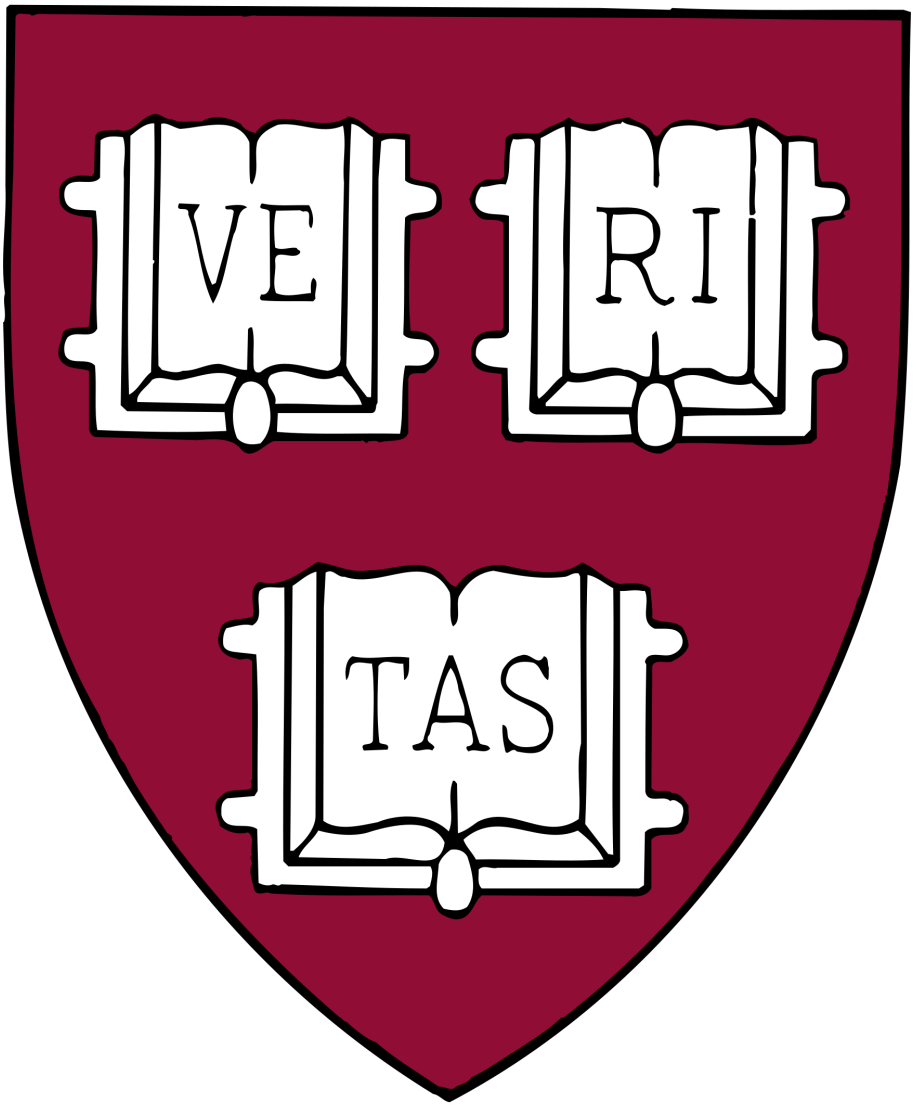


HarvardX - PH125.9x - Data Science

Movie Lens Project

Paul Nardon

2022-11-21

Contents

Abstract

This project aims to build the best predictive model to evaluate the risk that a patient has diabetes.

We used plethora of models in order to obtain the best efficiency.

In order to train and evaluate our different predictive models, we have work on a data set from the National Institute of Diabetes and Digestive and Kidney Diseases. It should be specified that all patients contained into this data set are females at least 21 years old of Pima indian heritage.

We build six models which are :
- Generalized Linear model;
- Generalized Additive Model using LOESS;
- kNN model;
- Classification tree model;
- Random Forest model;
- Random Forest model with Rborist method.

Our different models have got an accuracy between 0.77 and 0.82 while the F1_score is between 0.83 and 0.87. The efficiency of our predictive algorithm model are satisfactory.
However, these results these results should be qualified with regard to the few observations contained in the data set.
it should also be noted that the data set contained a lot of missing value.

Introduction

Diabetes is a disease that affects a large number of people around the world. Diabetes is a disorder of assimilation, use and storage of sugars provided by food. This results in a high blood glucose level There are two types of diabetes :
- type 1 diabetes : The body no longer recognizes these beta cells and destroys them (beta cells are destroyed by antibodies and immune cells, lymphocytes, made by the body). It is said to be a disease autoimmune. The glucose that cannot enter the cells returns to the blood. The blood glucose level then rises.

- type 2 diabetes : Overweight, obesity and lack of physical activity are the telltale cause of type 2 diabetes in genetically predisposed people.
Two abnormalities are responsible for hyperglycemia:
- either the pancreas still produces insulin but not enough, compared to blood sugar: this is insulinopenia;
- either this insulin acts badly, we then speak of insulin resistance.

The principles causes of diabetes are :
- A genetic origin;
- An unbalanced diet;
- Lack of physical activity; - Overweight.

The objective of this study is to predict whether a patient has diabetes or not.

This report is composed of 3 parts. In the first part, we will describe the data set and we will make some reprocessing to prepare data for our predictive machine learning algorithms.
In the second part, we will analyze the distribution and characteristics of each variable from data visualization and data analysis to understand the relations between them and build different algorithms models.
In the last part of this report, we will build different models based on the analysis of the database.

Part I - Data structure and data reprocessing

In this part one, we will start by loading the data set from which we will build our predictive algorithm models.

After loading the data set, we will describe his structure and reprocessing their data which we will used for data analyzing and visualization (cf. part II - Data analyzing and visualization) and for building our predictive algorithm models (cf. Part III - Predictive algorithm models).

A - Analyzing the structure of the data set

The data set is composed of 768 observations and 9 variables which are :

- "Pregnancies" which corresponds to the number of pregnancies by woman;
- "Glucose" which represents the quantity of glucose in blood;
- "BloodPressure" which corresponds to the Diastolic blood pressure (mm/Hg);
- "SkinThickness" which represents the triceps skin fold thickness (mm);
- "Insulin" which corresponds to the level of insulin in blood $\mu U/ml$;
- "BMI" which represents the body mass index (BMI);
- "DiabetesPedigreeFunction" which corresponds to a function which scores likelihood of diabetes based on family history;
- "Age" which represents the age of the patient;
- "Outcome" which correspond to the result of a diabetes test. A result of 0 means that the patient has not diabetes while a result of 1 means the patient has diabetes.

Table 1: An overview of the data set

Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
6	148	72	35	0	33.6	0.627	50	1
1	85	66	29	0	26.6	0.351	31	0
8	183	64	0	0	23.3	0.672	32	1
1	89	66	23	94	28.1	0.167	21	0
0	137	40	35	168	43.1	2.288	33	1
5	116	74	0	0	25.6	0.201	30	0

B - Data reprocessing

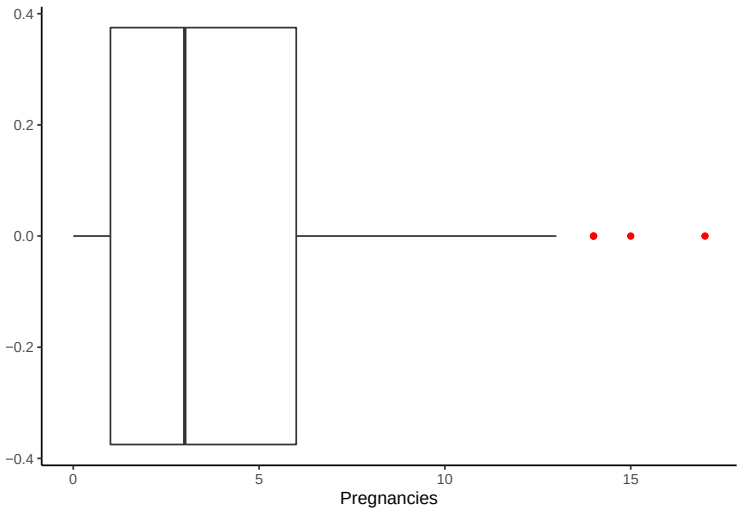
After having verify there wasn't NAS's values into the data set, we will reprocess data of each variable and create new data sets in order to can used them for, firstly, data analysis and visualization, and, on the other hand, training and evaluate our predictive algorithm models.

We will create box plot in order to find inconsistent values (i.e. outlier). We can make an assumption according to which the physiologically impossible values will be replace by NA value in the data set.

In order to save all the reprocessing that we are making, we will creat a new data frame whose name is "dataset_reprocessing".

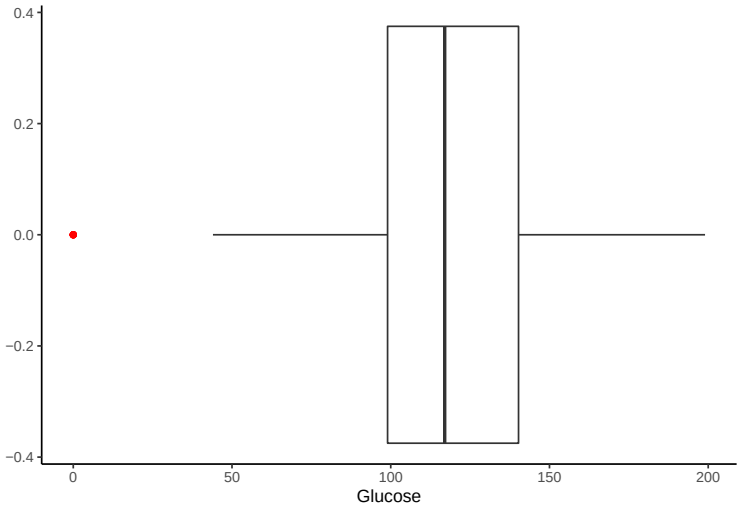
B.1 - Reprocessing of the Pregnancies variable

The box plot below shows us there is not value which is physiologically impossible.



B.2 - Reprocessing of the Glucose variable

We are looking for inconsistent data by making a box plot. We can see there is an outlier equal to 0.



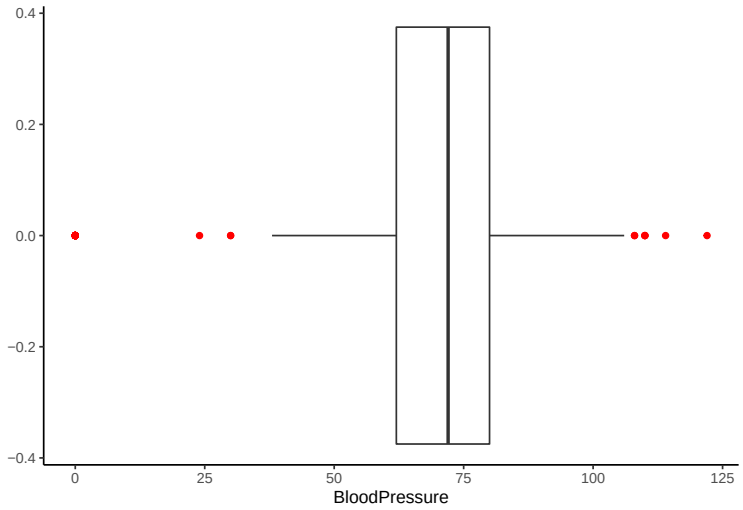
This last one is associated to 6 observations of the data set. A patient who would have a quantity of glucose equal to 0 would mean he would be die. We will give them the value of NA.

