Challenge Based Learning: Diabetes

Pedro Bueso-Inchausti, Ignacio Taguas

2020-1-10

Contents

1 Pre-requisites

We set the working directory.

```
setwd("C:/Users/User/Desktop/KRA")
```

We load the packages that will be needed.

```
options(warn=-1)

#These packages are used for dealing with dataframes

library (dplyr)
library (tidyr)
library (stringr)

#These packages are used for dealing with tables

library(knitr)
library(kableExtra)

#These packages are used for dealing with figures

library(grid)
library(ggplot2)
library(gridExtra)

#These packages are used for the statistical analysis

library (car)
library (VIM)
library (mice)
library (Hmisc)
library (mvnmle)
library(corrplot)
library (MissMech)
library (finalfit)
library (BaylorEdPsych)
library (rriskDistributions)
```

2 Challenge proposal (1st week)

One great problem that exists in data science is that machine learning can make anyone seem like a decent modeler. Let's think about the following situation. John is a software engineer student, so he kind of masters programming. In addition, he knows about several tools for doing data mining and analysis. In one of his many subjects, John is supposed to analyze a dataset. He picks one about diabetes; his grandfather, who died a couple years ago, had such disease. John applies everything he has been learning: imputations for handling data that is missing, feature selection for determining which variables should be considered and 'black boxes' for building models that will classify the patients. He forgets, however, about a very important thing; he barely looks the data and no explanation is given about the specific problem he is facing. After his presentation, the professor looks a bit confused, "So... tell me, what have you learned about diabetes?". John is unable to respond. His analysis, although correct from the technical point of view, could have been applied to any other dataset; and this is, frankly, the worst thing that can happen in a data science problem.

We ourselves don't want to be like John. Due to this, we have planned a workflow where everything is built on top of the content. Our dataset includes patients of type 2 Diabetes Mellitus, so the first thing we do is to get a clear idea of what is diabetes. How is it diagnosed? How is it treated? What can we tell about the pathogeny? What about the epidemiology? And about the symptomatology? The second thing we do is to investigate about the possible relations between our dataset variables and the fact of having type 2 DM. Once all of this is clear, we move to the dataset preparation and analysis. We first search for relationships between the variables and the outcome; we then focus on relations between the variables themselves; eventually, we try to obtain models that provide us with explainability and predictability. What is the goal of this all? Try to find, in our dataset, the same patterns and relations that have already been described in literature.

3 Guiding questions (1st week)

The questions that will guide our work can esentialy be divided in two groups:

Content related questions

- Which are the variables that have a greater impact on whether a patient has type 2 DM?
- Which is the explanation behind such impact? Has it already been reported?
- What other relationships can we find between the variables that appear in type 2 DM diagnosis?
- Can rule based models be used to properly diagnose type 2 DM?
- Can decision tree based models be used to properly diagnose type 2 DM?

Data related questions

- How should missing values be treated in a clinical problem?
- Which imputation methods perform better in scenarios where little is known about the missing values?
- Which are the analysis, exploratory and statistical, that turn datasets into interpretable information?
- How should classification be approached in a clinical problem?

4 Diabetes mellitus: General information (1st week)

In this section, we will study the very basics of diabetes. The information was taken from academic medicine textbooks and from [2].

4.1 Definition and social impact

Diabetes mellitus is a complex disease caused by deficiencies in insulin synthesis by pancreatic β cells (there is not enough insulin) or flaws in the action of insulin over its target tissues (the insulin does not work effectively). As a result of such insulin problems, glucose cannot enter the cells to provide energy, and it accumulates in the bloodstream instead. This can produce both acute and chronic complications, being the cardiovascular the most relevant.

Due to its high incidence, diabetes has an important impact on public health. It is estimated that this disease affects 450 million people in the world. The WHO (World Health organization) predicts a progression towards epidemic proportions; by 2040, it will affect 650 million people. In addition, complications associated to diabetes make it one of the main causes of morbimortality in our current society.

4.2 Pathogeny, epidemiology, symptomatology and treatment

4.2.1 Type 1 diabetes mellitus (type 1 DM)

It constitutes 5-10% of all cases and results from an absolute deficit of insulin secretion due to pancreatic β cells destruction, which can have an idiopathic (type 1B) or autoimmune (type 1A) origin. In the latter, both humoral -antibodies- and cellular immunity -cytotoxic lymphocytes T and macrophages- take part.

