Predictive Models for Coeliac Disease Diagnose

Health Data Analytics - Project

by Marta Tosolini

Project Introduction

Coeliac disease is a serious **autoimmune** disease that occurs in genetically predisposed people where the ingestion of gluten leads to damage in the small intestine. It is estimated to affect 1 in 100 people worldwide, but only about 30% are properly diagnosed.
Dataset source: https://www.kaggle.com/datasets/jackwin07/celiac-disease-coeliac-disease
There are multiple **risk factors** that can be associated with coeliac disease, such as **family history**, diseases associated (**diabetes, down syndrome**).
The symptoms can be both **typical** and **atypical** for patients.
Scientific Questions of the study:

- What is the probability (risk) to be diagnosed with Coeliac Disease?
- What are the major risk factors of developing Coeliac Disease?

Libraries

```
library(tidyverse) #tidy data
library(e1071) #statistics
library(DataExplorer) #IDA
library(SmartEDA) #IDA
library(summarytools) #IDA
library(ggplot2) #plots
library(moments) #kurtosis
library(dplyr) #tidy data
library(MASS) #statistics
library(boot) #statistics
library(rms) #confounding variables
library(caret) #cross validation
library(ROCit) #roc curve
library(generalhoslem) #calibration test
library(predtools) #plot calibration
library(rfUtilities) #calibration
library(randomForest) #randomForest algorithm
library(gt) #table
```

1. IDA

We perform some **Initial Data Analysis** to assess:

- **Type** of Variables
- Variables **frequencies**
- **Missing data/Errors**

It will help us to explore the dataset and identify as well the possible variables to consider to train the models we will define.

Let's read the csv dataset (adding IDs as well):

```
data <- read_csv('celiac_disease_lab_data.csv')
## Rows: 2206 Columns: 15
data$Disease_Diagnose = as.factor(data$Disease_Diagnose)
data <- tibble::rowid_to_column(data, "ID")
names(data)
##  [1] "ID"             "Age"            "Gender"         "Diabetes"
##  [5] "Diabetes Type"  "Diarrhoea"      "Abdominal"      "Short_Stature"
##  [9] "Sticky_Stool"   "Weight_loss"    "IgA"            "IgG"
## [13] "IgM"            "Marsh"          "cd_type"        "Disease_Diagnose"
```

Variables Brief Description

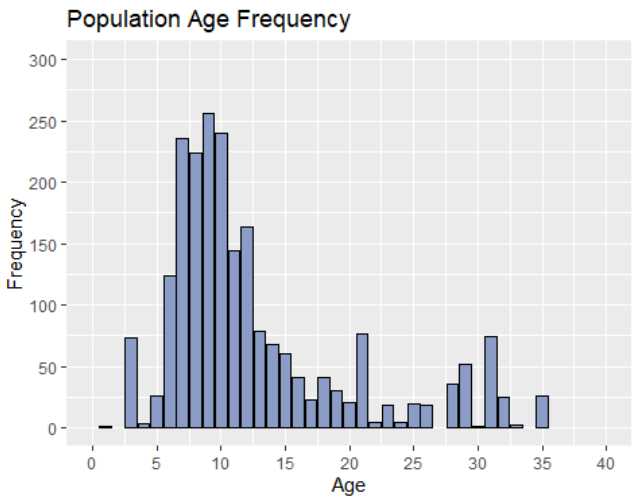
"Age" - The age of the patient
"Gender" - The gender of the patient (Male/Female)
"Diabetes" - Presence/Absence of Diabetes disease
"Diabetes Type" - Type of Diabetes, if present (none, Type 1/2)
"Diarrhoea" - Presence/Absence of Diarrhoea symptom
"Abdominal" - Presence/Absence of Abdominal pain symptom
"Short_Stature" - Type of short stature: Normal Variant, Psychosocial (PSS), Disproportionate (DSS)
"Sticky_Stool" - Presence/Absence of Sticky Stool symptom
"Weight_loss" - Presence/Absence of Weight loss symptom
"IgA" - Immunoglobulin A value
"IgG" - Immunoglobulin G value
"IgM" - Immunoglobulin M value
"Marsh" - Marsh Classification of coeliac disease: 0,1,2,3a,3b,3c
"cd_type" - Coeliac disease type: silent, typical, atypical, latent
"Disease_Diagnose" - Presence/Absence of Coeliac disease

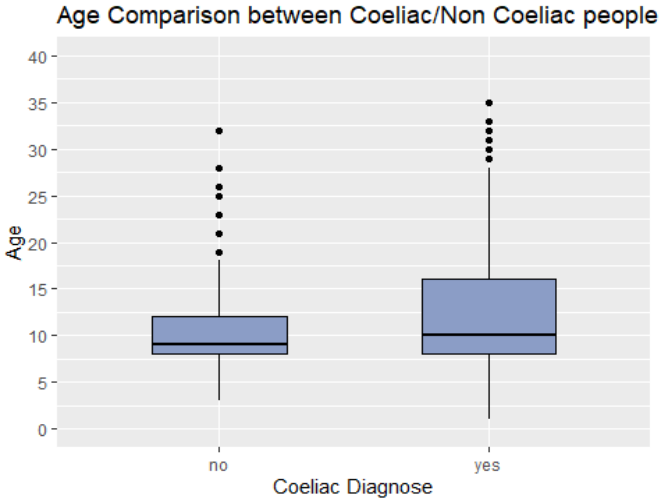
Age

Let's start by plotting the distribution of the **Age** in the population:

```
ggplot(data=data) +
 geom_bar(aes(x=Age), width = 0.9, color="black", fill='#8B9DC6') +
 labs(title="Population Age Frequency") +
 xlab("Age") + ylab("Frequency") +
 expand_limits(x=c(0,40), y=c(0, 300)) +
 scale_x_continuous(breaks = c(0,5,10,15,20,25,30,35,40)) +
 scale_y_continuous(breaks = c(0,50,100,150,200,250,300))
```

```
ggplot(data=data, aes(y=Age, x=Disease_Diagnose)) +
 geom_boxplot(color="black", fill='#8B9DC6', width=0.5) +
 labs(title="Age Comparison between Coeliac/Non Coeliac people") +
 xlab("Coeliac Diagnose") + ylab("Age") +
 expand_limits(y=c(0,40)) +
 scale_y_continuous(breaks = c(0,5,10,15,20,25,30,35,40))
```






```
print('Summary of Population Age:')
summary(data$Age)
print(paste('Skewness Value:', skewness(data$Age)) )
print(paste('Kurtosis Value:',kurtosis(data$Age)) )
## [1] "Summary of Population Age:"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    8.00   10.00   12.77   15.00   35.00
## [1] "Skewness Value: 1.37236011276209"
## [1] "Kurtosis Value: 4.03670517887152"
```

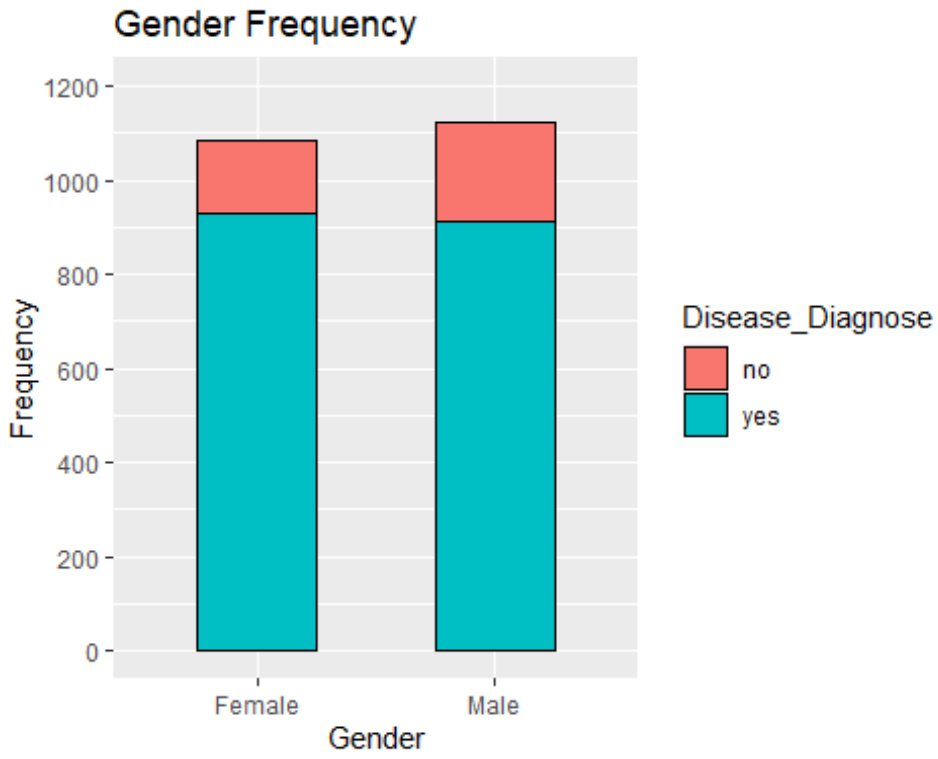
We can see that most of the patients in the Population are **children**, around **8-12 years old**.
The distribution of ages is **asymmetric**, **skewed** to the **right** (skewness = 1.37 > 0), median is less than the mean. **Kurtosis** is greater than 3, as we could see in the plot it has high peak **(leptokurtic**).
By doing a comparison between Age frequency of Coeliac and non-Coeliac diagnosed people (box plot), the 3rd quantile is higher for coeliac case, we can see outliers for both cases.

We have to remember that the dataset is **unbalanced** between positives and negatives, so the comparisons may be not accurate enough, by the way the distribution is a bit different for the two classes, even if the median is almost the same for both cases (child age: around 8-10).
Non coeliac diagnosed people seems older.

Gender

Let's plot the **Gender** distribution, by taking into account presence/absence of Coeliac disease:

```
ggplot(data=data) +
 geom_bar(aes(x=Gender, fill=Disease_Diagnose), color='black', width = 0.5) +
 labs(title="Gender Frequency") +
 xlab("Gender") + ylab("Frequency") +
 expand_limits(y=c(0, 1200)) +
 scale_y_continuous(breaks = c(0,200,400,600,800,1000,1200))
```

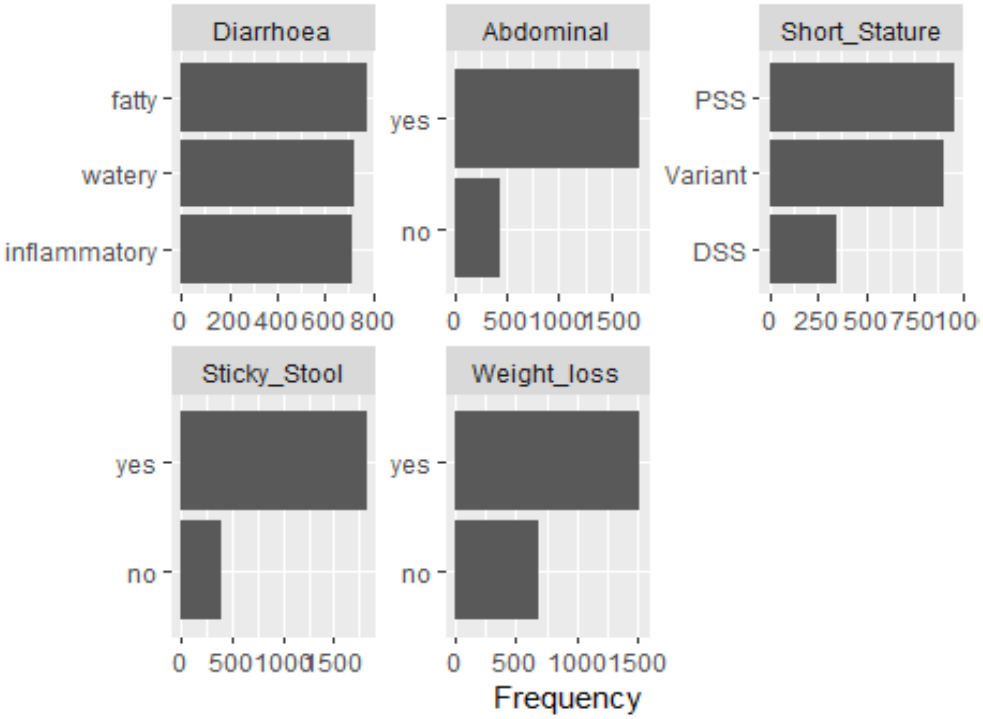


Gender seems almost **equally distributed** in frequencies, there are more males in dataset and also a little bit more males were not diagnosed with Coeliac disease (in proportion). In literature, like other autoimmune diseases, celiac disease occurs in more women than men.

