare/diabetes-dataset
2. https://docs.github.com/en/get-started/start-your-journey/downloading-files-from-github
3. https://github.com/whu-asurams/diabetes/archive/refs/heads/main.zip