Table 2: An overview of the observations associated to the outlier

Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
1	0	48	20	0	24.7	0.140	22	0
1	0	74	20	23	27.7	0.299	21	0
1	0	68	35	0	32.0	0.389	22	0
5	0	80	32	0	41.0	0.346	37	1
6	0	68	41	0	39.0	0.727	41	1

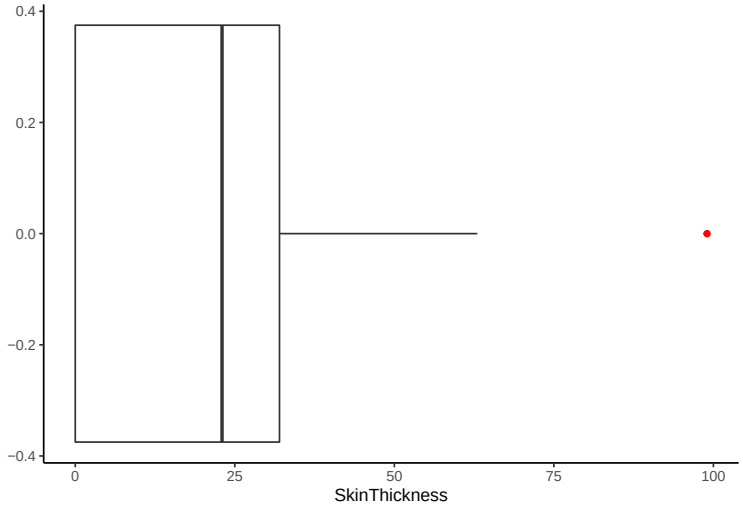
B.3 - Reprocessing of the Blood pressure variable

We do the same for the blood pressure variable. We will highlight the fact there are a lot of outliers for which the values are equal to 0, 24, 30, 108, 110, 114 and 122.
We will replace observations for which the distolic blood pressure is equal to 0 by NA in order to it is physiologically impossible. The patients having this value would be die.



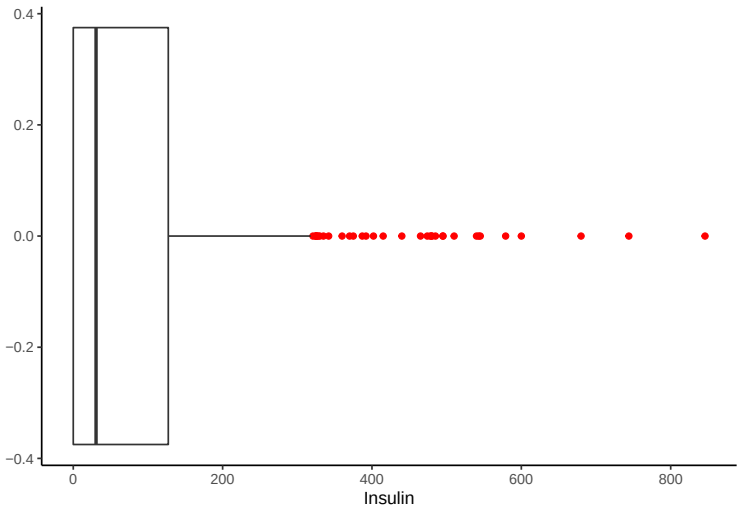
B.4 - Reprocessing of the triceps skin fold thickness variable

The boxplot below show us there is an outlier equal to 99 mm. We can also see there are plethora observation equal to 0 (i.e. 227 observations). We will replace them by NA.



B.5 - Reprocessing of the insulin variable

The analysis of the inconsistent data by making a box plot highligths the fact there are a lot of of outliers and the fact there are 374 observations for which the level of insulin in blood is equal to 0 $\mu U/ml$. We will replace them by NA in order to it is physiologically impossible. The patients having this value would be die.



B.6 - Reprocessing of the BMI variable

we will analyse the outliers by making a boxplot. We will see that there are 6 observations for whicth the BMI is equal to 0. We will replace them by NA.

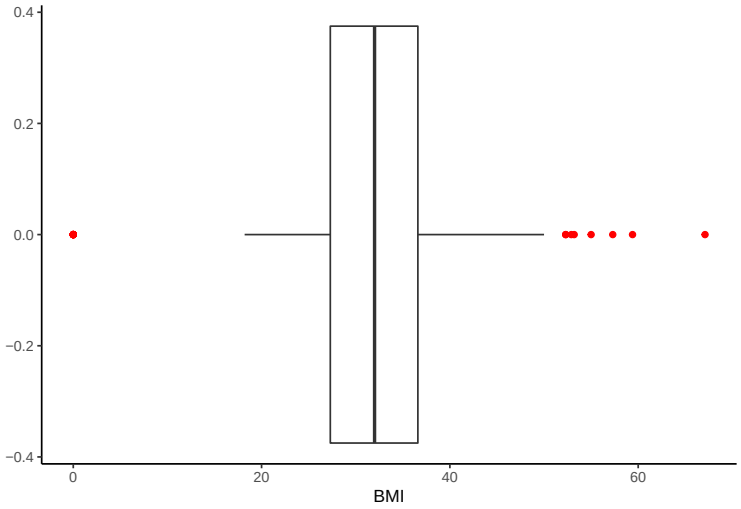
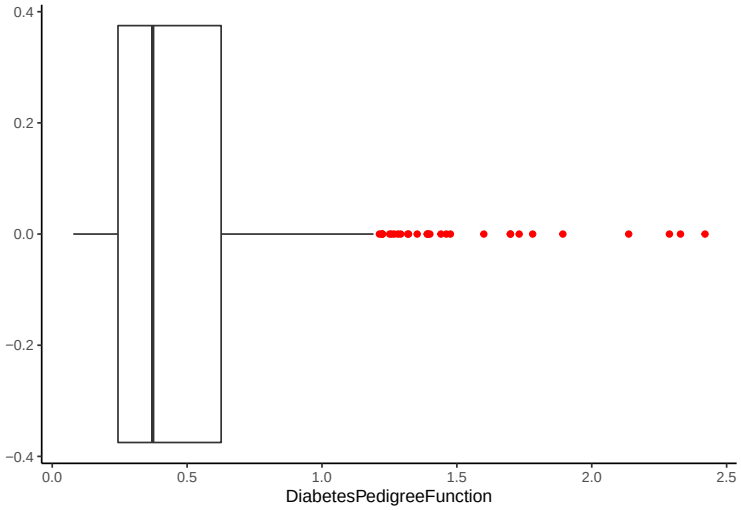


Table 3: An overview of the observations associated to the outlier equal to zero

Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
7	105	0	0	0	0	0.305	24	0
2	84	0	0	0	0	0.304	21	0
0	102	75	23	0	0	0.572	21	0
3	80	0	0	0	0	0.174	22	0
5	136	82	0	0	0	0.640	69	0
10	115	0	0	0	0	0.261	30	1

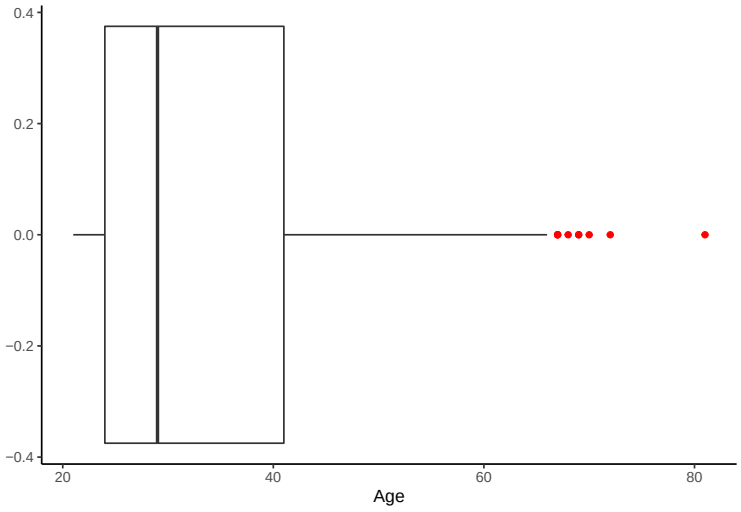
B.7 - Reprocessing of the diabetes pedigree function

The box plot below show us that there is not a value which is physiologically impossible.



B.8 - Reprocessing of the age variable

The box plot below shows us there is not value which is physiologically impossible.



B.9 Reprocessing of the outcome variable

We will transform data into factor value in order to use them for training and testing our predictive algorithm models.

Part II - Data analyzing and data visualization

First of all, we will analyse the distribution of each data set variables in order to understand their structure and make hypothesis on parameters which could influence our predictive algorithm models.
Afterwards, we will bring out correlation between variables and observations.

A - Distribution analysis

We will start by analyzing the number of pregnancies distribution by woman.

A.1 - Pregnancies distribution

The data analysis of the number of pregnancies by woman show us the mean is about 3.85 while the median is equal to 3. As for her, the standard deviation is equal to 3.37.
The best number of pregnancies by woman is equal to 17 while the minimum of pregnancies by woman is equal to 0.

The first quantile is equal to 1 pregnancy. I.e. 25% of women have got less than 1 pregnancy whereas 75% of women have got less than 6 pregnancies (Third quantile) and 90% of them have less than 9 pregnancies.

Table 4: An overview of some quantiles and deciles of the number pregnancies by woman

	Value
10%	0
25%	1
50%	3
75%	6
90%	9
95%	10

Table 5: An overview of the principal statistical variables of the number of pregnancies by woman

min.	median	mean	standard_deviation	max.
0	3	3.85	3.37	17

Fig. 1 - An overview of the distribution of the number of pregnancies by woman.