Symptoms

Let's plot the frequency of **Symptoms** for all the patients:

```
plot_bar(data[6:10])
```

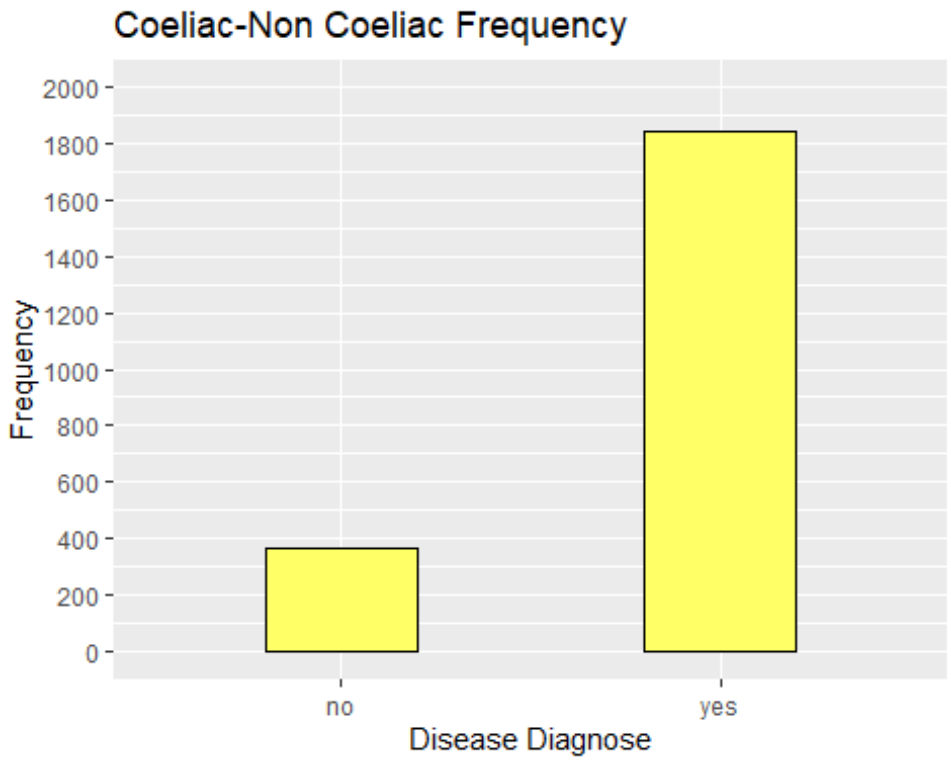


In the case of **Diarrhoea types**, they seem **equally distributed**. **Abdominal pain, Weight loss and Sticky stool** are more present than absent in this population. Disproportionate short stature (**DSS**) seems less frequent than **Variant** and **PSS** in general.

Celiac Diagnose Frequency

Let's compare the frequency of coeliac and non-coeliac diagnosed people:

```
ggplot(data=data) +
 geom_bar(aes(x=Disease_Diagnose), color='black', fill='#FFFF66', width = 0.4) +
 labs(title="Coeliac-Non Coeliac Frequency") +
 xlab("Disease Diagnose") + ylab("Frequency") +
 expand_limits(y=c(0, 2000)) +
 scale_y_continuous(breaks = c(0,200,400,600,800,1000,1200,1400,1600,1800,2000))
```

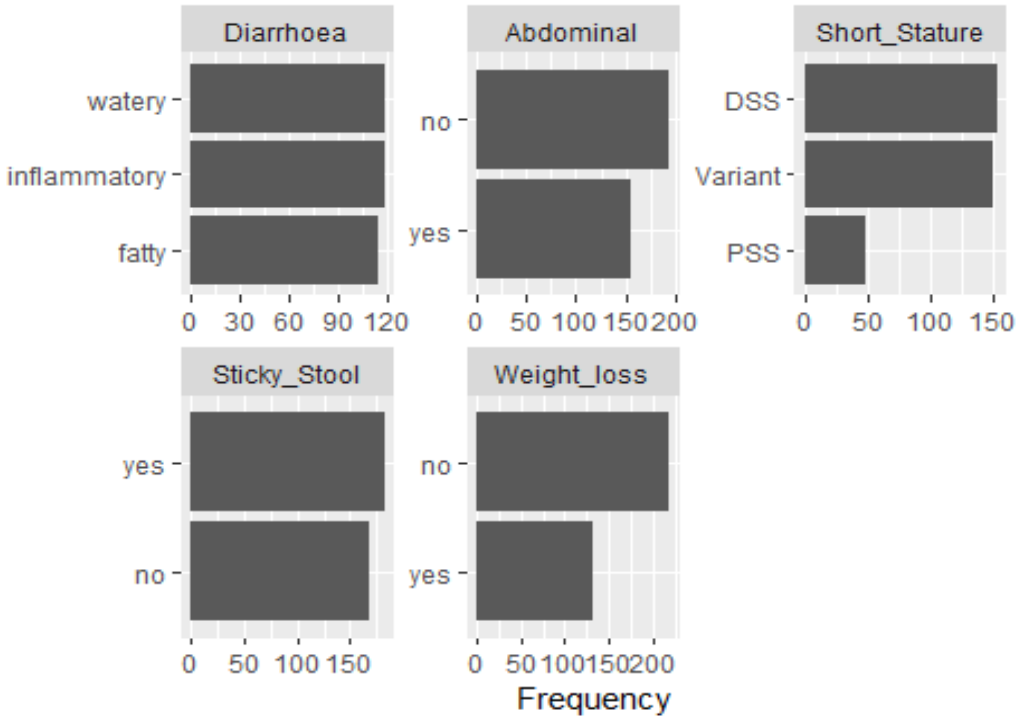



```
print('Proportions:')
print(paste( 'Coeliac: ', nrow(filter(data, Disease_Diagnose=='yes'))/nrow(data)))
print(paste( 'Non-Coeliac: ', nrow(filter(data, Disease_Diagnose=='no'))/nrow(data)))
## [1] "Proportions:"
## [1] "Coeliac:  0.835448776065277"
## [1] "Non-Coeliac:  0.164551223934723"
```

We immediately notice that the dataset is **inbalanced**, more than 83.5% of patients are diagnosed with coeliac disease in this dataset, when we build the model we will have to face this problem. Maybe we couldn't have enough information of negative diagnosed people, but more for positive ones.

It is intresting to plot the correlation between non coeliac patients and symptoms:

```
plot_bar(filter(data, cd_type == 'none')[6:10])
```



This suggest that probably **Diarrhoea** types **won't help us** to understand the presence of coeliac disease in a patients, because it seems present and equally distributed in each person
We can conclude that this dataset represents all patients correlated with diarrhoea symptoms, coeliac/not-coealiac diagnosed.
It seems a good idea to consider all the other symptoms, with the possibility to use short stature as well, because DSS seems more present in non-coeliac patients.

Coeliac Marsh Classification

Dr. Michael Marsh introduced the classification system in 1992 to describe the stages of damage in the small intestine as seen under a microscope

Stage 0 - Normal small intestine lined with healthy villi.

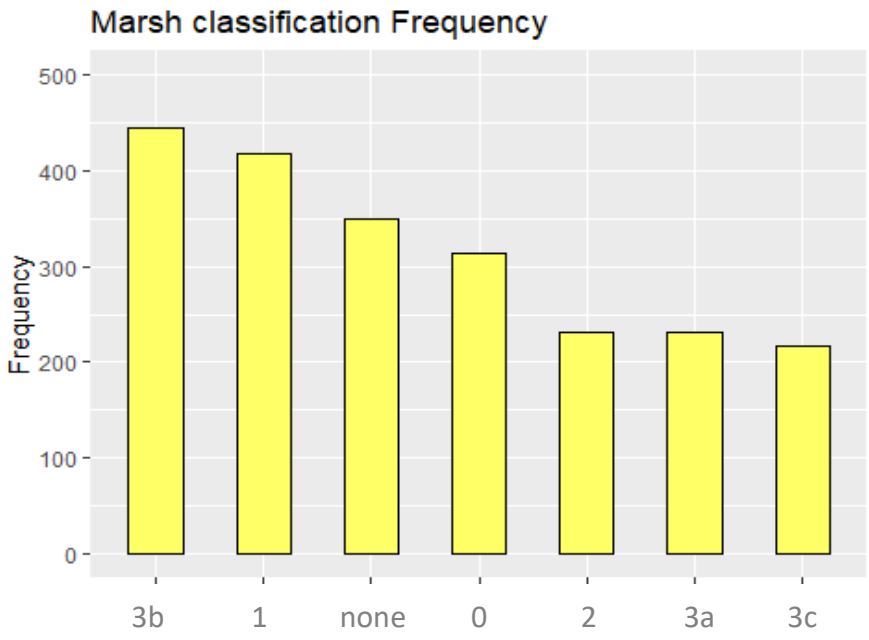
Stage 1 - The cells on the surface of the intestinal lining, known as the epithelial cells, have more lymphocytes among them than normal. The presence of too many lymphocytes indicates inflammation and the potential for damage.

Stage 2 - More lymphocytes than normal as well as bigger depressions than normal between the intestinal villi. These depressions are called “crypts,” and larger-than-normal crypts are called “hyperplastic.

Stage 3 - Most healthcare providers won’t diagnose celiac disease until the intestinal lining Marsh Score reaches this stage. There are three substages: Partial villous atrophy (3a), Subtotal villous atrophy (3b), Total villous atrophy (3c)

Let’s see the distribution of these types in the population:

```
ggplot(data=data) +
 geom_bar(aes(x=reorder(Marsh,Marsh, function(x)-length(x))),
       color='black', fill='#FFFF66', width = 0.5) +
 labs(title="Marsh classification Frequency") +
 xlab("Marsh Types") + ylab("Frequency") +
 expand_limits(y=c(0, 500))
```



Most of the people were assigned with Marsh Types 3b and 1.

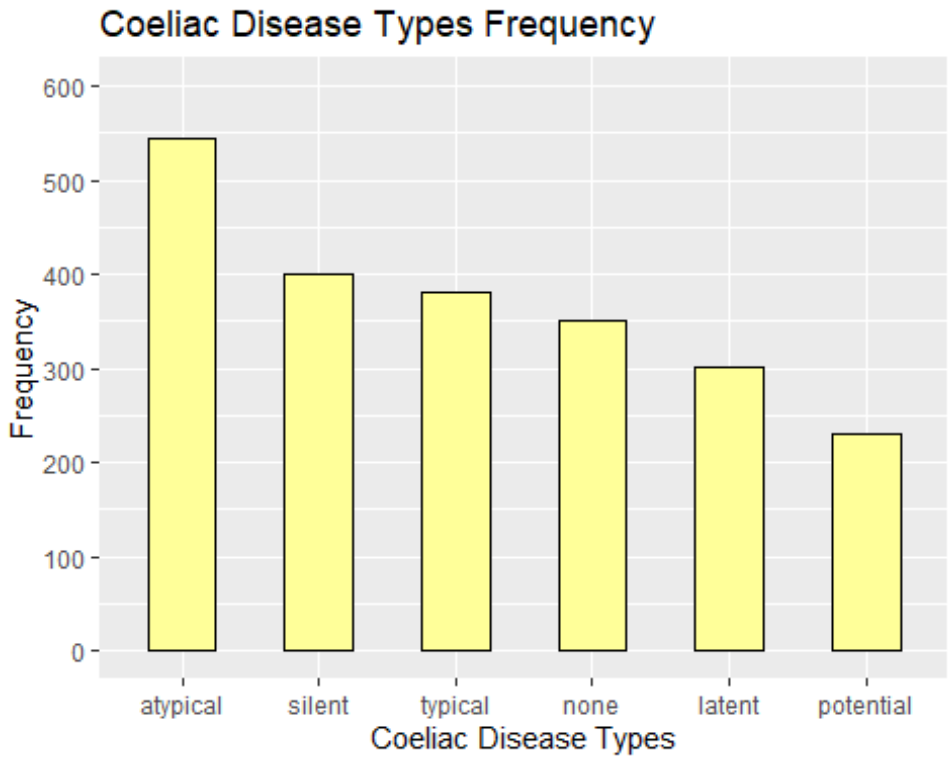
Considering the total of all the marsh types 3 (3a, 3b, 3c), we notice that marsh type 2 is the less frequent one, it is confirmed in literature that type 2 is very rare, seen occasionally in dermatitis herpetiformis. Marsh type 0 means celiac disease highly unlikely with normal intestinal villi, but a person may be coeliac even with these results, other factors may be considered such as symptoms and blood anaylsis. None was included as well, it means not diagnosed with coeliac at all.