As for the genetic factors, type 1 DM is attributed to multiple gene polymorphisms. The HLA histocompatibility locus (chromosome 6) seems especially relevant as more than 90% of type 1 diabetics carry HLA-DR3 and/or HLA-DR4 haplotypes. With family history, the probability to develop type 1 DM is of 5-10% (5 times higher if it comes from the father). As for the environmental factors, type 1 DM can be triggered, in genetically predisposed individuals, by some processes (viral infections, early exposition to serum albumin, milk casein or cereals, exposition nitrates, vitamin D or ω-3 acids deficiency). The concordance between homozygotic twins oscillates between 30-70%. In terms of epidemiology, type 1 DM can appear at any age (more frequent under 35 and with peaks between 5-7 and 11-16 years old), sex (from 14 years old onwards, more common in male) or time of the year (higher incidence in cold months). Regarding the symptoms, they appear brusquely and are polyuria (increased production of urine), polydipsia (increased thirst), polyphagia (increased appetite), asthenia (lack of energy or strength) and weight loss. Type 1 DM treatment requires administration of insulin or a pancreatic transplant.

4.2.2 Type 2 diabetes mellitus (type 2 DM)

It constitutes 90% of all cases. It is initially caused by a flaw in the action of insulin over its target tissues. Such resistance, which is explained by the receptors incapability to recognize insulin, makes the body think that no insulin is being produced at all, so the pancreatic β cells keep producing even when insulin is at high levels. At late stages of the diseases, overuse leads to deterioration and so insulin is produced at low levels, leading to a relative deficit.

As for the genetic factors, type 2 DM is attributed to genes related with the development and function of the pancreatic β cells and with the synthesis and action of insulin. The genetic influence is more important than in type 1 DM, as it is derived from the fact that 40% of patients have a diabetic parent and the probability to develop diabetes is 5-10 times higher for those with family history. As for the environmental factors, type 2 DM can be triggered, in genetically predisposed individuals, by some processes (obesity, lack of activity, hypercaloric diets, aging). The concordance between homozygotic twins is close to 90%. In terms of epidemiology, type 2 DM can appear at any age (more frequent over 45) or sex (more common in male, increasing with age). Regarding the symptoms, they appear gradually, which complicates its diagnosis. Type 2 DM treatment is based on normalizing blood sugar and controlling the cardiovascular risk factors; weight loss and changing eating and activity habits can correct blood sugar levels; if that were not sufficient, insulin or insulin sensitizer can be used.

4.3 Acute and chronic complications

Some acute complications are diabetic ketoacidosis (type 1 DM), hyperosmolar hyperglycemic decompensation (type 2 DM) and hypoglycemia (secondary effect to pharmacologic treatment). The most common chronic complications are vascular (retinopathy, nephropathy, neuropathy, ischemic cardiophatology, arterial disease), gastrointestinal and dermatological.

4.4 Screening and diagnosis

In a symptomatic patient, a plasma glucose test over 200 mg/dl allows diagnosing diabetes. In an asymptomatic patient, the tests recommended for diagnosing diabetes are:

- Glycates haemoglobin test: 5.7-6.4% → prediabetes, $>$ 6.5% → diabetes.
- 8 hours fasting plasma glucose: 100-126 mg/dl → prediabetes, $>$ 126 mg/dl → diabetes.
- 2 hours glucose tolerance test: 140-200 mg/dl → prediabetes, $>$ 200 mg/dl → diabetes.

The screening must be carried out every 3 years for asymptomatic patients over 45 years old or for patients with risk factors (obesity BMI $\geq$ 25 kg/m2, lack of activity, gestational diabetes, family history, insulin resistance, carbohydrates intolerance, triglycerides increase, HDL decrease, cardiovascular diseases, hypertension, polycystic ovarian syndrome).

The distinction between both types of diabetes can be unclear. In a situation of doubt, the doctors can use antibody (glutamic acid antidescarboxilase or anti-GAD) detection or peptide C detection (low levels in type 1 DM -the pancreatic β cells are destroyed- and high levels in type 2 DM).