Fig. 1.1 − Density curve of the number of pregnancies

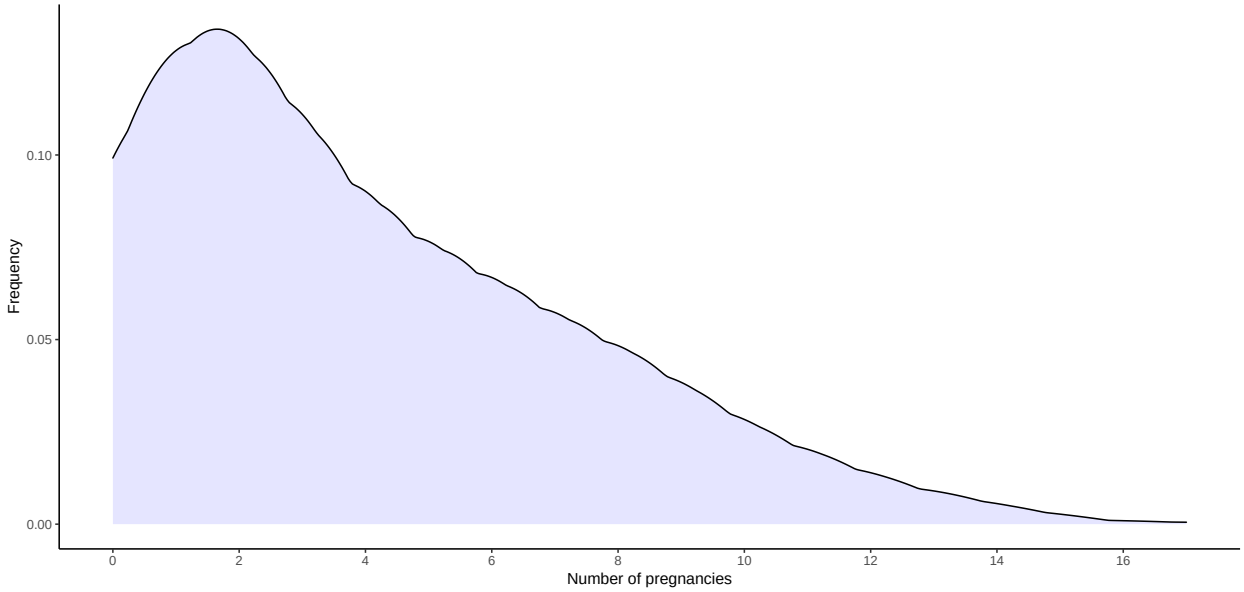
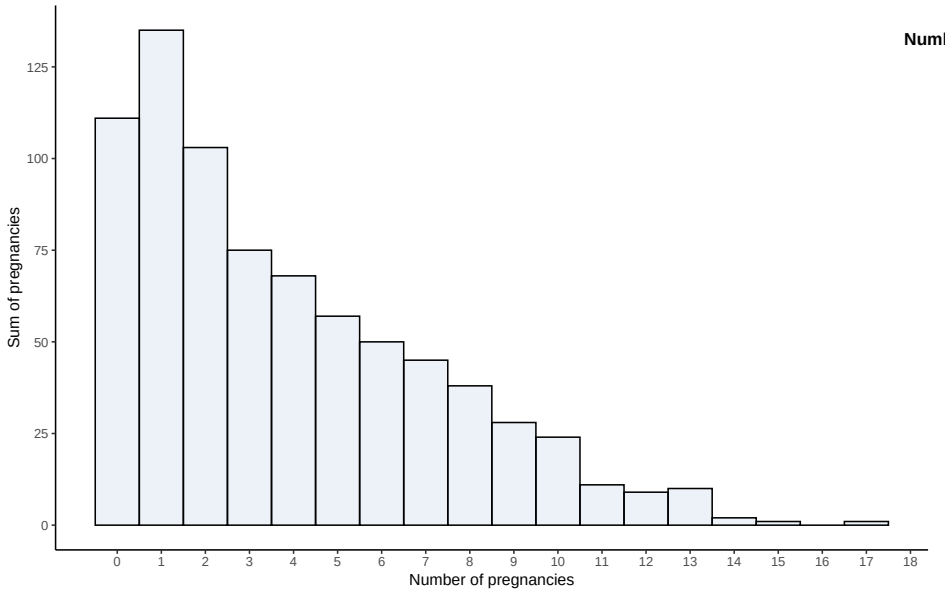


Fig. 1.3 − Histogram of the number of pregnancies



Number of pregnancies	Sum of pregnan
0	111
1	135
2	103
3	75
4	68
5	57
6	50
7	45
8	38
9	28
10	24
11	11
12	9
13	10
14	2
15	1
17	1

A.2 - Glucose distribution

The data analysis of the quantities of glucose concentration by woman show us the mean is about 122 while the median is equal to 117. As for her, the standard deviation is about 30.
The best quantities of glucose concentration by woman is equal to 199 while the minimum is equal to 44.

The first quantile is equal to 86. I.e. 25% of women have got less than 86 whereas 75% of women have got less than 141 quantities of glucose (Third quantile) and 90% of them have less than 167.

Table 6: An overview of some quantiles and deciles of the quantities of glucose by woman

	Value
10%	86.2
25%	99.0
50%	117.0
75%	141.0
90%	167.0
95%	181.0

Table 7: An overview of the principal statistical variables of the quantities of glucose by woman

min.	median	mean	standard_deviation	max.
44	117	121.69	30.54	199

Fig. 2 - An overview of the distribution of the quantities of glucose by woman.

Fig. 2.1 − Density curve of the quantities of glucose by woman

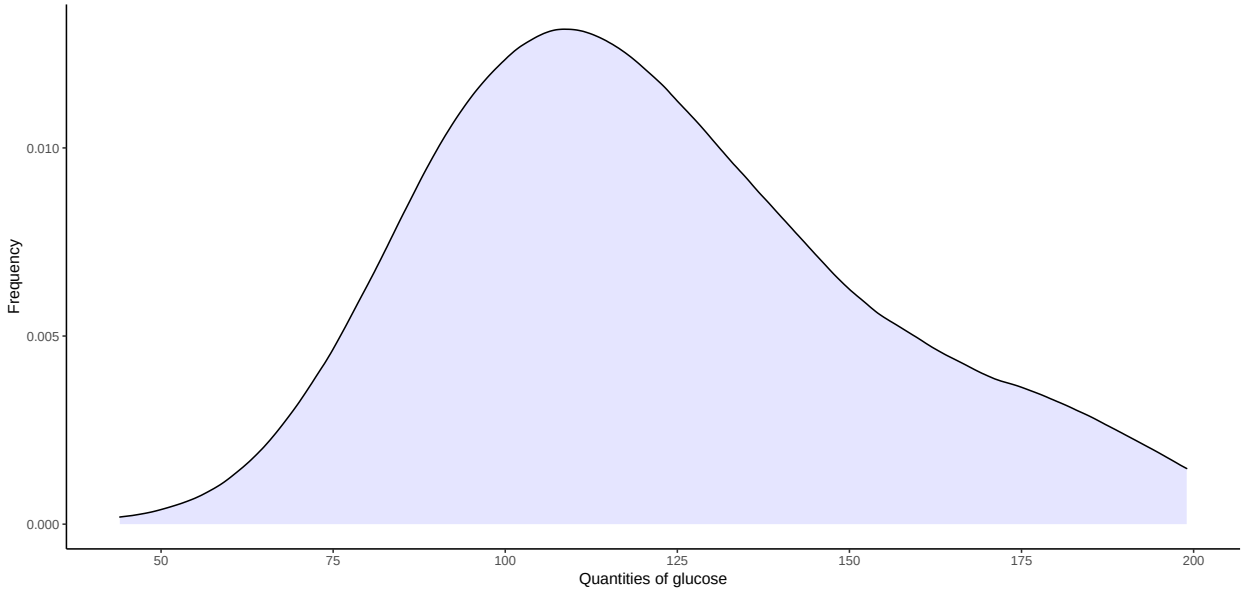
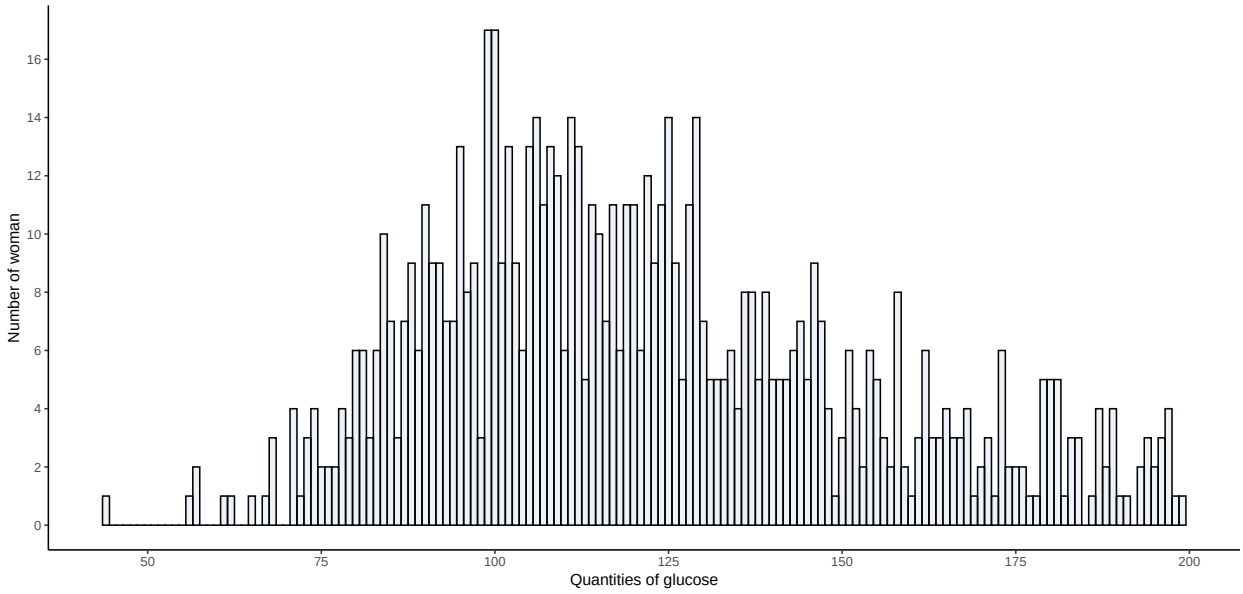


Fig. 2.3 − Histogram of the quantities of glucose by woman



A.3 - Blood pressure distribution

The data analysis of the diastolic blood pressure by woman show us the mean is about 72 mm/Hg while the median is equal to 72. As for her, the standard deviation is about 12.
The best diastolic blood pressure by woman is equal to 122 while the minimum is equal to 24.

The first quantile is equal to 64 mm/Hg. I.e. 25% of women have got less than 64 mm/hg whereas 75% of women have got less than 80 mm/Hg (Third quantile) and 90% of them have less than 88.

Table 8: An overview of some quantiles and deciles of the Diastolic blood pressure (mm Hg) by woman

	Value
10%	58
25%	64
50%	72
75%	80
90%	88
95%	92

Table 9: An overview of the principal statistical variables of the Diastolic blood pressure (mm Hg) by woman

min.	median	mean	standard_deviation	max.
24	72	72.41	12.38	122

Fig. 3 - An overview of the diastolic blood pressure by woman.