Coeliac Disease Types

Let's list the different **types of coeliac disease** (included in the dataset):

- **Potential,** defined by the presence of positive serum antibodies, HLA-DQ2/DQ8 haplotypes, and a normal small intestinal mucosa (Marsh grade 0-1)
- **Latent,** is diagnosed in patients who carry the genes for the disorder but haven't yet developed symptoms. They're on a gluten-based diet and the endoscopic biopsy shows no atrophy in their intestinal villi
- **Typical,** classic or symptomatic with intestinal symptoms and signs
- **Silent,** is also known as asymptomatic celiac disease. Patients do not complain of any symptoms, but still experience villous atrophy damage to their small intestine
- Atypical, with extraintestinal or atypical symptoms and signs

```
ggplot(data=data) +
 geom_bar(aes(x=reorder(cd_type,cd_type,function(x)-length(x))),
       color='black', fill='#FFFF99', width = 0.5) +
 labs(title="Coeliac Disease Types Frequency") +
 xlab("Coeliac Disease Types") + ylab("Frequency") +
 expand_limits(y=c(0, 600)) +
 scale_y_continuous(breaks = c(0,100,200,300,400,500,600))
```



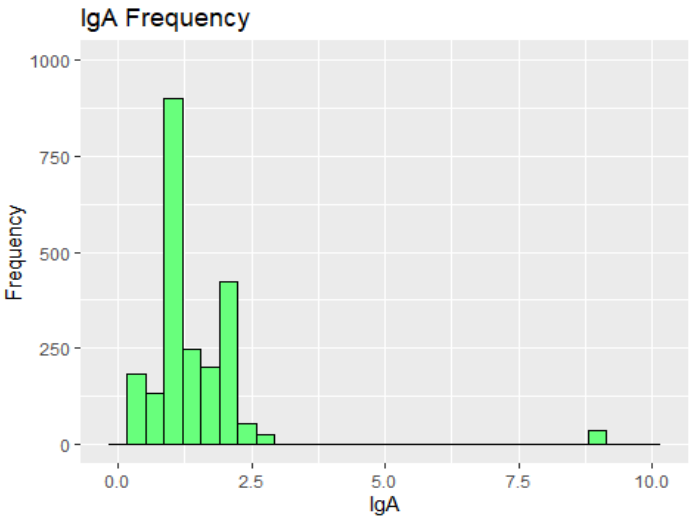
It seems that atypical is the most common type of coeliac disease, followed by silent and typical (standard type). Potential is the less frequent type. In literature it is said that non-classic symptoms are more common in coeliac disease, so the frequency seems as we expected.

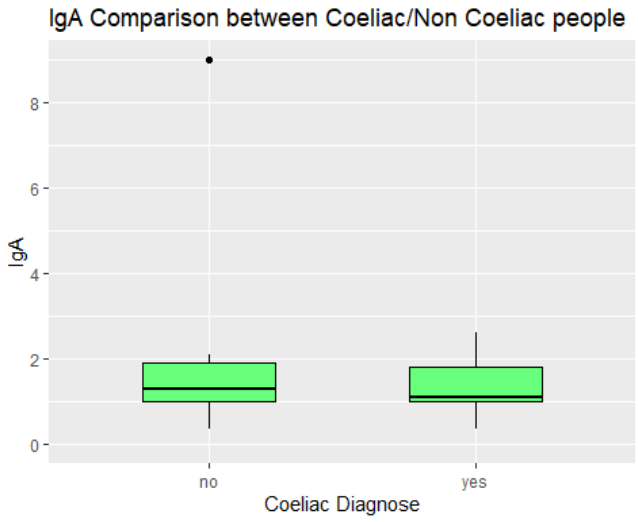
IgA

Immunoglobulin A (IgA) is usually ordered with the tTG-IgA test to detect **IgA deficiency**, which occurs in about 2-3% of people with celiac disease. For accurate diagnostic test results, a patient must be eating a diet that contains gluten.
Negative: less than 4.0 U/mL Weak Positive: between 4.0 and 10.0 U/mL Positive: greater than 10.0 U/mL
Let's see the frequency of values by plotting an histogram:

```
ggplot(data=data) +
 geom_histogram(aes(x=IgA), width=0.3, color="black", fill='#68FF7C', bins=30) +
 labs(title="IgA Frequency") +
 xlab("IgA") + ylab("Frequency") +
 expand_limits(x=c(0,10), y=c(0, 1000))
```

```
ggplot(data=data, aes(y=IgA, x=Disease_Diagnose)) +
 geom_boxplot(color="black", fill='#68FF7C', width=0.5) +
 labs(title="IgA Comparison between Coeliac/Non Coeliac people") +
 xlab("Coeliac Diagnose") + ylab("IgA") +
 expand_limits(y=c(0,8)) +
 scale_y_continuous(breaks = c(0,2,4,6,8))
```





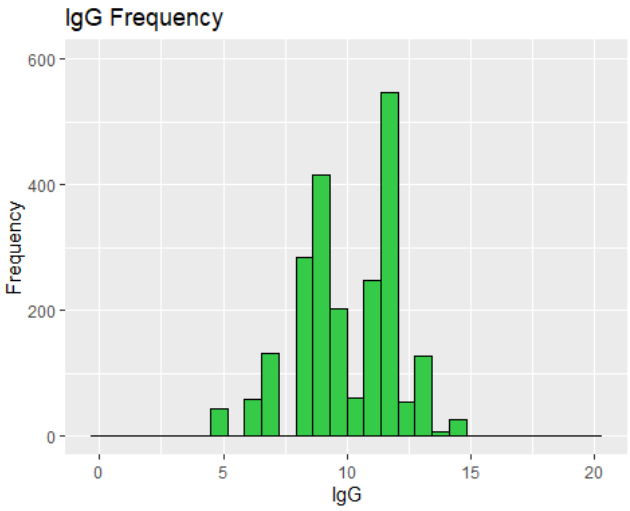
We can see that most of the values are distributed between (more than) 0.0 and 2.5, with outliers greater than 8.0. Assuming in this dataset that the values are measured in U/mL, most of the people resulted negative in the test, that can suggest us that maybe people were already on a gluten diet when the values were taken. We could consider to not use this as a possible indipendent variable, but we aren't sure that the values were actually measured U/mL, it is just and assumption. In the boxplot we can see that outliers in the non coeliac diagnosed people are reducing the height of the two boxes, apart from that the distribution seems similar, but the median is a little lower in coeliac diagnosed case

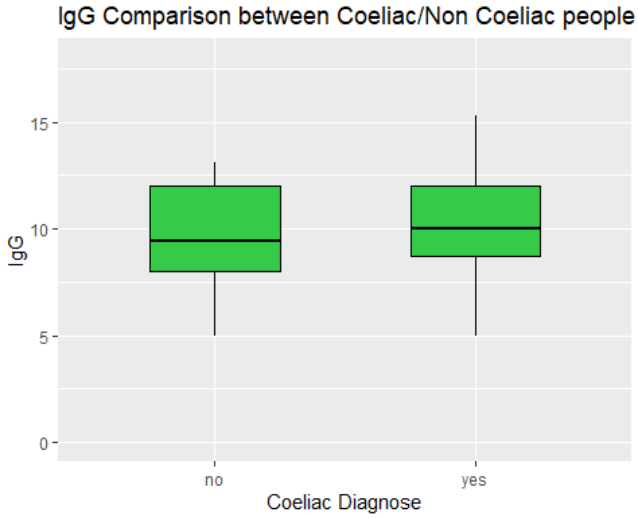
IgG

The serologic tests that check for IgA antibodies are more sensitive for celiac disease than the tests for **IgG** antibodies. However, health care professionals may order DGP tests in certain circumstances. For example, because tTG and EMA tests may be less sensitive in infants and young children, some experts recommend combining the DGP tests with the tTG-IgA test in children younger than age 2. Negative: less than 6.0 U/mL Weak Positive: between 6.0 and 9.0 U/mL Positive: greater than 9.0

```
ggplot(data=data) +
 geom_histogram(aes(x=IgG), width=0.3, color="black", fill='#35CB49', bins=30) +
 labs(title="IgG Frequency") +
 xlab("IgG") + ylab("Frequency") +
 expand_limits(x=c(0,20), y=c(0, 600))
```

```
ggplot(data=data, aes(y=IgG, x=Disease_Diagnose)) +
 geom_boxplot(color="black", fill='#35CB49', width=0.5) +
 labs(title="IgG Comparison between Coeliac/Non Coeliac people") +
 xlab("Coeliac Diagnose") + ylab("IgG") +
 expand_limits(y=c(0,18))
```





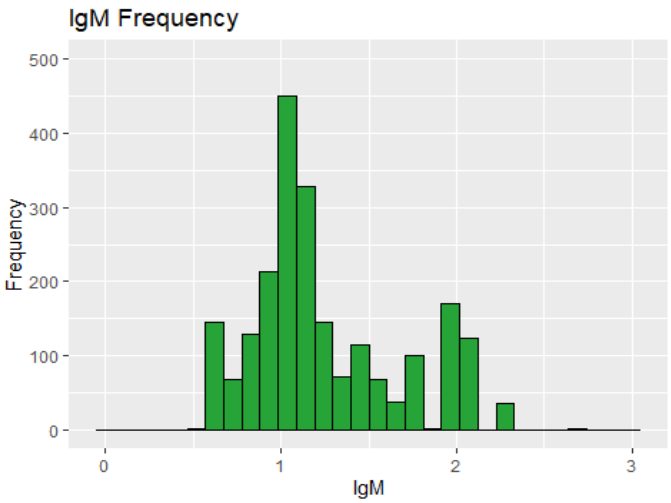
Again making the assumption that the values were measured in U/mL, we can conclude that most of the poeple are weak positive or directly positive, just few cases are negative. The comparison between frequencies for coeliac and non coealic diagnosed people seems not to helpful like in the previous case, because the boxes are similar, the median is a little higher for non-coeliac diagnosed people.

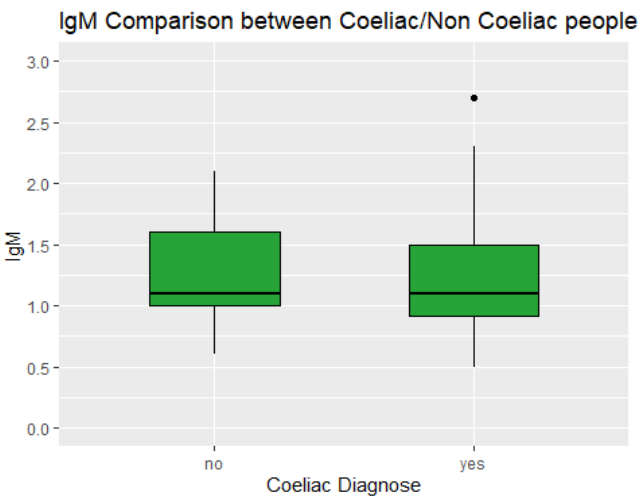
IgM

Selective IgM deficiency (sIGMD) is a very rare disease (less than 300 cases described in literature) and is defined by low levels of IgM (< 0.40 g/L according to current guidelines). It may be associated with coeliac disease. Let's plot the histogram of frequencies again:

```
ggplot(data=data) +
 geom_histogram(aes(x=IgM), width=0.3, color="black", fill='#27A438', bins=30) +
 labs(title="IgM Frequency") +
 xlab("IgM") + ylab("Frequency") +
 expand_limits(x=c(0,3), y=c(0, 500))
```

```
ggplot(data=data, aes(y=IgM, x=Disease_Diagnose)) +
 geom_boxplot(color="black", fill='#27A438', width=0.5) +
 labs(title="IgM Comparison between Coeliac/Non Coeliac people") +
 xlab("Coeliac Diagnose") + ylab("IgM") +
 expand_limits(y=c(0,3)) +
 scale_y_continuous(breaks = c(0,0.5,1,1.5,2,2.5,3))
```



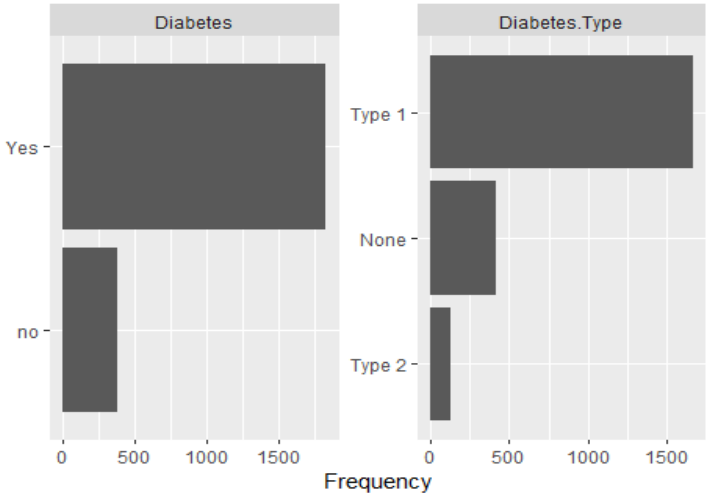


Again we don't know for sure the unit of measure, assuming that it is actually g/L (current guidelines), all of the patients are associated with a value greater than 0.40, so can't be diagnosed with selective IgM deficiency.