5 Dataset: Introduction (2nd week)

The dataset we will use is available at Kaggle (an online community of data scientists and machine learners that allows building predictive models on datasets) at: https://www.kaggle.com/johndasilva/diabetes. This dataset was taken from a hospital in Frankfurt, Germany. All patients are females of at least 21 years old. The objective is to diagnostically predict whether a patient has type 2 diabetes mellitus. There are 8 medical predictors (independent variables) and 1 prediction (dependent variable).

The dataset is loaded and the outcome is changed for a numeric one.

```
diabetes = read.csv("diabetes.csv",header=T,sep=",")
diabetes$Outcome = str_replace(diabetes$Outcome,"yes","1")
diabetes$Outcome = str_replace(diabetes$Outcome,"no","0")
diabetes$Outcome = as.numeric(diabetes$Outcome)
variables = names(diabetes); predictive_variables = variables[1:8]
```

This table shows the structure of the dataset (only the first rows are considered).

```
temp1 = diabetes
temp1 = rename(temp1,DiabPed=DiabetesPedigreeFunction,BloodP=BloodPressure,SkinT=SkinThickness)
```

Pregnancies	Glucose	BloodP	SkinT	Insulin	BMI	DiabPed	Age	Outcome
2	138	62	35	0	33.6	0.127	47	1
0	84	82	31	125	38.2	0.233	23	0
0	145	0	0	0	44.2	0.630	31	1
0	135	68	42	250	42.3	0.365	24	1
1	139	62	41	480	40.7	0.536	21	0
0	173	78	32	265	46.5	1.159	58	0

6 Dataset: Dealing with missing data (2nd week)

6.1 Missing data theory

How missing data should be handled is strongly determined by the missing mechanisms (why certain are values missing). These are usually divided into three groups [2]:

- **Missing completely at random** (MCAR): the missing data mechanism is unrelated to the values of any variable, neither missing nor observed (including values of the missing data variable itself). Most missing data handling methods will give unbiased estimates.
- **Missing at random** (MAR): the missing data mechanism is unrelated to the missing variable but may be related to other observed variables. Many imputation methods handle this degree of missing structure in the data well, because missing values can be explained by other (observed) variables.
- **Missing not at random** (MNAR): the missing data mechanism is related to the missing variable (for example, mostly high or low scores are missing), or of other variables that are not available in the dataset. The missing mechanism is difficult to identify, thus the risk of bias is high.

There are tests to determine whether data are MCAR or not. If a relationship between the probability for missing data and the observed variables cannot be detected (and there are no specific reasons why data is missing), then we assume that the data is MCAR; otherwise, the missing data may be MAR or MNAR. Distinguishing between MAR and MNAR is very difficult, unless there is a knowledge on how the data were collected or some of the missing data can be accessed.

6.2 Data imputation theory

The most common way of handling missing data is **Complete Case Analysis** (CCA), which means deleting all instances with missing information. At first glance, this might seem like the safer choice, since data is not altered; however, CCA implies disregarding information contained in the data set, which could potentially lead to biased results. On the other side, imputation techniques, which replace missing data with plausible values, are often seen as riskier but they are just a way of transferring information already contained in the dataset. Therefore, imputation techniques are usually more trustworthy than CAA. Imputations can be:

6.2.1 Single imputation

For each missing data point, a single likely value is calculated. The problem with single imputation is that, in some cases, it will give an artificially low standard deviation due to an underestimation of the uncertainty in the data. Some examples of single imputations are:

- **Hot deck imputation** (HD): non-parametric method that replaces the missing values for an instance (called the recipient) with observed values from another instance (the donor) which is similar to the recipient in those variables in which both have observed values. Single HD selects the nearest neighbor as the donor [3].
- **K nearest neighbor imputation** (KNN): non-parametric method that aggregates the k values of the nearest neighbors and use them as imputed value.

6.2.2 Multiple imputation

For each missing data point, more than just one single likely value is calculated. Multiple imputation implies performing more than single imputations, generating m different datasets without missing values. Then, the results for each missing value are pooled together to one summary estimate. In this case, the standard error for the pooled result combines the variation within the m complete datasets with the variation between the m complete datasets, resulting in more precise confidence intervals and p-values. Multiple imputation is appropriate if missingness may be well predicted from observed values (if data missing data are MAR). Some examples of multiple imputation are:

- **Hot deck imputation** (HD): non-parametric method that replaces the missing values for an instance (called the recipient) with observed values from another instance (the donor) which is similar to the recipient in those variables in which both have observed values. Multiple HD selects the donor randomly from a set of potential donors [3].
- **Expectation maximization imputation** (EM): iterative method where all variables, except for the response one, are regressors [4].