Fig. 3.1 − Density curve of the diastolic blood pressure by woman

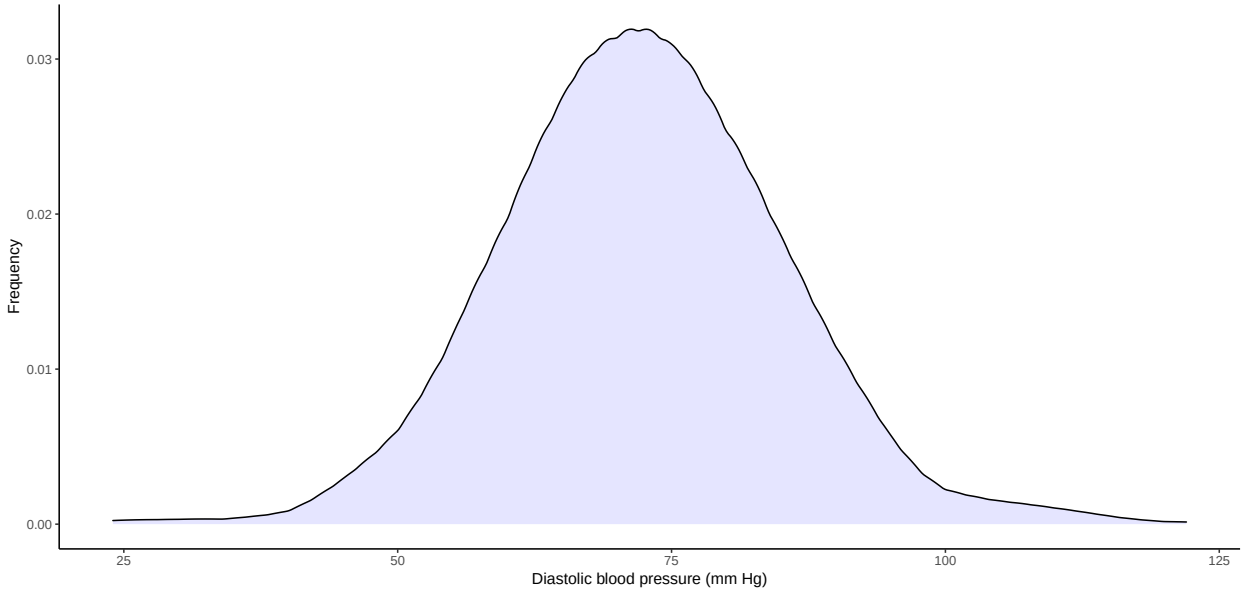
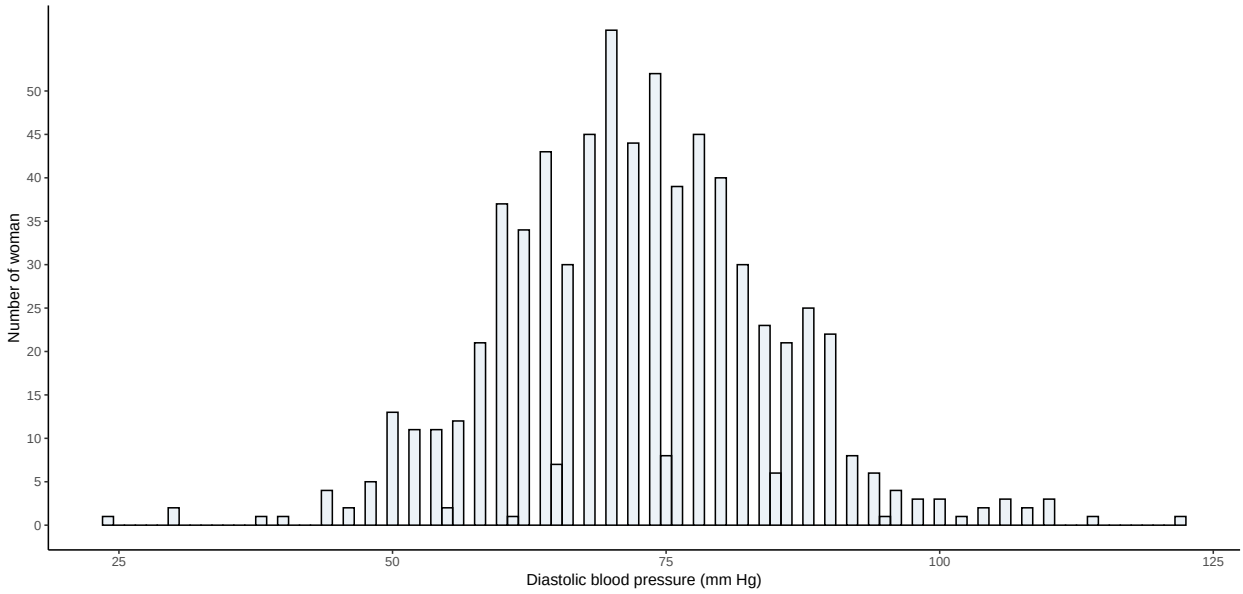


Fig. 3.2 − Histogram of the diastolic blood pressure by woman



A.4 - Triceps skin fold thickness

The data analysis of the Triceps skin fold thickness by woman shows us the mean is about 29 mm while the median is equal to 29. As for her, the standard deviation is about 10.
The most triceps skin fold thickness by woman is equal to 99 mm while the minimum is equal to 7 mm.

The first quantile is equal to 22 mm. I.e. 25% of women have got less than 22 whereas 75% of women have got less than 36 mm (Third quantile) and 90% of them have less than 42.

Table 10: An overview of some quantiles and deciles of the triceps skinfold thickness (mm) by woman

	Value
10%	16
25%	22
50%	29
75%	36
90%	42
95%	46

Table 11: An overview of the principal statistical variables of the triceps skinfold thickness (mm) by woman

min.	median	mean	standard_deviation	max.
7	29	29.15	10.48	99

Fig. 4 - An overview of the triceps skin fold thickness by woman.

Fig. 4.1 − Density curve of the Triceps skin fold thickness by woman

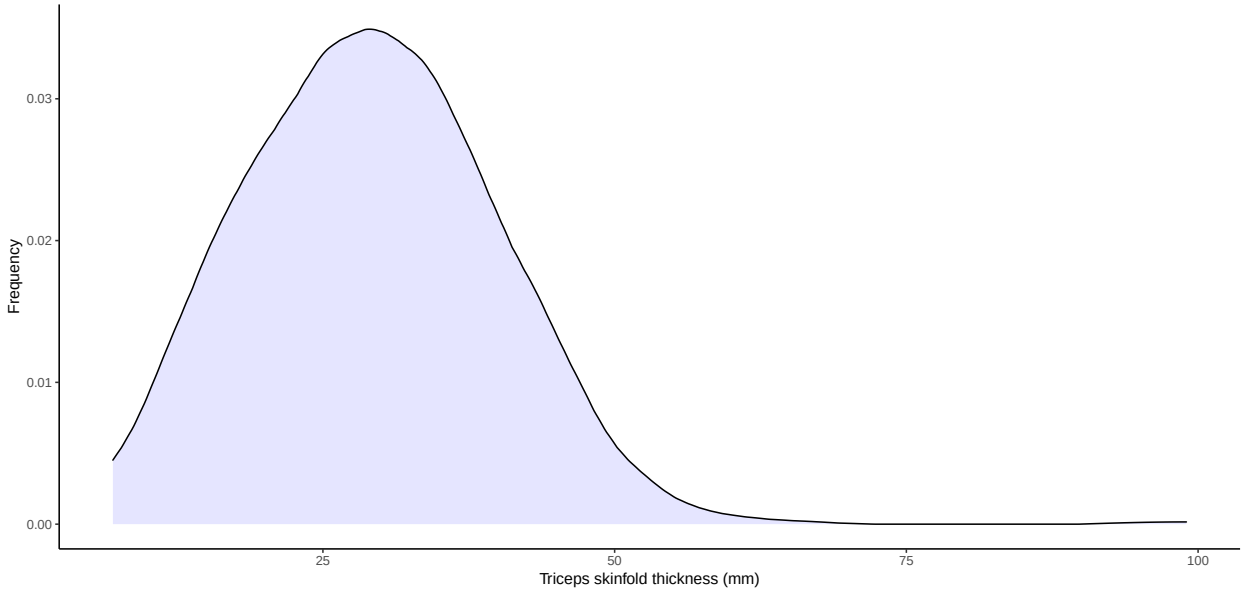
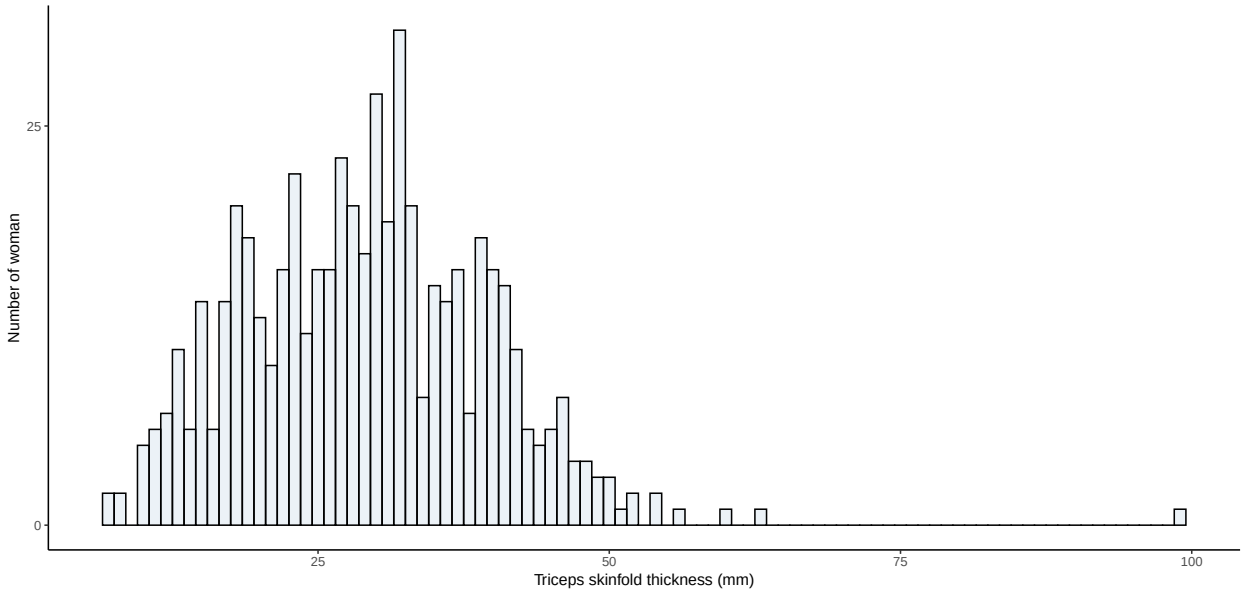


Fig. 4.2 − Histogram of the Triceps skinfold thickness by woman



A.5 - Insulin level distribution

The data analysis of the insulin by woman show us the mean is about 156 $\mu U/ml$ while the median is equal to 125. As for her, the standard deviation is about 119.
The most insulin level by woman is equal to 846 while the minimum is equal to 14 $\mu U/ml$.

The first quantile is equal to 50 $\mu U/ml$. I.e. 25% of women have got less than 50 $\mu U/ml$ whereas 75% of women have got less than 190 $\mu U/ml$ (Third quantile) and 90% of them have less than 292.