Coealiac Disease vs Diabetes

It is know in literature that **Diabetes Type 1** and **Coealiac Disease** are **correlated** in about 10% of cases. Let's try to analyze the possible correlation between these two diseases. First of all, let's plot the frequency of diabetes patients (types as well):

```
plot_bar(data[4:5])
```



Most of them are diagnosed with **Diabetes Type 1**. Let's explore the correlation between this disease (Diabetes type 1/2) and Coeliac disease with a simple contingency table:

```
tab <- table(dplyr::rename(dplyr::select(data, Diabetes, Disease_Diagnose), Coeliac = 'Disease_Diagnose'))
addmargins(tab)
chi <- chisq.test(tab)
chi
##        Coeliac
## Diabetes   no  yes  Sum
##      no   301   76  377
##      Yes   62 1767 1829
##      Sum  363 1843 2206
##
##  Pearson's Chi-squared test with Yates' continuity correction
## X-squared = 1323.4, df = 1, p-value < 2.2e-16
```

The contingency table suggests us that most of the people developed coeliac disease and diabetes (1767 cases).

It is usefult to perform a chi square test to identify the possibility of dependency/independency between the two diseases.

From the test, we can see that the p-value is less than the significance level of 0.05, so we can reject the null hypothesis and conclude that the two diseases are not independent. Obviously, with this dataset, it is easy to predict the possibility that a patient is diagnosed with coeliac disease by already knowing that it developed diabates.

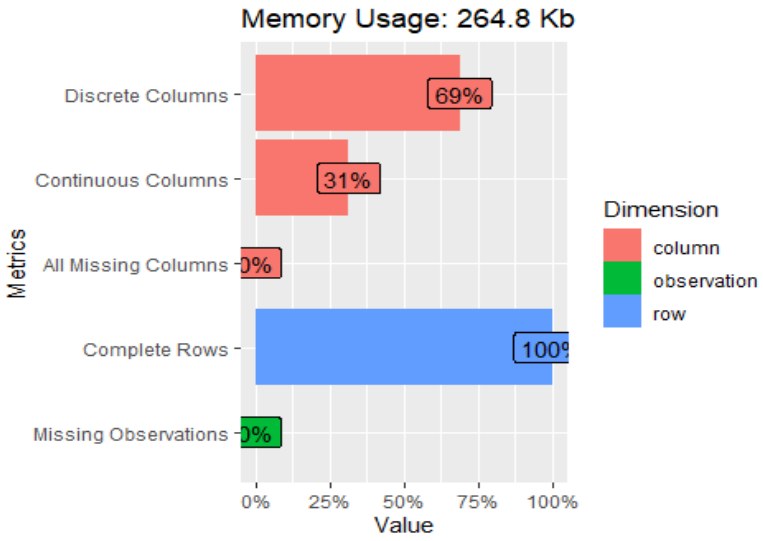
Anyway, considering that in this specific dataset most of the patients presented both diabetes and coeliac disease it doesn't actually mean that in general it is that frequent, in this case around 80%. Again, literature says that about 10% of patients developed both diseases, not 80%, so we can assume that in this specific dataset patients already diagnosed with diabetes were specifically selected. To

understand better the impact of the other independent variables (symptoms and blood analysis), we can assume that we don't actually know if a patient developed diabetes or not.

Missing Data / Possible Mistakes

Final checks, about the possibility of **missing data** and **mistakes** in the dataset.

```
plot_intro(data)
```



We can clearly see from the plot that there are no missing observations in the dataset
Let's try to check some possible mistakes in the dataset.

- If **Disease_Diagnose = None**, than **cd_type** and **Marsh** must be **= None**

```
remove1 <- filter(data, Disease_Diagnose == 'no' & cd_type != 'none' & Marsh != 'none')
remove1
## # A tibble: 13 × 16
##      ID   Age Gender Diabetes `Diabetes Type` Diarrhoea Abdominal Short_Stature
##   <int> <dbl> <chr>  <chr>   <chr>            <chr>     <chr>     <chr>
## 1   107     5 Male   no      None             watery    yes       PSS
## 2   270     5 Male   no      None             inflamma… yes       PSS
....
data <- data[ !(data$ID %in% remove1$ID), ]
```

In these cases, actual coeliac disease types paired with disease diagnose = no makes no sense. We decided to remove these rows from the dataset.

- If **Diabetes = no**, then **Diabetes Type** must be **= None**

```
filter(data, Diabetes == 'no' & `Diabetes Type` != 'None')
## # A tibble: 0 × 16
```

The result is empty, so **no errors for diabetes disease**.
So in conclusion, we removed a total of 13 rows from the dataset, that may be lead to mistakes in the models.

2. Regression Models

Defining Problem

The aim of this study is to diagnostically predict whether or not a patient can be diagnosed with **coeliac disease**, based on certain **diagnostic measurements** included in the dataset.

Type of the Regression Model

We take into account that we need to predict a **binary outcome**, coeliac/non-coeliac, so we need to build a **logistic regression model**

Selection of Variables

A crucial part for this study.
We need to **select the independent variables** that can help us to predict the possibility that a person presents coeliac disease. We can start from an **initial subset** of reasonable independent variables and apply a **stepwise approach** to select the correct subset, then check some possible **confounding effects** of variables as well.

Models

First Logistic Regression Model

The first model will be built considering a starting subset of indipendent variables:

- **Age, Gender, Abdominal, Sticky_Stool, Weight_loss, IgA,IgG**

The subset was chosen considering the **symptoms, blood tests, age and gender of a patient** as possible **risk factors** to develop coeliac disease. After the IDA phase, we can conclude that IgM and Diarrhoea types seems not too helpful for our prediction (maybe be try to use if after..)
First of all, the binary categorical variables will be converted into numbers:

```
data <- mutate(data, Gender = ifelse(Gender == 'Male', 0, 1),
                     Abdominal = ifelse(Abdominal == 'yes', 1, 0),
                     Sticky_Stool = ifelse(Sticky_Stool == 'yes', 1, 0),
                     Weight_loss = ifelse(Weight_loss == 'yes', 1, 0),
                     Disease_Diagnose = ifelse(Disease_Diagnose == 'yes', 1, 0),
                     Diabetes = ifelse(Diabetes == 'Yes', 1, 0))
```

We will split the data in a standard proportion: 70% of rows for training set, 30% of rows for testing

```
train_length = as.integer(0.70*nrow(data))

subset_data <- data[c('Age', 'Gender', 'Abdominal', 'Sticky_Stool', 'Weight_loss', 'IgA',
                      'IgG', 'Disease_Diagnose')]

train_data = subset_data[1:train_length,]
test_data = subset_data[(train_length+1):nrow(subset_data),]
```

After the splitting phase, we can apply the model, with a stepwise approach (as mentioned before) to select the better set of indipendent variables, taking the **lowest AIC**:

```
model1 <- glm(Disease_Diagnose~.,
          family=binomial,
          data=train_data)

step.model1 <- stepAIC(model1, direction = "both", trace = FALSE)
summary(step.model1)
##
## Call:
## glm(formula = Disease_Diagnose ~ Age + Abdominal + Sticky_Stool +
##     Weight_loss + IgA, family = binomial, data = train_data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.46453    0.25785  -1.802   0.0716 .
## Age           0.05293    0.02116   2.501   0.0124 *
## Abdominal     2.11245    0.18913  11.170  < 2e-16 ***
## Sticky_Stool  1.56480    0.24485   6.391 1.65e-10 ***
## Weight_loss   0.35606    0.23288   1.529   0.1263
## IgA          -0.66556    0.08082  -8.236  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1282.88  on 1534  degrees of freedom
## Residual deviance:  829.51  on 1529  degrees of freedom
## AIC: 841.51
##
## Number of Fisher Scoring iterations: 6
```

We can notice that **IgG** and **Gender** varibales **were removed** from the model after the stepwise approach, going from 7 indipendent variables to 5. The **intercept** and **Weight loss** don't seem **statistical significant**, in comparison to the remaining variables that seem more significant, particulary the presence of **Abdominal pain** and **Sticky Stool**, and **IgA value** as well

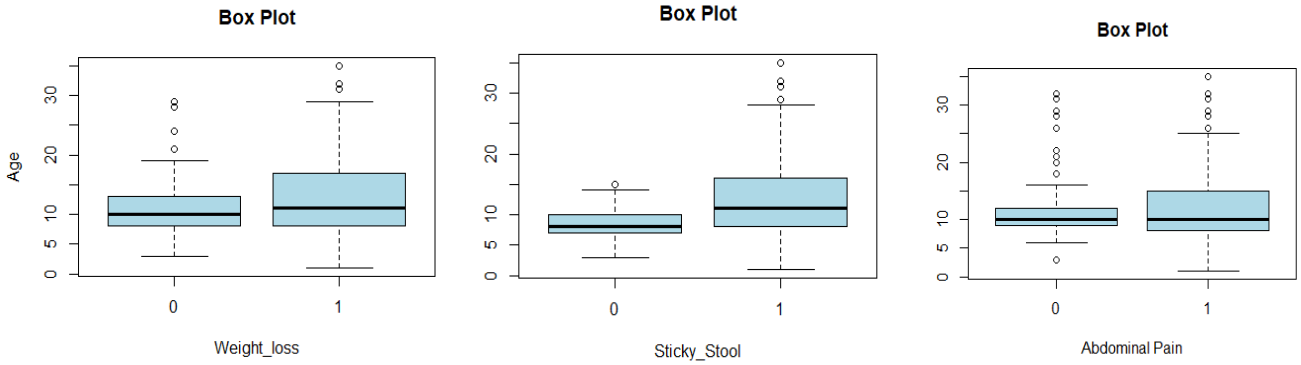
It is important to analyze the possibility of the presence of **confouding effect i**n some variables. The p values of Age, Abdominal, Sticky_Stool and IgA are less than 0.05, so they're statistically significant, but are they correlated with the dependent variable and the other independent variables? Let's see some possible correlation between independent variables:

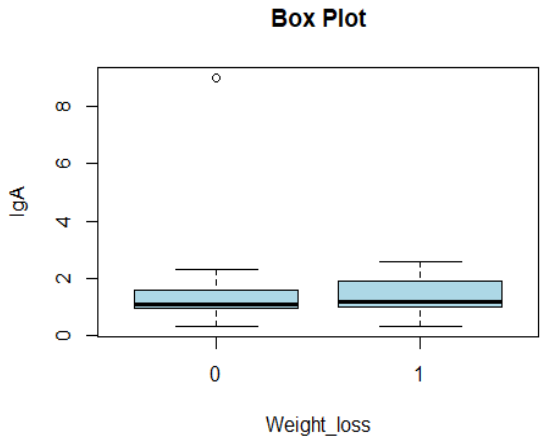
```
#Age
boxplot(Age ~ Abdominal, data = train_data, col = "lightblue", main = "Box Plot", xlab = "Abdominal Pain", ylab = "Age")
boxplot(Age ~ Sticky_Stool, data = train_data, col = "lightblue", main = "Box Plot", xlab = "Sticky_Stool", ylab = "Age")
boxplot(Age ~ Weight_loss, data = train_data, col = "lightblue", main = "Box Plot", xlab = "Weight_loss", ylab = "Age")
ggplot(train_data, aes(x = Age, y = IgA)) +
  geom_point() +
  labs(title = "Scatter Plot", x = "Age", y = "IgA")
```