Table 12: An overview of some quantiles and deciles of the insulin level (mu U/ml) by woman

	Value
10%	50.30
25%	76.25
50%	125.00
75%	190.00
90%	292.40
95%	395.50

Table 13: An overview of the principal statistical variables of the insulin level (mu U/ml) by woman

min.	median	mean	standard_deviation	max.
14	125	155.55	118.78	846

Fig. 5 - An overview of the insulin level (mu U/ml) by woman

Fig. 5.1 − Density curve of insulin level by woman

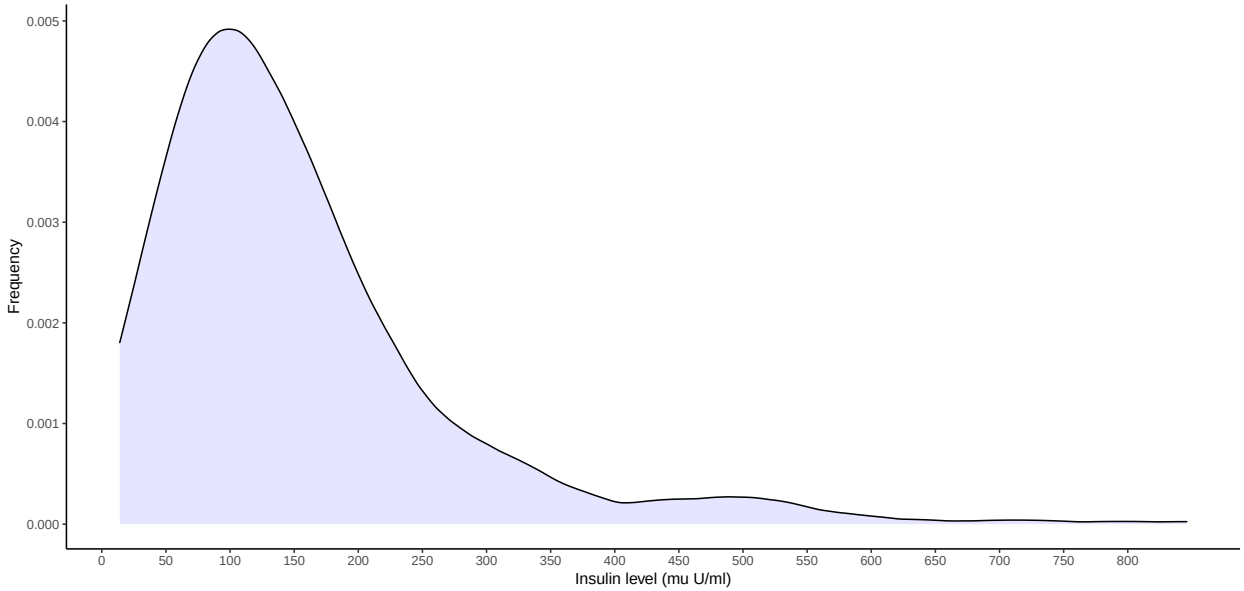
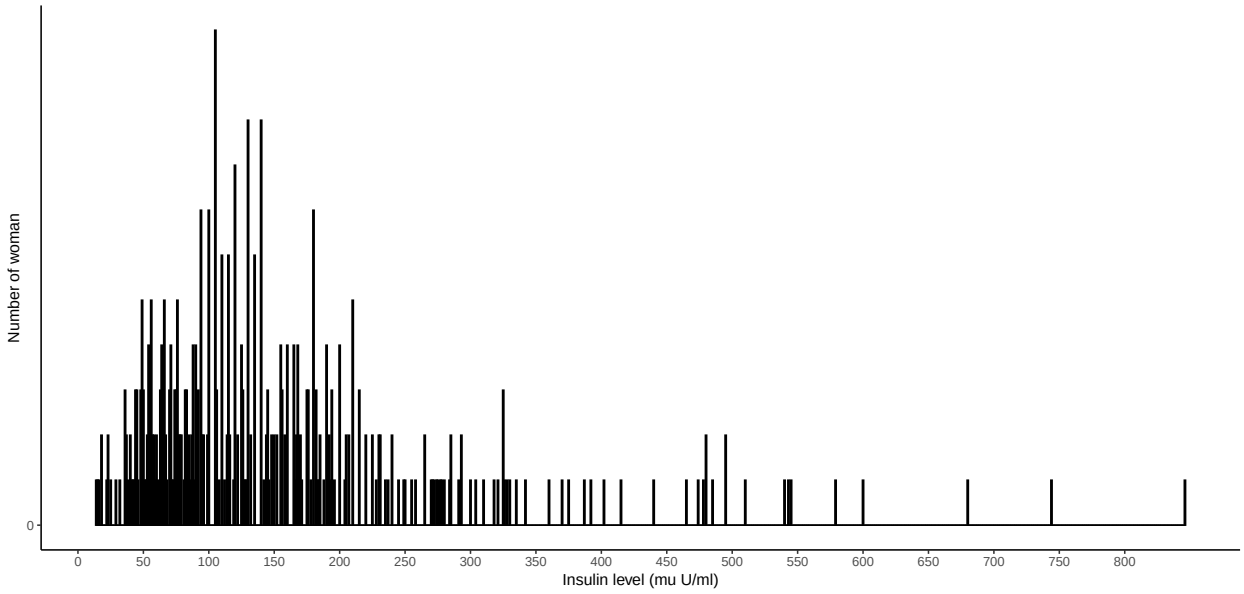


Fig. 5.2 − Histogram of the insulin level by woman



A.6 - Body mass index (BMI) distribution

The data analysis of the body mass index by woman show us the mean is about 32 while the median is equal to 32. As for her, the standard deviation is about 7.
The best BMI by woman is equal to 67 while the minimum is equal to 18.

The first quantile is equal to 27.50. i.e. 25% of women have got less than 27.50 whereas 75% of women have got less than 37 (Third quantile) and 90% of them have less than 42.

Table 14: An overview of some quantiles and deciles of the body mass index (BMI) by woman

	Value
10%	24.00
25%	27.50
50%	32.30
75%	36.60
90%	41.62
95%	44.50

Table 15: An overview of the principal statistical variables of the body mass index (BMI) by woman

min.	median	mean	standard_deviation	max.
18.2	32.3	32.46	6.92	67.1

Fig. 6 - An overview of the body mass index (BMI) by woman

Fig. 6.1 − Density curve of the BMI by woman

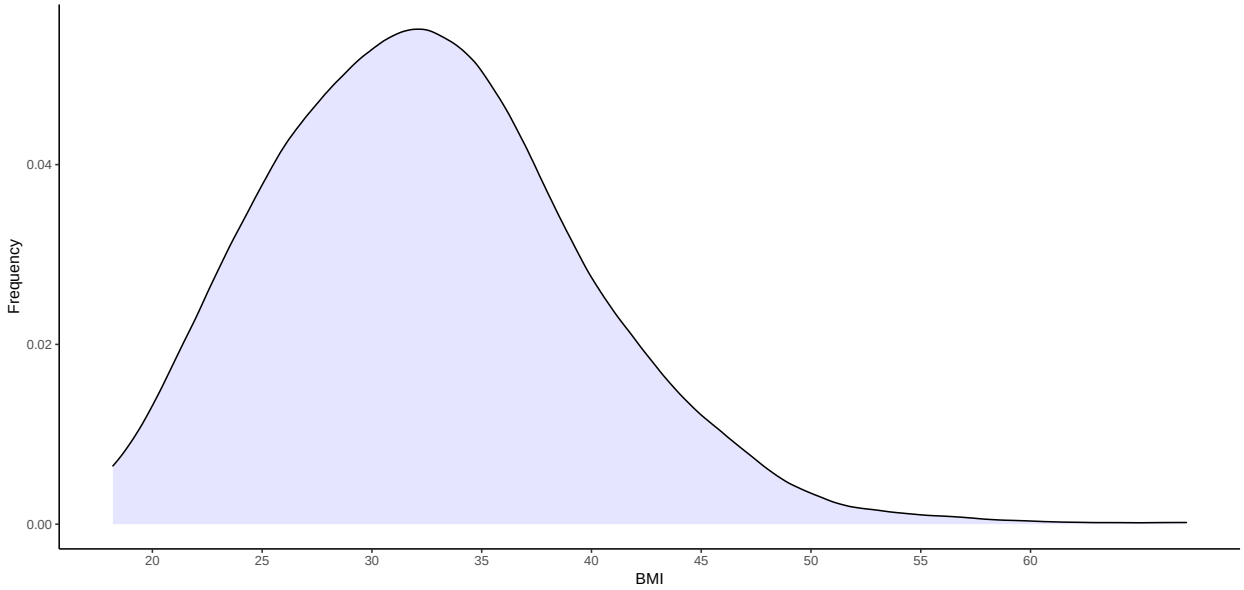
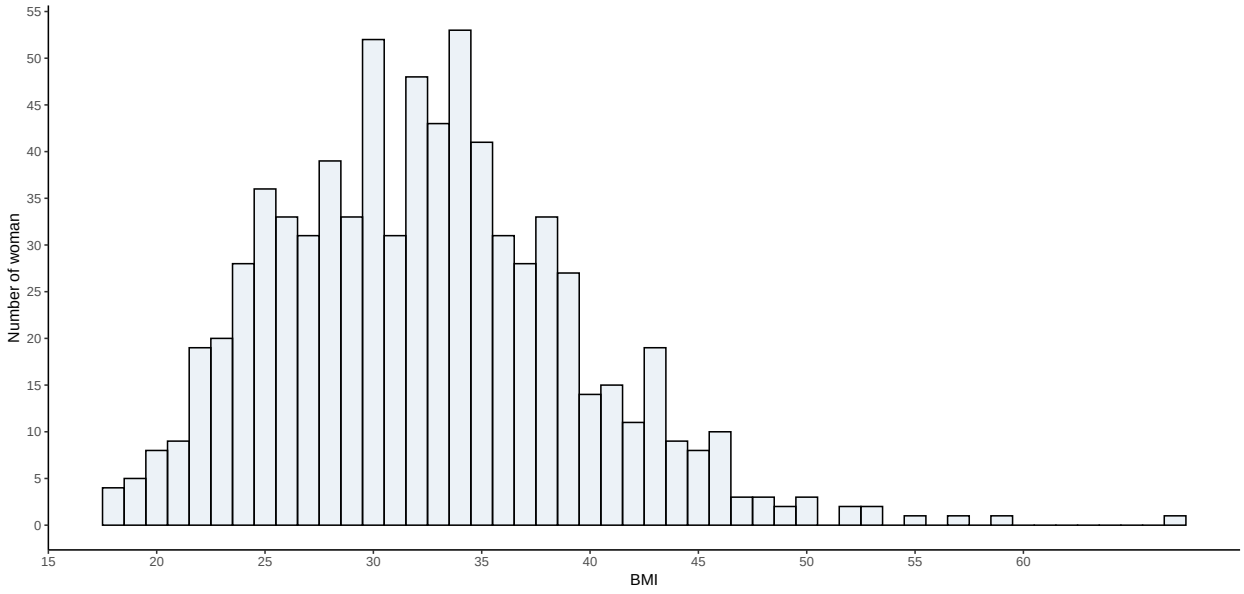


Fig. 6.2 − Histogram of the BMI by woman



A.7 - Diabetes pedigree function distribution

The data analysis of diabetes pedigree function by woman show us the mean is about 0.47 while the median is equal to 0.37 As for her, the standard deviation is about 0.33.
The best diabetes pedigree function by woman is equal to 2.42 while the minimum is equal to 0.08.

The first quantile is equal to 0.16. I.e. 25% of women have got less than 0.16 whereas 75% of women have got less than 0.62 (Third quantile) and 90% of them have less than 0.88.

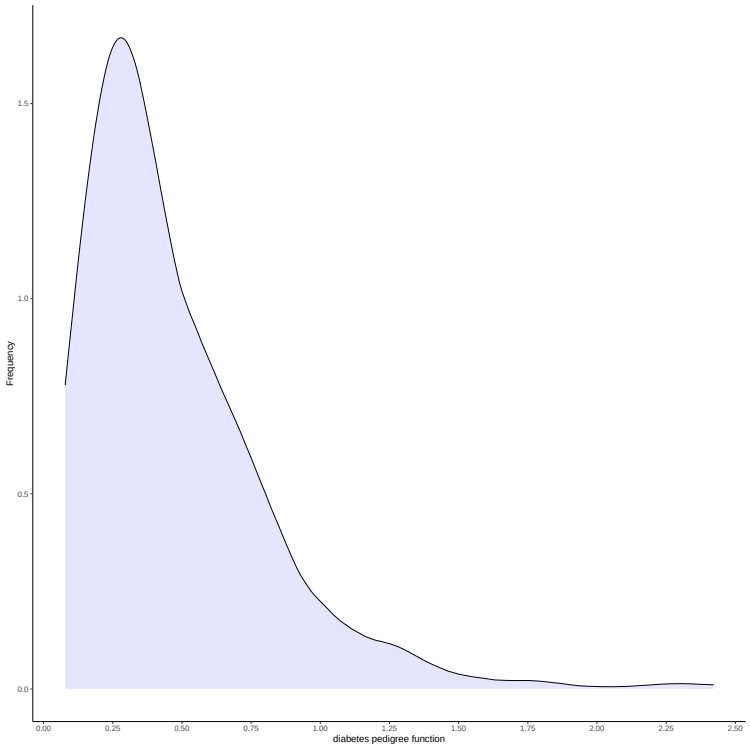
Table 16: An overview of some quantiles and deciles of the diabetes pedigree function by woman

	Value
10%	0.16500
25%	0.24375
50%	0.37250
75%	0.62625
90%	0.87860
95%	1.13285

Table 17: An overview of the principal statistical variables of the diabetes pedigree function by woman

min.	median	mean	standard_deviation	max.
0.08	0.37	0.47	0.33	2.42

Fig. 7 - An overview of the diabetes pedigree function by woman



A.8 - Age distribution

The data analysis of the age by woman show us the mean is about 33 while the median is equal to 29 As for her, the standard deviation is about 12.
The most age by woman is equal to 81while the minimum is equal to 21.

The first quantile is equal to 24. I.e. 25% of women have got less than 24 years old whereas 75% of women have got less than 41 (Third quantile) and 90% of them have less than 51.

Table 18: An overview of some quantiles and deciles of the age by woman

	Value
10%	22
25%	24
50%	29
75%	41
90%	51
95%	58

Table 19: An overview of the principal statistical variables of the age by woman

min.	median	mean	standard_deviation	max.
21	29	33.24	11.76	81

Fig.8 - An overview of the age by woman

Fig. 8.1 − Density curve of the age by woman

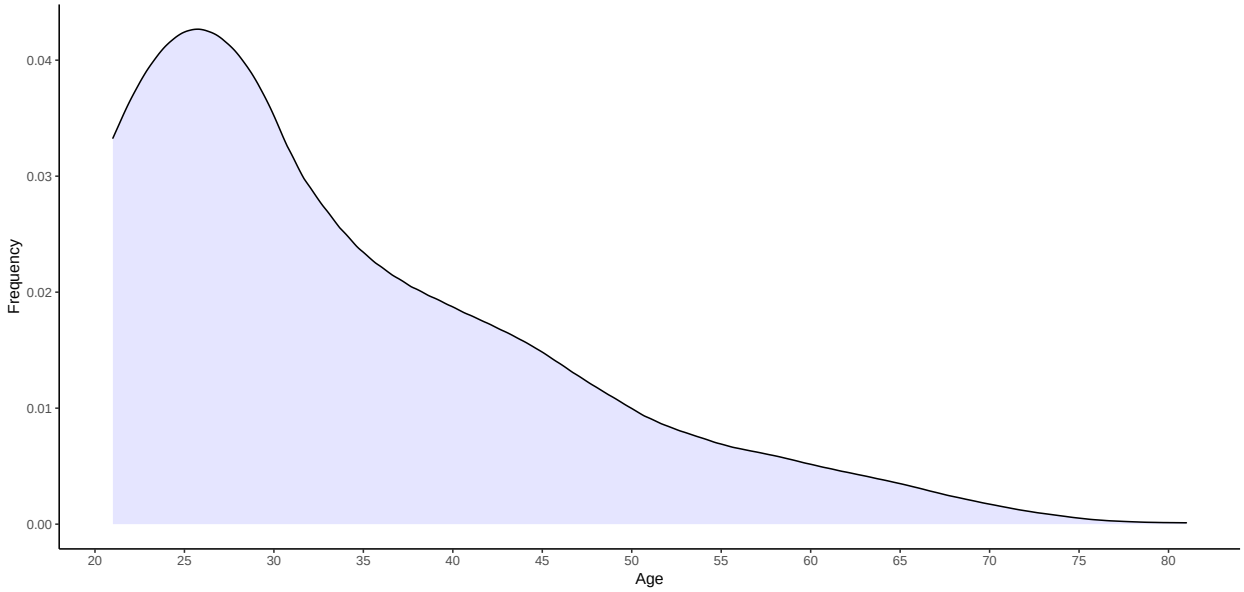
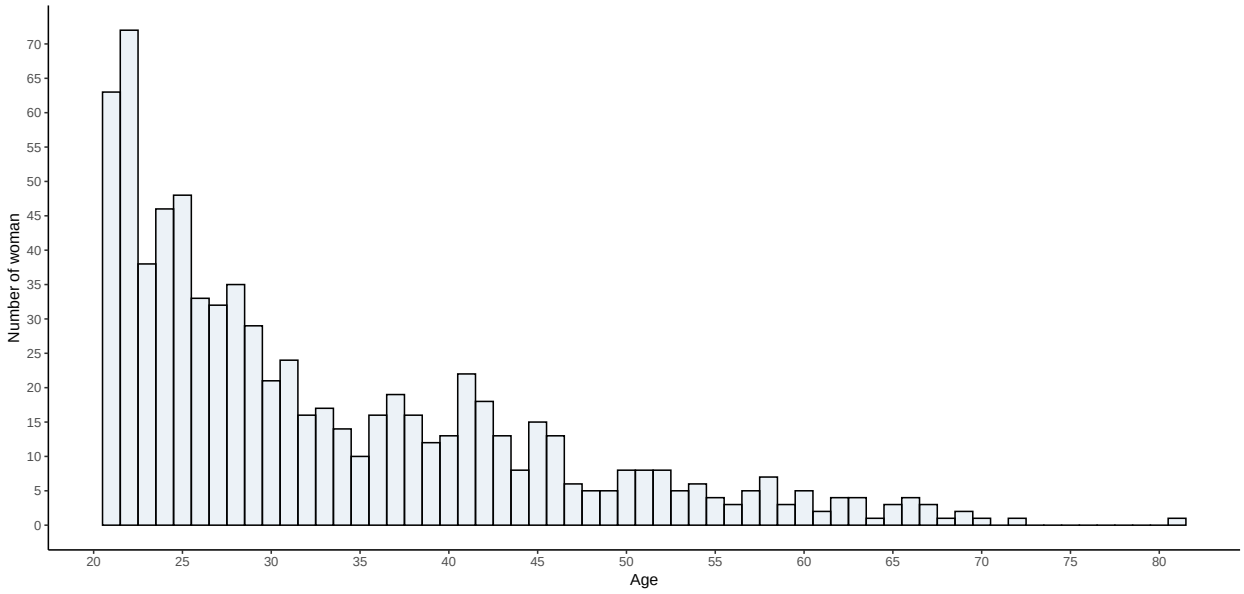


Fig. 8.2 − Histogram of the age by woman

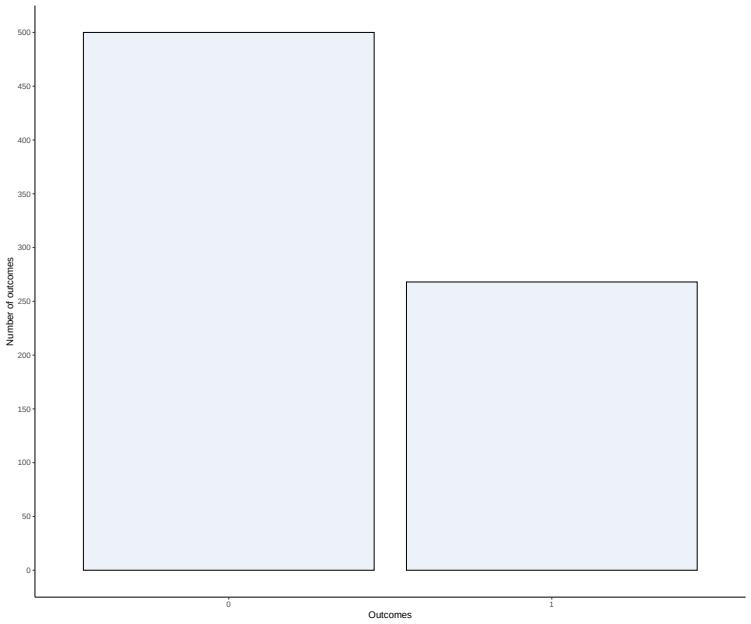


A.9 - Outcomes distribution

The data analysis of the outcome by woman highlights the fact that 500 women have not the diabetes while 268 of them are.

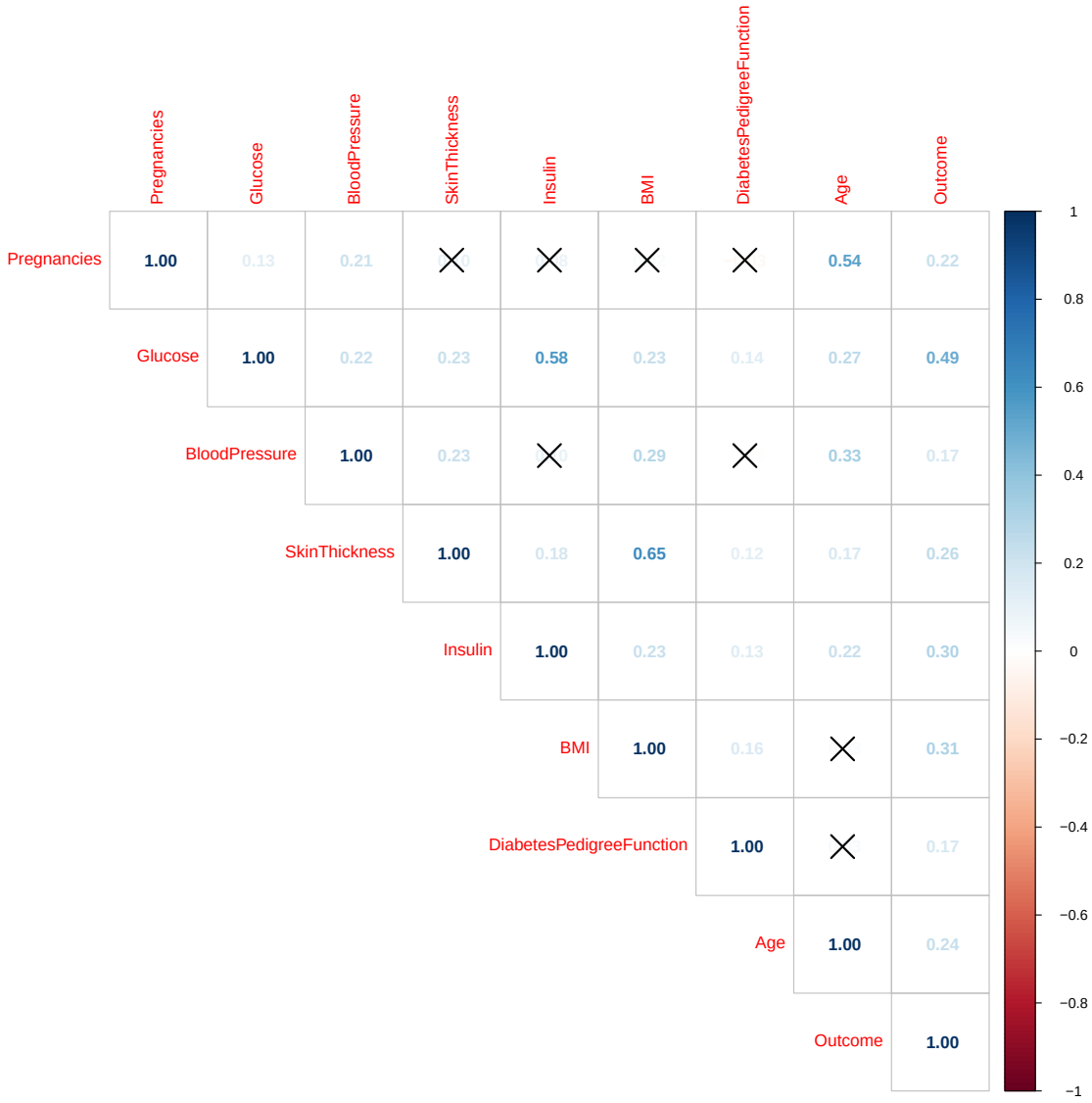
Fig. 9 - An overview of the outcome by woman