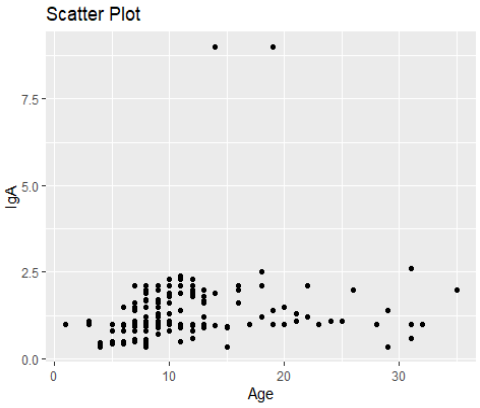
```
#IgA
boxplot(IgA ~ Abdominal, data = train_data, col = "lightblue", main = "Box Plot", xlab = "Abdominal Pain", ylab = "IgA")
boxplot(IgA ~ Sticky_Stool, data = train_data, col = "lightblue", main = "Box Plot", xlab = "Sticky_Stool", ylab = "IgA")
boxplot(IgA ~ Weight_loss, data = train_data, col = "lightblue", main = "Box Plot", xlab = "Weight_loss", ylab = "IgA")
```

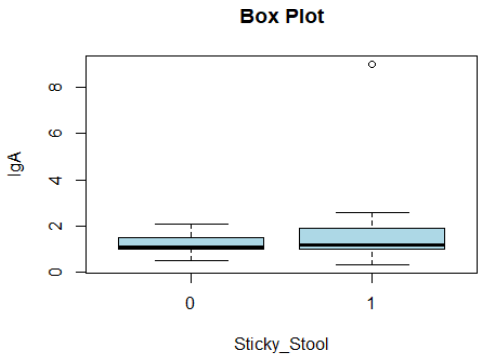
```
#Abdominal Pain
tabaw <- table(dplyr::rename(dplyr::select(train_data, Abdominal, Weight_loss)))
addmargins(tabaw)
chisq.test(tabaw)
tabas <- table(dplyr::rename(dplyr::select(train_data, Abdominal, Sticky_Stool)))
addmargins(tabas)
chisq.test(tabas)
#Sticky Stool
tabws <- table(dplyr::rename(dplyr::select(train_data, Weight_loss, Sticky_Stool)))
addmargins(tabws)
chisq.test(tabws)
##         Weight_loss
## Abdominal    0    1  Sum
##       0    158  147  305
##       1    322  908 1230
##       Sum  480 1055 1535
##
##  Pearson's Chi-squared test with Yates' continuity correction
## X-squared = 73.479, df = 1, p-value < 2.2e-16
##         Sticky_Stool
## Abdominal    0    1  Sum
##       0    141  164  305
##       1    117 1113 1230
##       Sum  258 1277 1535
```

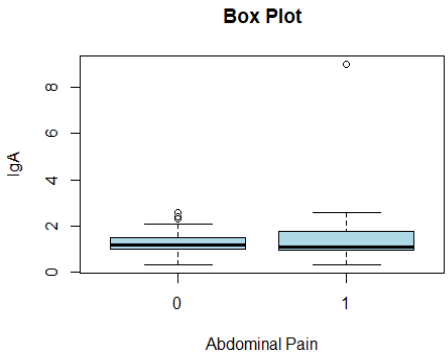
```
##
##  Pearson's Chi-squared test with Yates' continuity correction
## X-squared = 233.02, df = 1, p-value < 2.2e-16
##          Sticky_Stool
## Weight_loss    0    1  Sum
##        0    220  260  480
##        1     38 1017 1055
##        Sum  258 1277 1535
##
##  Pearson's Chi-squared test with Yates' continuity correction
## X-squared = 417.77, df = 1, p-value < 2.2e-16
```











Looking at plots and tables, we can conclude that:

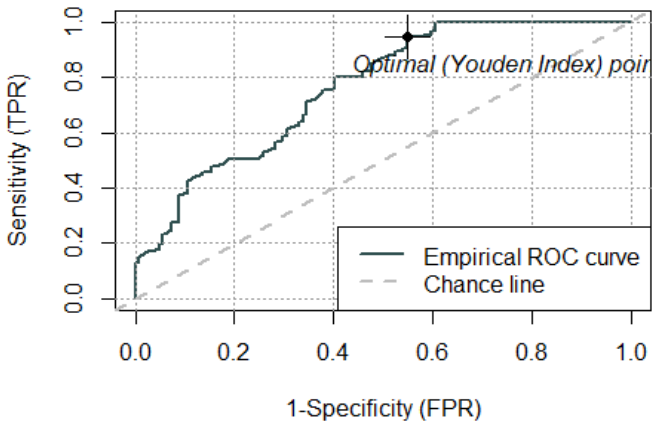
- **Age**: seems that older people developed more symptoms of **Abdominal pain**, **Weight loss** and **Sticky Stool** (so there's a correlation)
- **IgA**: it is not correlated with **Age**, but higher values of it are correlated with presence of **Sticky Stool**, **Abdominal pain** and **Weight loss.**
- **Abdominal pain**, **Weight loss** and **Sticky Stool** seem correlated between each other, that makes sense because it is common that a person may present all these symptoms at the same time.

It is a possibility to consider the removal of an indipendent variable, now it seems right to **remove weight loss** because of a less statistical significance, by the way we have to remember that both sticky stool and abdominal pain are important symptoms of coeliac disease.
We can leave them both for now, taking into account the confounding effect of them in the models.

Let's compute the metricts to assess **Discrimination** of the first model defined:

```
#testing
expected <- test_data$Disease_Diagnose
testProbabilities <- predict(step.model1, newdata = test_data, type = "response")
testPredictions <- ifelse(testProbabilities >= 0.5, 1, 0)
cm <- confusionMatrix(factor(testPredictions), factor(expected), mode = "everything", positive = "1")
cm
ROCit_obj <- rocit(score=testProbabilities,class=expected)
plot(ROCit_obj)
ciAUC(ROCit_obj)
# cross validation for overfitting check
ctrl <- trainControl(method = "cv", number = 10)
cv_model <- train(Disease_Diagnose ~ Abdominal + Age + Sticky_Stool + Weight_loss + IgA, data = train_data,
method = "glm", family = binomial, trControl = ctrl)
print(cv_model)

## Confusion Matrix and Statistics
##
##          Reference
## Prediction   0   1
##          0  49   9
##          1  75 525
##
##                Accuracy : 0.8723
##                  95% CI : (0.8444, 0.8969)
##     No Information Rate : 0.8116
##     P-Value [Acc > NIR] : 1.927e-05
##
##                   Kappa : 0.4755
##
##  Mcnemar's Test P-Value : 1.321e-12
##
```

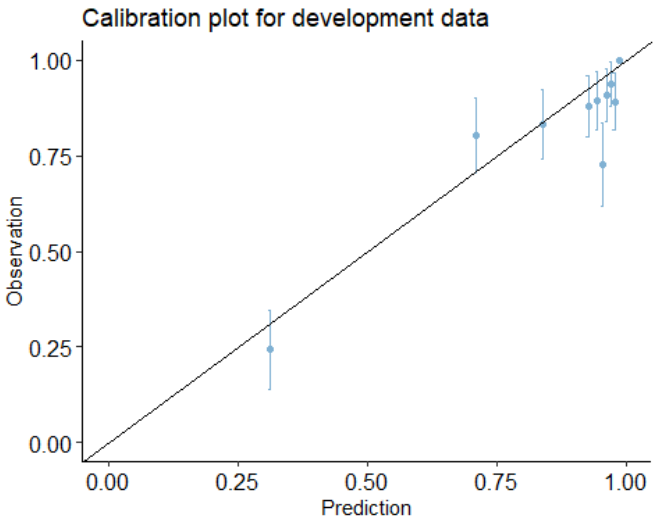
```
##            Sensitivity : 0.9831
##            Specificity : 0.3952
##         Pos Pred Value : 0.8750
##         Neg Pred Value : 0.8448
##             Precision : 0.8750
##                Recall : 0.9831
##                    F1 : 0.9259
##            Prevalence : 0.8116
##         Detection Rate : 0.7979
##   Detection Prevalence : 0.9119
##      Balanced Accuracy : 0.6892
##
##       'Positive' Class : 1
##
##
##   estimated AUC : 0.763622085296605
##   AUC estimation method : empirical
##
##   CI of AUC
##   confidence level = 95%
##   lower = 0.723183765448637     upper = 0.804060405144573

## Generalized Linear Model
##
## 1535 samples
##    5 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1381, 1381, 1381, 1382, 1382, 1381, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   0.2687541  0.4274938  0.1520166
```



Accuracy is around **87.2%**, it is a pretty good result. Of course we need to take into account the fact that the dataset is inbalanced, so most of the people were diagnosed as coeliac, and the accuracy can't be the only valid measure.
The **True Positive Rate** is pretty high, around **98%**, but, as we expected, the model is classifying as positives too many observations because the **True Negative Rate** is around **39%**, and the main reason is because the dataset was inbalanced from the beginning. **AUC** of **0,76** is a good result, **F1 score** is **0.92**, but the **Balanced Accuracy** that gives the same weight between positives and negatives is **0.68**, which is at least greater than 0.5 but we could try to do better than that.
The **ROC Curve** shows a not bad result, with the curve that is a litte far away from the up-left corner. **Overfitting was checked** as well with **cross validation** with **10 folds**, **RMSE** and **MAE** returned good results (both **low**) we need to try other models to see if and **R squared** of **0.42** could be acceptable, right now the model doesn't explain very well the variability, since the value is far from 1. Of course the interpretation of r squared is different from a linear model because it is a logistic one.

Let's assess the **Calibration** of the model

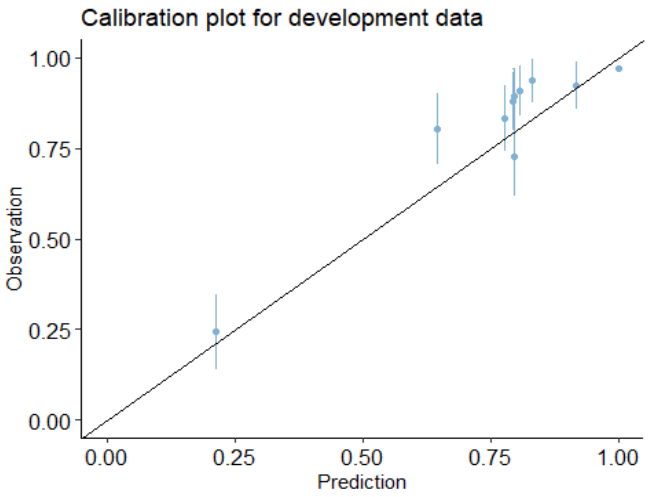
```
logitgof(train_data$Disease_Diagnose,fitted(step.model1))
##
##  Hosmer and Lemeshow test (binary model)
## data:  train_data$Disease_Diagnose, fitted(step.model1)
## X-squared = 193.75, df = 8, p-value < 2.2e-16
notcal <- data.frame(y = test_data$Disease_Diagnose, prob = testProbabilities)

calibration_plot(data = notcal, obs = "y", pred = "prob", title = "Calibration plot for development data", y_lim =
c(0, 1), x_lim=c(0, 1))
## $calibration_plot
```



The **Hosmer and Lemeshow test** returned a **very low p-value** (lesst than 2.2e-16) that suggest us that the model is **not well calibrated**, which is something that we could expect from an inbalanced dataset. The **calibration plot** doesn't look very bad though, but there are different points far away from the line in the middle that represents the perfect calibrated probabilities.
Of course we can expect an **overestimation of the positives**, considering we're working with an higher proportion of coeliac diagnosed patients. The distribution isn't in a sigmoid shape so we

shouldn't apply Platt Scaling as a possible calibration technique, we can give it a try with **Isotonic Regression** that can deal better with different shapes.

```
#remember the expected:
#expected <- test_data$Disease_Diagnose

#calibration
new_pr <- probability.calibration(test_data$Disease_Diagnose, testProbabilities, regularization = TRUE)

cal <- data.frame(y = test_data$Disease_Diagnose, prob = new_pr)

calibration_plot(data = cal, obs = "y", pred = "prob", title = "Calibration plot for development data", y_lim = c(0, 1), x_lim=c(0, 1))
## $calibration_plot
```



We can see a little transformation in the plot as a result to try to calibrate the model, but the points seems a little far away from the middle line again,
The new metrics of the calibrated model were computed but not included there because they're pretty much the same as before, so the calibration didn't help much to improve Discrimination.