Diabetes	Number of women
0	500
1	268



B - Analysys of the correlation between variables

The figure below highligth the correlation between variables. We will see that the variable havean impact on the outcome variable. The variable which represents the best correlation with the “outcome” variable is the “glucose” variable.



Part III - Predictive algorithm models

Before building different predictive algorithm models, we will split the data set into a training test and a validation test.
he training test will be used to train our predictive algorithms and select the best parameters which we will permit us to evaluate his efficiency.
The test set will be used to evaluate our predictive algorithms. It should be noted we remove rows into test set which contains NA's values. It's a requirement in order to use the function "train" of the "Caret" package.

```
# We will split the data set into a training test and a validation test
# The training test will be used to train our predictive algorithms and
```

```r
#select the best parameters which we will permits us to minimize the RMSE
# The test set will be used to evaluate our predictive algorithms

## Validation set will be 20% of data set
set.seed(1, sample.kind="Rounding") # if using R 3.5 or earlier, use `set.seed(1)`
test_index <- createDataPartition(y = dataset_reprocessing$Outcome,
                                  times = 1, p = 0.2, list = FALSE)
train_set_2 <- dataset_reprocessing[-test_index,]
test_set_2 <- dataset_reprocessing[test_index,]

rm(test_index) # remove object

invisible(invisible(gc())) # for cleaning memory

test_set_2<- na.omit(test_set_2)
```

We will start by building a predictive algorithm based on logistic regression.

A - Generalized Linear model

We will start by training the algorithm with the GLM method by using the "train" function. We will use all the variables to predict outcomes.
Afterwards, we use the "predict" function to predict outcomes from the test set and in relation to the training set.
In order to evaluate the efficiency of our model, we will use the "confusionMatrix" function.

```r
# We will train the predictive algorithm
train_glm_2<- train(Outcome ~ ., method= "glm", data = train_set_2, na.action = na.omit)

## We will predict outcome from the test set
pred_glm_2<- predict(train_glm_2, test_set_2, na.action = na.omit)

## We will use "confusionMatrix" to evaluate the model
acc_glm_2<- confusionMatrix(pred_glm_2, test_set_2$Outcome)
```

The table below compares the predictions with the real outcomes from the data set.

Table 20: The outcomes of the prediction

	0	1
0	49	13
1	4	13

The table below shows us the different indicator of the predictive algorithm model. We will see an accuracy equal to 0.785 and a F1_score equal to 0.852.

Table 21: An abstract of the results of the different models

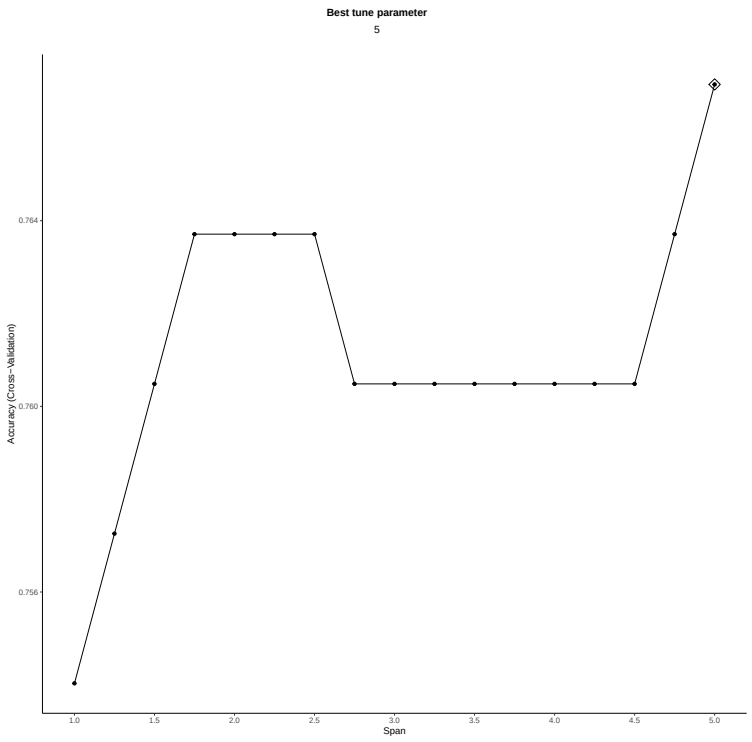
Model	Accuracy	F1_score	Prevalence	Precision	Recall	P_value
Generalized Linear model	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935

B - Generalized Additive Model using LOESS

We will use an another method of logictic regression whose name is "gamloess". We will also use cross validation in order to select the best tune parameter "span" to minimize the error rate of our model.

```
## We will train the predictive algorithm and select the best tune parameters
train_gamloess_2<- train(Outcome ~ ., method= "gamLoess", data = train_set_2,
                         tuneGrid= data.frame(span= seq(1, 5, .25), degree= 1),
                         trControl= trainControl(method = "cv", number = 10, p= 0.9),
                         na.action = na.omit)



## We will predict outcome from the test set
pred_gamloess_2<- predict(train_gamloess_2, test_set_2)

## We will use "confusionMatrix" to evaluate the model
acc_gamloess_2<- confusionMatrix(pred_gamloess_2, test_set_2$Outcome)
```

The figure below highlight results of the cross validation used to select the best tune parameter.



The table below compares the predictions with the real outcomes from the data set.

Table 22: The outcomes of the prediction

	0	1
0	49	13
1	4	13

The table below shows us the different indicator of the predictive algorithm model. We will see an accuracy equal to 0.785 and a F1_score equal to 0.852.

Table 23: An abstract of the results of the different models

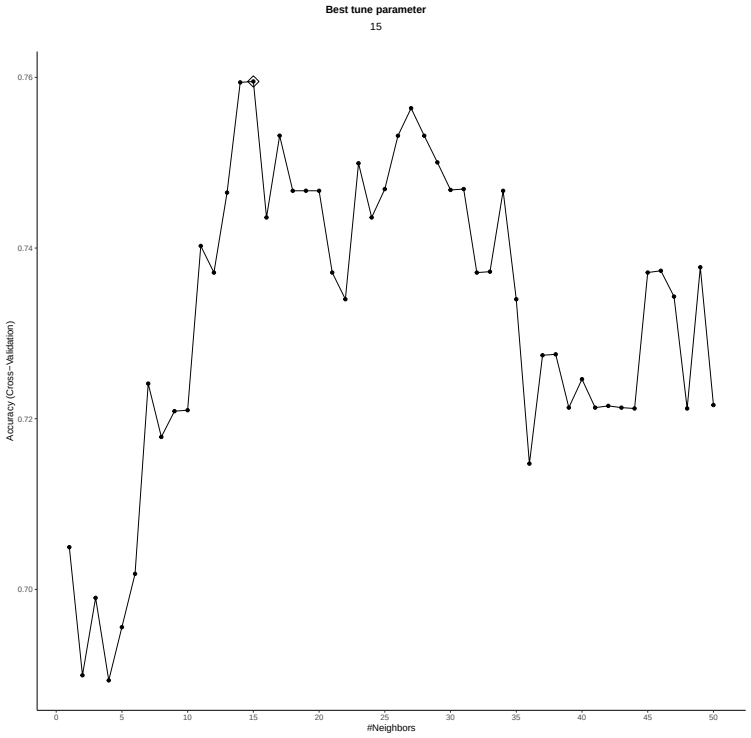
Model	Accuracy	F1_score	Prevalence	Precision	Recall	P_value
Generalized Linear model	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935
Generalized Additive Model using LOESS	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935

C - kNN model

We will use a knn model and cross validation to select the best tune parameter "k".

```
## We will train the predictive algorithm and select the best tune parameters
set.seed(1988, sample.kind="Rounding")
train_knn_2<- train(Outcome ~ ., method= "knn", data = train_set_2,
                    tuneGrid= data.frame(k= seq(1,50,1)),
                    trControl= trainControl(method = "cv", number = 10, p= 0.9),
                    na.action = na.omit)

## We will predict outcome from the test set
pred_knn_2<- predict(train_knn_2, test_set_2)

## We will use "confusionMatrix" to evaluate the model
acc_knn_2<- confusionMatrix(pred_knn_2, test_set_2$Outcome)
```

The figure below highlight results of the cross validation used to select the best tune parameter.



The table below compares the predictions with the real outcomes from the data set.

Table 24: The outcomes of the prediction

	0	1
0	51	14
1	2	12

The table below shows us the different indicator of the predictive algorithm model. We will see an accuracy equal to 0.797 and a F1_score equal to 0.864.