Logistic Regression Model with more indipendent variables

Recall the fact that we started with this subset of variables:

- **Age, Gender, Abdominal, Sticky_Stool, Weight_loss, IgA,IgG**

We have to remember that there are other variables that could help us better understand some patterns between coeliac and non-coeliac people. **Short_Stature** and **Diarrhoea** variables contain 3 categories, but **one-hot encoding technique** can be used to encode the variables correctly, creating new columns:

```
shortstature_col <- as.data.frame(model.matrix(~ Short_Stature - 1, data = data))
diarrhoea_col <- as.data.frame(model.matrix(~ Diarrhoea - 1, data = data))

data <- cbind(data, shortstature_col)
data <- cbind(data, diarrhoea_col)
```

After the enconding, we can reason about **IgM** variable. It is not sure if it could actually help us, it is a possibility that may indicate a coeliac person, so we can try to include it as well.
This can be the new subset:

- **Age, Gender, Abdominal, Sticky_Stool, Weight_loss, IgA,IgG, IgM, Diarrhoea(3 variables), Short_Stature(3 variables)**

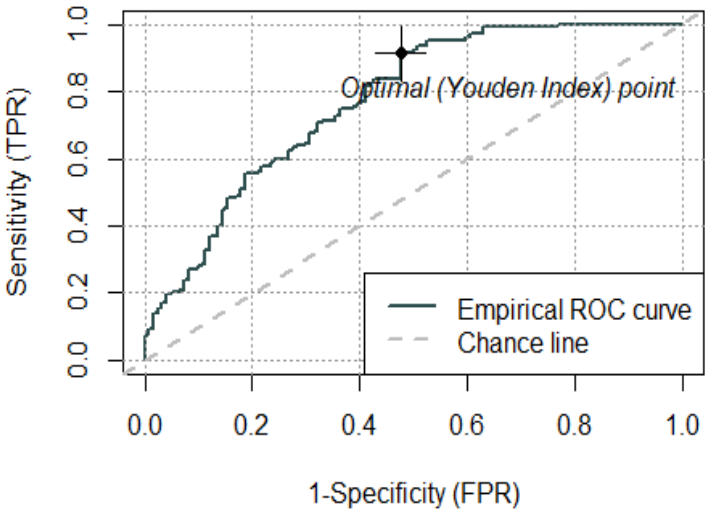
```
train_length = as.integer(0.70*nrow(data))

subset_data2 <- data[c('Age', 'Gender', 'Abdominal', 'Sticky_Stool', 'Weight_loss', 'IgA',
                       'IgG', 'IgM', 'Disease_Diagnose', 'Diarrhoeafatty', 'Diarrhoeainflammatory',
                       'Diarrhoeawatery', 'Short_StatureDSS', 'Short_StaturePSS',
                       'Short_StatureVariant')]

train_data2 = subset_data2[1:train_length,]
test_data2 = subset_data2[(train_length+1):nrow(subset_data2),]

model2 <- glm(Disease_Diagnose~.,
              family=binomial,
              data=train_data2)


step.model2 <- stepAIC(model2, direction = "both", trace = FALSE)
summary(step.model2)
```

```
##
## Call:
## glm(formula = Disease_Diagnose ~ Age + Abdominal + Sticky_Stool +
##     IgA + IgG + IgM + Short_StatureDSS + Short_StaturePSS, family = binomial,
##     data = train_data2)
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -1.78029    0.69953  -2.545  0.01093 *
## Age              0.05712    0.02191   2.607  0.00913 **
## Abdominal        2.09211    0.23264   8.993  < 2e-16 ***
## Sticky_Stool     2.19690    0.24935   8.810  < 2e-16 ***
## IgA             -0.64887    0.08785  -7.387 1.51e-13 ***
## IgG             -0.09162    0.05725  -1.600  0.10953
## IgM              1.16773    0.25844   4.518 6.23e-06 ***
## Short_StatureDSS -0.49024    0.24048  -2.039  0.04149 *
## Short_StaturePSS  2.29550    0.31788   7.221 5.15e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1282.88  on 1534  degrees of freedom
## Residual deviance:  701.89  on 1526  degrees of freedom
## AIC: 719.89
##
## Number of Fisher Scoring iterations: 7
```

The model with the lowest AIC chosen (with stepwise) is built with this subset of variables:
Age, Abdominal, Sticky_Stool, IgA, IgG, IgM, Short_StatureDSS, Short_StaturePSS
The intresting thing is that **IgM** and **Short Stature type PSS** came out **statistically significant**, while **Weight loss**, **Gender** (again) and **Diarrhoea types** were removed. We noticed before that they were not very helpful in distinguish between coeliac and non-coeliac diagnosed people.
IgG is present but it seems **not statistically significant**.
Let's test the discrimination of the model:

```
expected <- test_data2$Disease_Diagnose

testProbabilities <- predict(step.model2, newdata = test_data2, type = "response")
testPredictions <- ifelse(testProbabilities >= 0.5, 1, 0)

cm <- confusionMatrix(factor(testPredictions), factor(expected), mode = "everything", positive = "1")
cm
ROCit_obj <- rocit(score=testProbabilities,class=expected)
plot(ROCit_obj)
ciAUC(ROCit_obj)
```

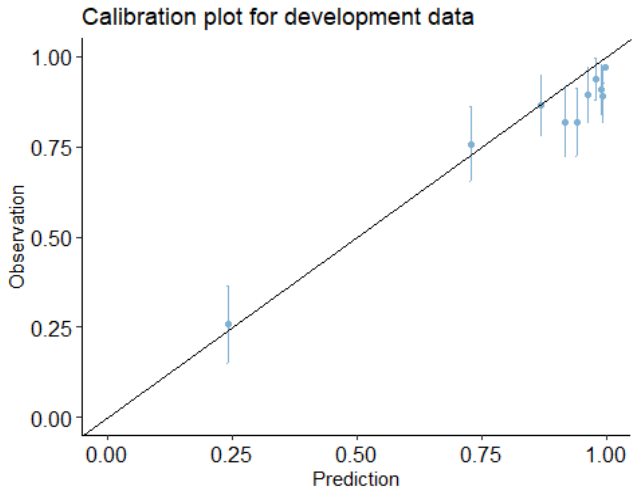
```r
# cross validation for overfitting check
ctrl <- trainControl(method = "cv", number = 10)
cv_model <- train(Disease_Diagnose ~ Age + Abdominal + Sticky_Stool + IgA  + IgG + IgM + Short_StatureDSS +
Short_StaturePSS, data = train_data2, method = "glm", family = binomial, trControl = ctrl)
print(cv_model)

## Confusion Matrix and Statistics
##          Reference
## Prediction   0   1
##         0  46  13
##         1  78 521
##
##               Accuracy : 0.8617
##                 95% CI : (0.8329, 0.8872)
##    No Information Rate : 0.8116
##    P-Value [Acc > NIR] : 0.0003998
##
##                  Kappa : 0.4339
##
##  Mcnemar's Test P-Value : 1.959e-11
##
##             Sensitivity : 0.9757
##             Specificity : 0.3710
##          Pos Pred Value : 0.8698
##          Neg Pred Value : 0.7797
##               Precision : 0.8698
##                  Recall : 0.9757
##                      F1 : 0.9197
##              Prevalence : 0.8116
##          Detection Rate : 0.7918
##    Detection Prevalence : 0.9103
##       Balanced Accuracy : 0.6733
##
##        'Positive' Class : 1
##
##    estimated AUC : 0.771981092183158
##    AUC estimation method : empirical
##
##    CI of AUC
##    confidence level = 95%
##    lower = 0.732381965117996     upper = 0.811580219248321

## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1381, 1381, 1382, 1381, 1382, 1381, ...
## Resampling results:
##   RMSE      Rsquared   MAE
##   0.2450612  0.5132599  0.1270459
```



We don't notice too many changes, we were interested particularly in **True Negative Rate** to see if this model could capture better patterns with the additional variables, but this rate is **0.37** that is even worse than before (around 0.40), **accuracy** is still pretty high, **F1 score** again is high, **balanced accuracy** again around 0.67. **ROC curve** seems a little bit better then before, **AUC** goes from 0.76 to 0.77 (so a very little change).
About the results of cross validations, **RMSE** and **MAE** are low as before, **R squared** is higher than before (more than **0.52**), still far from 1. Of course higher values of R squared are better.

What about the **calibration** of the model?

```
logitgof(train_data2$Disease_Diagnose,fitted(step.model2))
notcal <- data.frame(y = test_data2$Disease_Diagnose, prob = testProbabilities)

calibration_plot(data = notcal, obs = "y", pred = "prob", title = "Calibration plot for development data", y_lim =
c(0, 1), x_lim=c(0, 1))
##
##  Hosmer and Lemeshow test (binary model)
##
## data:  train_data2$Disease_Diagnose, fitted(step.model2)
## X-squared = 65.188, df = 8, p-value = 4.43e-11

## $calibration_plot
```



Again the test returned a very low p value and the probabilities seems a little bit **miscalibrated** (as we expected)

We can conclude that we obtained good models in general, but that showed some difficulties in negatives classification and in generating calibrated probabilities.

An **optimal subset** of **indipedent variables** to be considered could be:

Age, Abdominal, Sticky_Stool, IgA,IgM, Short Stature DSS, Short Stature PSS

So what happened is that weight loss, IgG and Gender were removed because they seemed not statistically significant in the previous models.

Logistic Regression model with Down Sampling

With the **down-sampling technique** we reduce the positive observations (in this case coeliac diagnosed patients)

```
# Count classes
conta_classi <- data %>%
 count(Disease_Diagnose)

min_istanze <- min(conta_classi$n)

# Down-sampling major class
data_downsampled <- data %>%
 group_by(Disease_Diagnose) %>%
 sample_n(min_istanze)

#print new count
conta_classi <- data_downsampled %>%
 count(Disease_Diagnose)

print(conta_classi)
#shuffling
set.seed(123)
random_rows <- sample(nrow(data_downsampled))
data_downsampled <- data_downsampled[random_rows, ]
data_downsampled

## # A tibble: 2 × 2
## # Groups:   Disease_Diagnose [2]
##   Disease_Diagnose     n
##              <dbl> <int>
## 1                0   350
## 2                1   350
```

Now we obtained a situation with exactly **50% of positives** and **50% of negatives**.
Of course the result is a reducing of the number of observatios in the dataset, now it is **700**.
Again, we apply the model (considering the **optimal subset defined before**):

```
train_length_d = as.integer(0.70*nrow(data_downsampled))

subset_data_d <-data_downsampled[c('Age', 'Abdominal', 'Sticky_Stool', 'IgA',
                          'IgM', 'Disease_Diagnose', 'Short_StatureDSS', 'Short_StaturePSS')]

train_data_d = subset_data_d[1:train_length_d,]
test_data_d = subset_data_d[(train_length_d+1):nrow(subset_data_d ),]
```

```
model3 <- glm(Disease_Diagnose~.,
          family=binomial,
          data=train_data_d)

step.model3 <- stepAIC(model3, direction = "both", trace = FALSE)
summary(step.model3)
##
## Call:
## glm(formula = Disease_Diagnose ~ Age + Abdominal + Sticky_Stool +
##     IgA + IgM + Short_StatureDSS + Short_StaturePSS, family = binomial,
##     data = train_data_d)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -3.63738    0.59760  -6.087 1.15e-09 ***
## Age              0.06096    0.02210   2.759 0.005804 **
## Abdominal        1.75213    0.30068   5.827 5.63e-09 ***
## Sticky_Stool     1.34808    0.33852   3.982 6.83e-05 ***
## IgA             -0.54420    0.11921  -4.565 4.99e-06 ***
## IgM              1.16085    0.31630   3.670 0.000242 ***
## Short_StatureDSS -0.68148    0.32788  -2.078 0.037667 *
## Short_StaturePSS  1.12388    0.29070   3.866 0.000111 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 677.80  on 488  degrees of freedom
## Residual deviance: 446.74  on 481  degrees of freedom
## AIC: 462.74
##
## Number of Fisher Scoring iterations: 5
```

We can notice that the best AIC value is obtained with the **entire subset of indipendent variables**, all of them seems statistically significant, a little less for IgM, Short stature types and Sticky Stool (in comparison of previous models), but still p values lower than 0.05.

Let's check the **discriminaton** of the model:

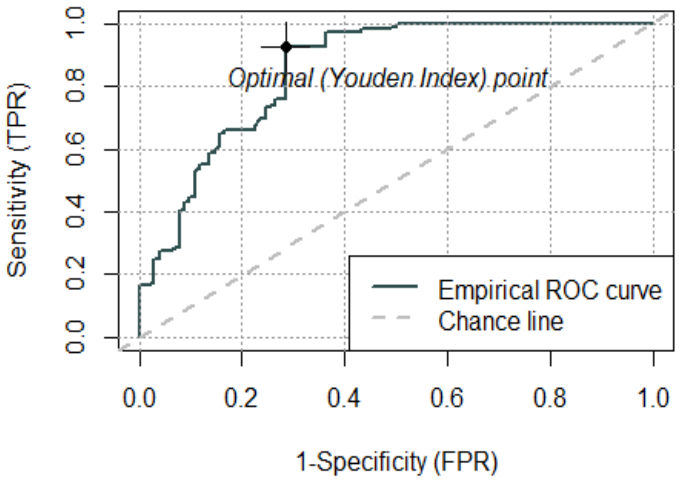
```
expected <- test_data_d$Disease_Diagnose

testProbabilities <- predict(step.model3, newdata = test_data_d, type = "response")
testPredictions <- ifelse(testProbabilities >= 0.5, 1, 0)

cm <- confusionMatrix(factor(testPredictions), factor(expected), mode = "everything", positive = "1")
cm
ROCit_obj <- rocit(score=testProbabilities,class=expected)
plot(ROCit_obj)
```

```r
ciAUC(ROCit_obj)
ctrl <- trainControl(method = "cv", number = 5)
cv_model <- train(Disease_Diagnose ~ Age + Abdominal + Sticky_Stool + IgA + IgM + Short_StatureDSS +
Short_StaturePSS, data = train_data_d, method = "glm", family = binomial, trControl = ctrl)
print(cv_model)
```

```
## Confusion Matrix and Statistics
##
##          Reference
## Prediction  0  1
##         0 73 19
##         1 29 90
##
##               Accuracy : 0.7725
##                 95% CI : (0.7099, 0.8272)
##    No Information Rate : 0.5166
##    P-Value [Acc > NIR] : 1.597e-14
##
##                  Kappa : 0.5431
##
##  Mcnemar's Test P-Value : 0.1939
##
##             Sensitivity : 0.8257
##             Specificity : 0.7157
##          Pos Pred Value : 0.7563
##          Neg Pred Value : 0.7935
##              Precision : 0.7563
##                 Recall : 0.8257
##                     F1 : 0.7895
##             Prevalence : 0.5166
##          Detection Rate : 0.4265
##    Detection Prevalence : 0.5640
##       Balanced Accuracy : 0.7707
##
##       'Positive' Class : 1
##
```

```
##
##    estimated AUC : 0.853705702464472
##    AUC estimation method : empirical
##
##    CI of AUC
##    confidence level = 95%
##    lower = 0.802274657203745     upper = 0.905136747725199
## Generalized Linear Model
##
## 489 samples
##   7 predictor
```

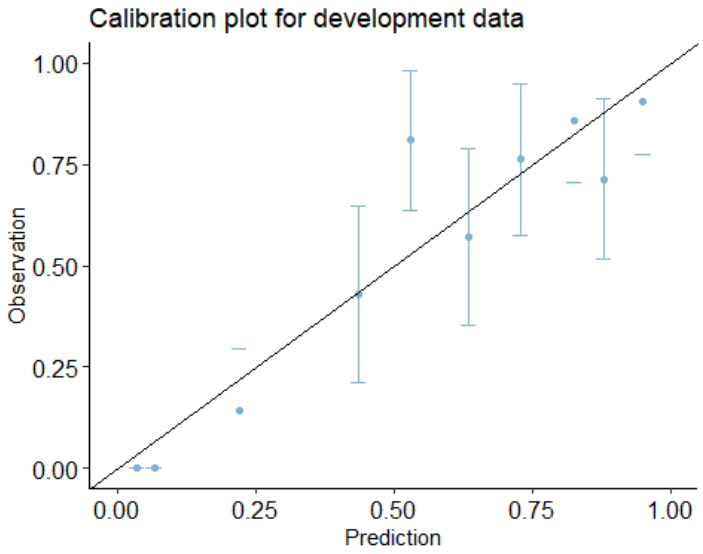
```
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 391, 391, 392, 391, 391
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   0.3928775  0.3958075  0.3025911
```



Of course the result can be less accurate because of the reducing of the size of dataset.
The **accuracy i**s lower (**0.77**), but the model seems to handle the negatives better, with a **specificity** of **0.72**. But, we notice that the **F1 score** is lower than before (**0.78**). **AUC** has increased, reaching almost **0.80**.
We can conclude that in general the model handles the negatives better than before, and positives are handled well anyway (**0.80** is a good result).
RMSE and **MAE** are a little higher than previous models (as we could expect), **R squared** of **0.39** is not a very much a satisfying result, lower than before.
Let's check the **calibration** of the model:

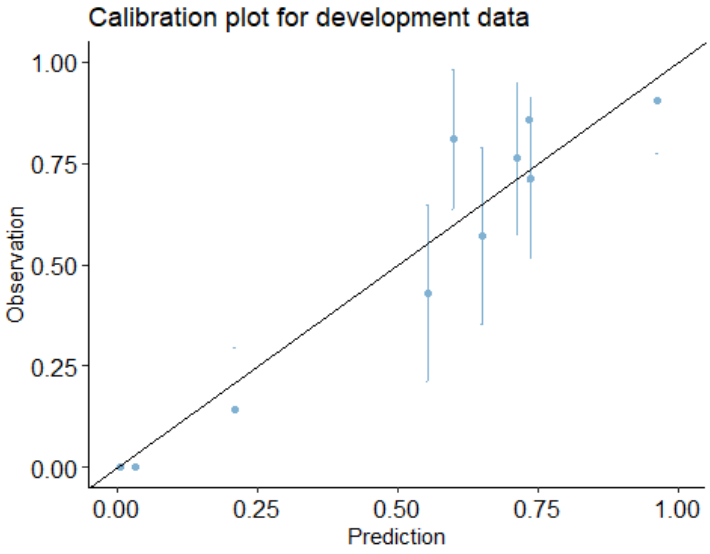
```
logitgof(train_data_d$Disease_Diagnose,fitted(step.model3))
##
##  Hosmer and Lemeshow test (binary model)
##
## data:  train_data_d$Disease_Diagnose, fitted(step.model3)
## X-squared = 38.269, df = 8, p-value = 6.714e-06
notcal <- data.frame(y = test_data_d$Disease_Diagnose, prob = testProbabilities)

calibration_plot(data = notcal, obs = "y", pred = "prob", title = "Calibration plot for development data", y_lim =
c(0, 1), x_lim=c(0, 1))
```

```
## $calibration_plot
```



The test and the plot shows that the calibration is a little better as before, with a p value still low. We still notice **some miscalibration** in the plot, so let's apply Isotonic Regression calibration:

```
#calibration

new_pr <- probability.calibration(test_data_d$Disease_Diagnose, testProbabilities, regularization = TRUE)

cal <- data.frame(y = test_data_d$Disease_Diagnose, prob = new_pr)

calibration_plot(data = cal, obs = "y", pred = "prob", title = "Calibration plot for development data", y_lim = c(0, 1), x_lim=c(0, 1))
## $calibration_plot
```

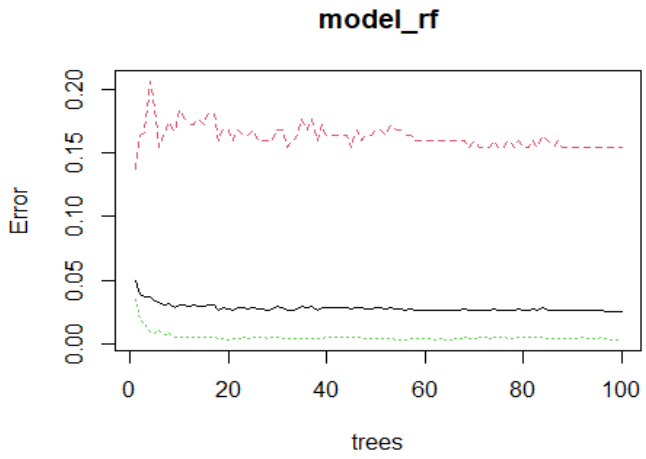


We still notice some **miscalibration i**n the plot and the metrics are not changing in a significant way, that's why they're not included, we just report that **AUC** has reached **0.80** (that is an optimal result).
In general we can consider these models with down sampling data **more balanced between negatives and positives**, but the metrics are pretty good in all the logistic models defined.
It is intresting that we found an optimal set of indipendent variables that can be considered as the main risk factors of coeliac disease in this study.

Random Forest

The models that we tried seems good models after all, but of course the main concern is about the **inbalance** of the dataset, only 350 negative observations out of more than 2000 in total.
The down sampling technique seems to help improving the TNR, but of course we have to consider the fact that we're working with less observations, infact the accuracy is worse.
What about try to use a **different algorithm**? **Random Forest** is an easy powerful alternative, remembering that this kind of algorithms tend to produce miscalibrated probabilities, pushing towards 0 and 1.
Let's see the result of the application on **inbalanced dataset**, with a first try of **100 trees**:

```
model_rf <- randomForest(as.factor(Disease_Diagnose) ~ Age + Abdominal + Sticky_Stool + IgA + IgM +
Short_StatureDSS + Short_StaturePSS, data = train_data2, ntree = 100, type = "classification")

plot(model_rf)
```



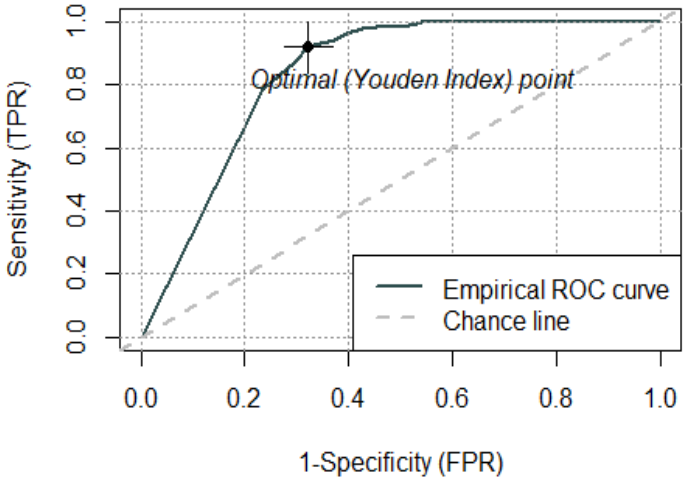
As we expected, we can notice major difficulties in **classifying negatives** (inbalanced dataset), other than that we can observe that after n = 15/20 trees the error is pretty much the same, so we don't need 100 trees. Let's apply the model with 20 trees:

```
model_rf <- randomForest(as.factor(Disease_Diagnose) ~ Age + Abdominal + Sticky_Stool + IgA + IgM +
Short_StatureDSS + Short_StaturePSS, data = train_data2, ntree = 20, type = "classification", cv = 10)

print(model_rf)
cm <- confusionMatrix(testPredictions, as.factor(expected), mode = "everything", positive = "1")
cm
ROCit_obj <- rocit(score=testProbabilities,class=expected)
plot(ROCit_obj)
ciAUC(ROCit_obj)
expected <- test_data2$Disease_Diagnose
testProbabilities <- predict(model_rf, test_data2, type='prob')[,2]
testPredictions <- predict(model_rf, test_data2, type='response')
# we don't need: ifelse(testProbabilities >= 0.5, 1, 0)
```

```
##
## Call:
##  randomForest(formula = as.factor(Disease_Diagnose) ~ Age + Abdominal +      Sticky_Stool + IgA + IgM +
Short_StatureDSS + Short_StaturePSS,      data = train_data2, ntree = 20, type = "classification",      cv = 10)
##               Type of random forest: classification
##                     Number of trees: 20
## No. of variables tried at each split: 2
##
##        OOB estimate of  error rate: 2.93%
## Confusion matrix:
##     0    1 class.error
## 0 186   40  0.17699115
## 1   5 1304  0.00381971
```

```
## Confusion Matrix and Statistics
##
##          Reference
## Prediction   0   1
##         0  57   3
##         1  67 531
##
##               Accuracy : 0.8936
##                 95% CI : (0.8675, 0.9161)
##    No Information Rate : 0.8116
##    P-Value [Acc > NIR] : 6.161e-09
##
##                  Kappa : 0.5663
##
##  Mcnemar's Test P-Value : 5.076e-14
##
##             Sensitivity : 0.9944
##             Specificity : 0.4597
##          Pos Pred Value : 0.8880
##          Neg Pred Value : 0.9500
##               Precision : 0.8880
##                  Recall : 0.9944
##                      F1 : 0.9382
##              Prevalence : 0.8116
##          Detection Rate : 0.8070
##    Detection Prevalence : 0.9088
##       Balanced Accuracy : 0.7270
##
##        'Positive' Class : 1
##
```