Table 25: An abstract of the results of the different models

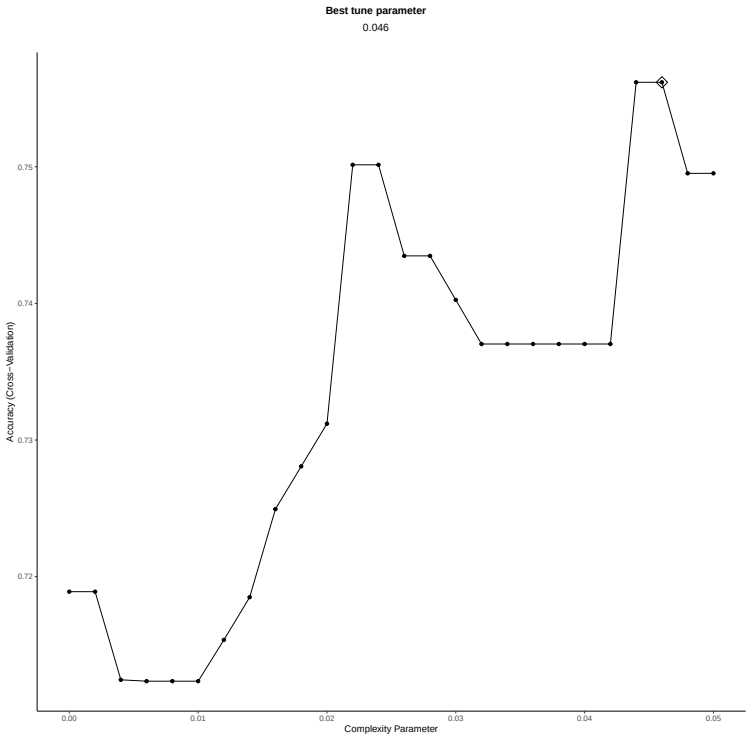
Model	Accuracy	F1_score	Prevalence	Precision	Recall	P_value
Generalized Linear model	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935
Generalized Additive Model using LOESS	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935
kNN Model	0.7974684	0.8644068	0.6708861	0.7846154	0.9622642	0.0093854

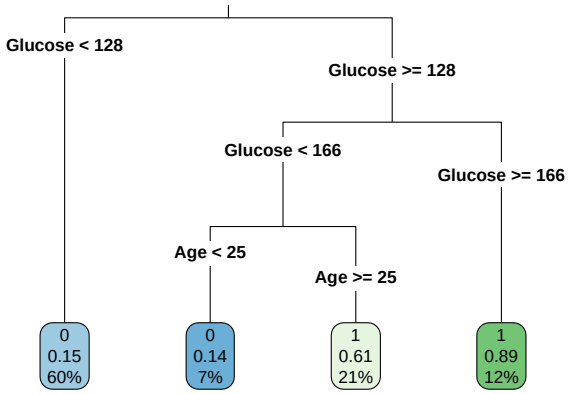
D - Classification tree model

We will use a classification model and cross validation in order tho select the best tune parameter "cp".

```
## We will train the predictive algorithm and select the best tune parameters
set.seed(1988, sample.kind="Rounding")
train_rpart_2<- train(Outcome ~ ., method= "rpart", data = train_set_2,
                      tuneGrid= data.frame(cp = seq(0, 0.05, 0.002)),
                      trControl= trainControl(method = "cv", number = 10, p= 0.9),
                      na.action = na.omit)



## We will predict outcome from the test set
pred_rpart_2<- predict(train_rpart_2, test_set_2)

## We will use "confusionMatrix" to evaluate the model
acc_rpart_2<- confusionMatrix(pred_rpart_2, test_set_2$Outcome)
```

The figures below show us the result of the cross validation and the decision tree used by the model to predict outcomes.


Best tune parameter
0.046
Accuracy (Cross−Validation)
0.75
0.74
0.73
0.72
0.00
0.01
0.02
0.03
0.04
0.05
Complexity Parameter



The table below compares the predictions with the real outcomes from the data set.

Table 26: The outcomes of the prediction

	0	1
0	48	9
1	5	17

The table below shows us the different indicator of the predictive algorithm model. We will see an accuracy equal to 0.822 and a F1_score equal to 0.873.

Table 27: An abstract of the results of the different models

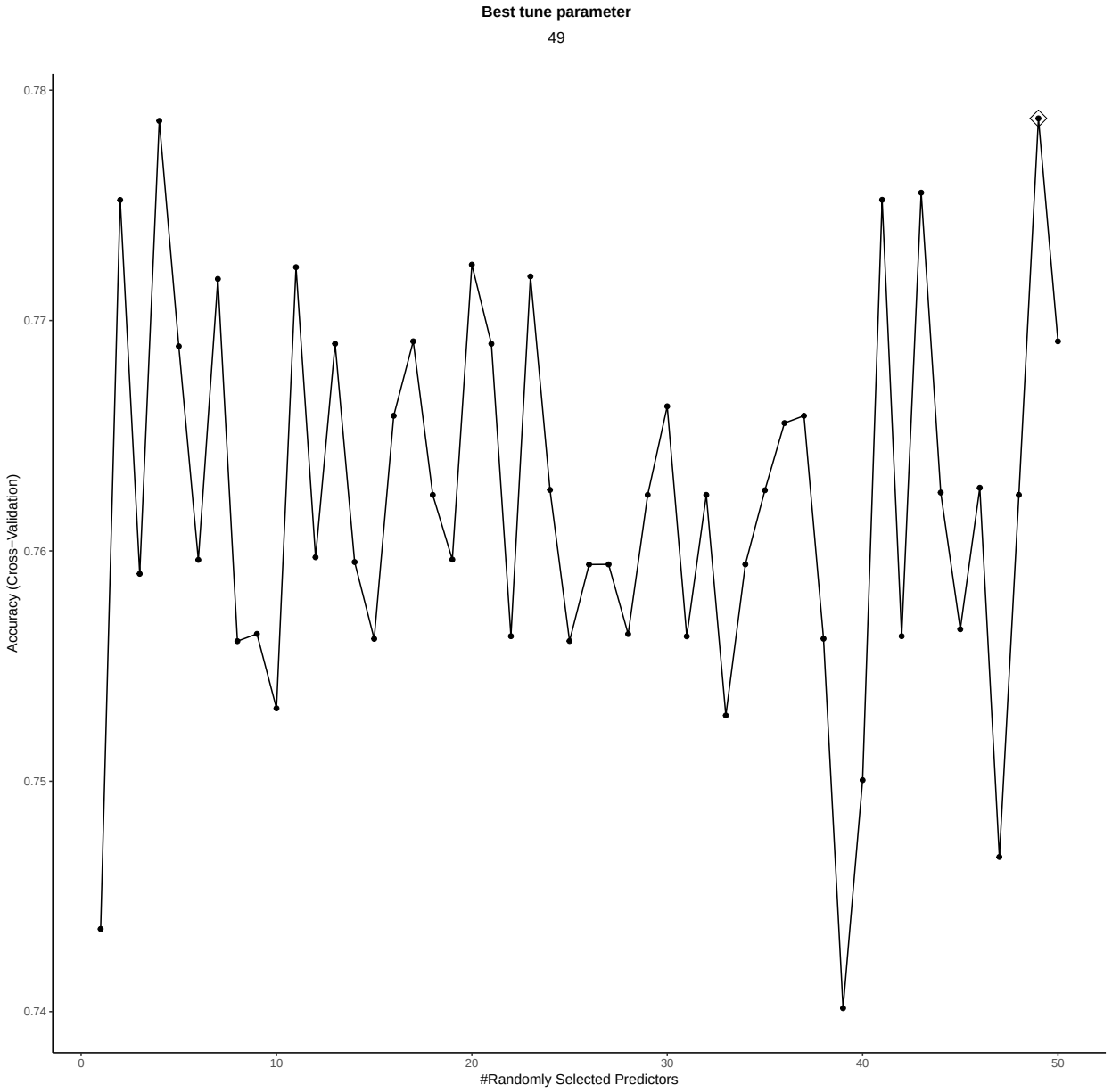
Model	Accuracy	F1_score	Prevalence	Precision	Recall	P_value
Generalized Linear model	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935
Generalized Additive Model using LOESS	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935
kNN Model	0.7974684	0.8644068	0.6708861	0.7846154	0.9622642	0.0093854
Classification tree model	0.8227848	0.8727273	0.6708861	0.8421053	0.9056604	0.0019884

E - Random Forest model

We will use a random forest model and cross validation in order to select the best tune parameter "mtry".

```
## We will train the predictive algorithm and select the best tune parameters
set.seed(1988, sample.kind="Rounding")
train_rf_2<- train(Outcome ~ ., method= "rf", data = train_set_2,
                   ntree= 100,
                   tuneGrid= data.frame(mtry= seq(1:50)),
                   trControl= trainControl(method = "cv", number = 10, p= 0.9), na.action = na.omit)



## We will predict outcome from the test set
pred_rf_2<- predict(train_rf_2, test_set_2)

## We will use "confusionMatrix" to evaluate the model
acc_rf_2<- confusionMatrix(pred_rf_2, test_set_2$Outcome)
```

The figure below shows us the result of the cross validation.



The table below compares the predictions with the real outcomes from the data set.

Table 28: The outcomes of the prediction

	0	1
0	47	12
1	6	14

The table below shows us the different indicator of the predictive algorithm model. We will see an accuracy equal to 0.772 and a F1_score equal to 0.839.

Table 29: An abstract of the results of the different models

Model	Accuracy	F1_score	Prevalence	Precision	Recall	P_value
Generalized Linear model	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935
Generalized Additive Model using LOESS	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935
kNN Model	0.7974684	0.8644068	0.6708861	0.7846154	0.9622642	0.0093854
Classification tree model	0.8227848	0.8727273	0.6708861	0.8421053	0.9056604	0.0019884
Random Forest model	0.7721519	0.8392857	0.6708861	0.7966102	0.8867925	0.0333457

F - Random Forest model with Rborist method

We will use the random forest principle with Rborist method and croos validation to select the best tune parameters "minNode" and "predFixed".

```
## We will train the predictive algorithm and select the best tune parameters
set.seed(1988, sample.kind="Rounding")
train_rborist_2<- train(Outcome ~ ., method= "Rborist", data = train_set_2,
                        tuneGrid= data.frame(minNode= seq(1:10), predFixed= seq(1:50)),
                        trControl= trainControl(method = "cv", number = 10, p= 0.9),
                        na.action = na.omit)

## We will represent the best tune parameter
train_rborist_2$finalModel$tuneValue %>%
  knitr::kable(caption = "The best Tune parameter") %>%
  kable_styling(latex_options = "HOLD_position")
```

Table 30: The best Tune parameter

	predFixed	minNode
36	8	8

```
## We will predict outcome from the test set
pred_rborist_2<- predict(train_rborist_2, test_set_2)

## We will use "confusionMatrix" to evaluate the model
acc_rborist_2<- confusionMatrix(pred_rborist_2, test_set_2$Outcome)
```

The table below compares the predictions with the real outcomes from the data set.

Table 31: The outcomes of the prediction

	0	1
0	47	12
1	6	14

The table below shows us the different indicator of the predictive algorithm model. We will see an accuracy equal to 0.772 and a F1_score equal to 0.839.

Table 32: An abstract of the results of the different models

Model	Accuracy	F1_score	Prevalence	Precision	Recall	P_value
Generalized Linear model	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935
Generalized Additive Model using LOESS	0.7848101	0.8521739	0.6708861	0.7903226	0.9245283	0.0182935
kNN Model	0.7974684	0.8644068	0.6708861	0.7846154	0.9622642	0.0093854
Classification tree model	0.8227848	0.8727273	0.6708861	0.8421053	0.9056604	0.0019884
Random Forest model	0.7721519	0.8392857	0.6708861	0.7966102	0.8867925	0.0333457
Random Forest model with Rborist method	0.7721519	0.8392857	0.6708861	0.7966102	0.8867925	0.0333457

Conclusion

Our different models have got an accuracy between 0.77 and 0.82 while the F1_score is between 0.83 and 0.87. The efficiency of our predictive algorithm model are satisfactory.
However, these results these results should be qualified with regard to the few observations contained in the data set.
it should also be noted that the data set contained a lot of missing value.