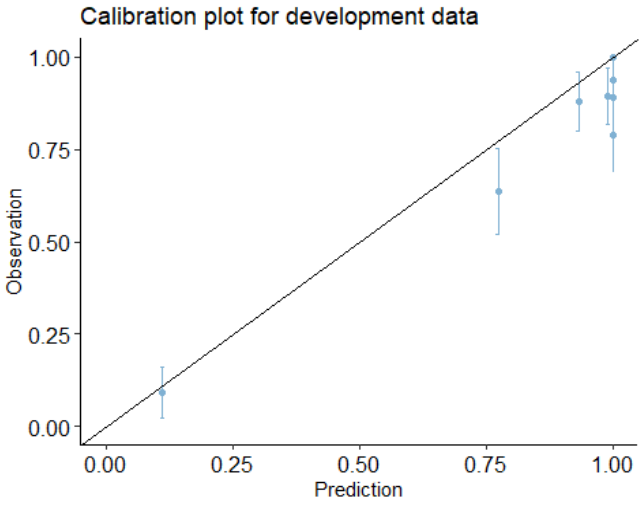
```
##
##   estimated AUC : 0.837599673794853
##   AUC estimation method : empirical
##
##   CI of AUC
##   confidence level = 95%
##   lower = 0.805382143797132     upper = 0.869817203792575
```



The **accuracy** is pretty high, almost **0.90**, **F1 score** is high as well. **Specificity** is **0.46**, like a random classifier, but it is a little better than previous logistic models (with no down sampling), **OOB** is very low (**3%**). In general, the results are better even with an inbalanced dataset, the **balanced accuracy** is higher as well (**0.72**)
But what about the calibration of the probabilities? Classification trees are well known as powerful models but that may lead to miscalibrated probabilities. Let's check it :

```
notcal <- data.frame(y = test_data2$Disease_Diagnose, prob = testProbabilities)

calibration_plot(data = notcal, obs = "y", pred = "prob", title = "Calibration plot for development data", y_lim = c(0, 1), x_lim=c(0, 1))
```

$calibration_plot



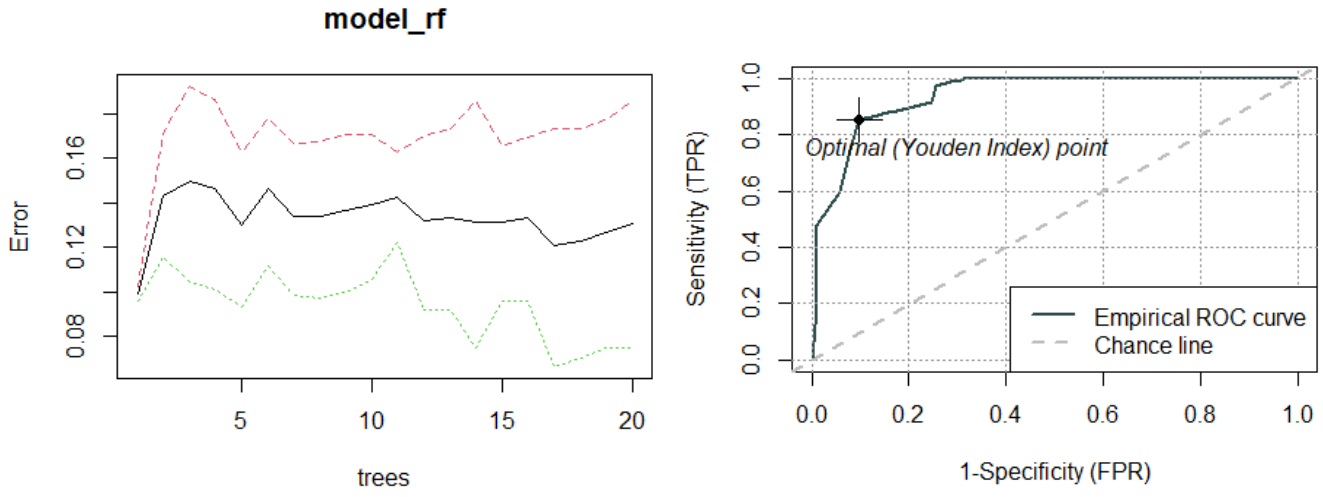
Probabilities seems a little miscalibrated but not that much as the previous logistic models. We could expect worse of course, so misalibration here is not the main concern, but more the Specificy that performs like a random classifier.

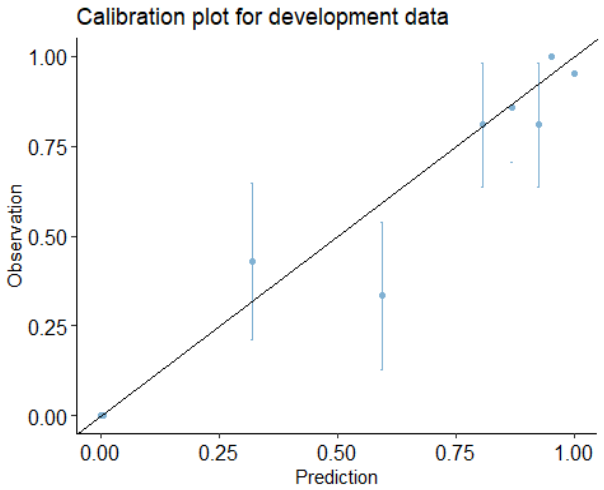
Random Forest Model with Down Sampling

What about applying the random forest algorithm with the previous **down sampled** dataset (50% positives, 50% negatives)? We will keep the number of trees = 20:

```
model_rf <- randomForest(as.factor(Disease_Diagnose) ~ Age + Abdominal + Sticky_Stool + IgA + IgM +
Short_StatureDSS + Short_StaturePSS, data = train_data_d, ntree = 20, type = "classification", cv = 5)

print(model_rf)
plot(model_rf)
expected <- test_data_d$Disease_Diagnose

testProbabilities <- predict(model_rf, test_data_d, type='prob')[,2]
testPredictions <- predict(model_rf, test_data_d, type='response')
# we don't need: ifelse(testProbabilities >= 0.5, 1, 0)

cm <- confusionMatrix(testPredictions, as.factor(expected), mode = "everything", positive = "1")
cm
ROCit_obj <- rocit(score=testProbabilities,class=expected)
plot(ROCit_obj)
ciAUC(ROCit_obj)

notcal <- data.frame(y = expected, prob = testProbabilities)
calibration_plot(data = notcal, obs = "y", pred = "prob", title = "Calibration plot for development data", y_lim =
c(0, 1), x_lim=c(0, 1))

##
## Call:
##  randomForest(formula = as.factor(Disease_Diagnose) ~ Age + Abdominal +      Sticky_Stool + IgA + IgM +
Short_StatureDSS + Short_StaturePSS,      data = train_data_d, ntree = 20, type = "classification",      cv = 5)
##               Type of random forest: classification
##                     Number of trees: 20
## No. of variables tried at each split: 2
##
##        OOB estimate of  error rate: 13.09%
## Confusion matrix:
##     0   1 class.error
## 0 202  46   0.1854839
## 1  18 223   0.0746888
## Confusion Matrix and Statistics
##
##          Reference
## Prediction  0  1
##          0 82 12
##          1 20 97
##
##               Accuracy : 0.8483
##                 95% CI : (0.7927, 0.8939)
```

```
##    No Information Rate : 0.5166
##    P-Value [Acc > NIR] : <2e-16
##
##                  Kappa : 0.6956
##
##  Mcnemar's Test P-Value : 0.2159
##
##             Sensitivity : 0.8899
##             Specificity : 0.8039
##          Pos Pred Value : 0.8291
##          Neg Pred Value : 0.8723
##               Precision : 0.8291
##                  Recall : 0.8899
##                      F1 : 0.8584
##              Prevalence : 0.5166
##          Detection Rate : 0.4597
##    Detection Prevalence : 0.5545
##       Balanced Accuracy : 0.8469
##
##       'Positive' Class : 1
##
##
##    estimated AUC : 0.93978233495233
##    AUC estimation method : empirical
##
##    CI of AUC
##    confidence level = 95%
##    lower = 0.906672135803981     upper = 0.972892534100678
## $calibration_plot
```





This final model, with **Accuracy = 0.84**, **OOB = 0.13 (with cv = 5)**, **balanced accuracy** almost reaching **0.85**, and **AUC** = **0,94**, is probably the **best model** so far, still showing some **miscalibration** though. Remember that with down sampling the dataset consists of less observations (700), then the measures can be less accurate with less observations in test data, but random forest with down sampling worked better with negatives. **F1 score** is lower than before though.
This final model is the one that **discriminates better** between positives and negatives.

Final Conclusions

The aim of this study was to determinate the probability of a person to develop coeliac disease, and possible risk factors associated. These are the final conclusions:

- **Age/Gender correlation**: In literature it is said that coeliac disease is a little bit more frequent in females, as a result in this study knowing the gender didn't seem helpful to predict the coeliac diagnose, so we can conclude that it is not a risk factor. About age, **being younger** can be considered as a **risk factor**, in fact coeliac disease is usually diagnosed in children. Non coeliac diagnosed people seemed a little older in this dataset.
- **Symptoms correlation**: We can consider **Sticky Stool** and **Abdominal Pain** as **some important risk factors**, and they are usually related between each other as well. On the other hand **weight loss** didn't seem too helpful for predicting the presence of coeliac disease, in fact it is a pretty generic symptom that can be associated with other diseases.
  Diarrhoea was present in all of the patients in the dataset, in literature it is said that the types can be helpful for a coeliac diagnose, but they weren't in this case, as they seemed distributed in the three types in a similar way for both classes.
- **Diseases Correlation**: Based on the data collected, **80%** of patients developed both **diabetes and coeliac disease**, so that actually confirmed that the **two diseases** are **correlated**, but in literature it is said that only 10% of patients develop both diseases. That suggests that maybe the data was collected specifically about patients already diagnosed with diabetes.
  We noticed that **short stature** of type **PSS** and **DSS** can be **correlated** with the possibility of a patient to be coeliac diagnosed, and not the **Variant** one.
- **Blood Tests:** Based on the models results, both **IgA** and **IgM values** were helpful to understand the possibility of a patient to be coeliac diagnosed. It wasn't clear the unit of measure in which the values were measured, so it was difficult to interpret the values in the correct way (IDA phase). **IgG** didn't seem too helpful. The more surprising result was actually about IgM: in literature it is not clear if this value could help to diagnose coeliac disease, but in this study it seems that it can be considered as a risk factor to develop the disease.
- **Models Results:** The models returned **pretty good results** in general, of course the **inbalancing between positives and negatives** caused the problem of an over estimation of the positives, worsening the results for **True Negatives Rate**. Random Forest worked better in general than Logistic Regression. Some miscalibration of probabilities was shown in all of the models.