6.3 Missing data in our dataset

One of the first things we notice in our data is the fact that there are many missing values. We assume that ceros are missing values in the variables 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin' and 'BMI'. In 'Pregnancies', we assume they are not since it makes sense for a woman to never have been pregnant.

The missing data in the dataset are identified as NA.

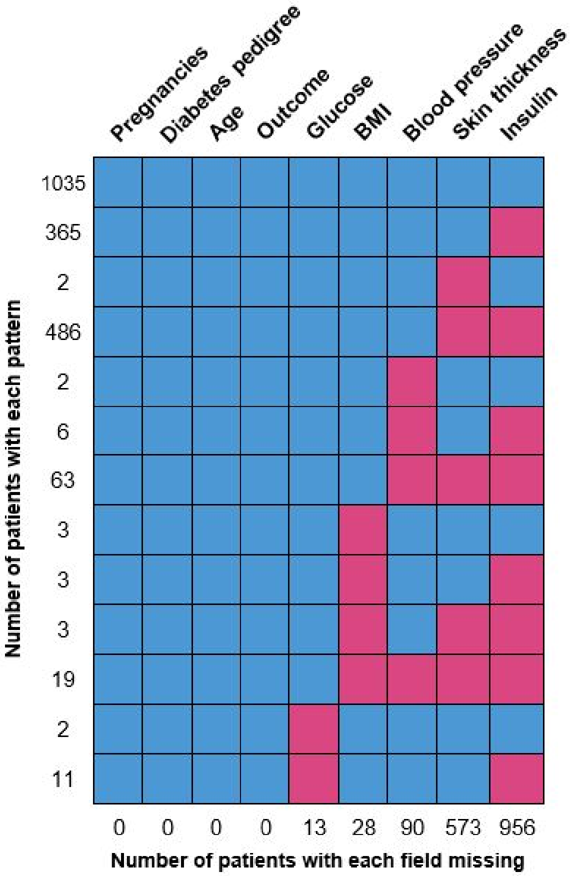
```
diabetes_na = diabetes
diabetes_na$Glucose = diabetes_na$Glucose %>% replace(.==0,NA)
diabetes_na$BloodPressure = diabetes_na$BloodPressure %>% replace(.==0,NA)
diabetes_na$SkinThickness = diabetes_na$SkinThickness %>% replace(.==0,NA)
diabetes_na$Insulin = diabetes_na$Insulin %>% replace(.==0,NA)
diabetes_na$BMI = diabetes_na$BMI %>% replace(.==0,NA)
```

This table shows how missing values are distributed across the different variables.

```
temp1 = diabetes_na
temp1 = rename(temp1,DiabPed=DiabetesPedigreeFunction,BloodP=BloodPressure,SkinT=SkinThickness)
temp1 = as.data.frame(ff_glimpse(temp1)[[1]]); temp1$label = NULL
temp1 = rename(temp1,Q25=quartile_25,Q75=quartile_75,missing_p=missing_percent)
```

	var_type	n	missing_n	missing_p	mean	sd	min	Q25	median	Q75	max
Pregnancies	<int>	2000	0	0.0	3.7	3.3	0.0	1.0	3.0	6.0	17.0
Glucose	<int>	1987	13	0.6	122.0	30.6	44.0	99.0	117.0	141.0	199.0
BloodP	<int>	1910	90	4.5	72.4	12.2	24.0	64.0	72.0	80.0	122.0
SkinT	<int>	1427	573	28.6	29.3	10.8	7.0	22.0	29.0	36.0	110.0
Insulin	<int>	1044	956	47.8	153.7	111.3	14.0	76.8	126.0	190.0	744.0
BMI	<dbl>	1972	28	1.4	32.7	7.2	18.2	27.5	32.4	36.8	80.6
DiabPed	<dbl>	2000	0	0.0	0.5	0.3	0.1	0.2	0.4	0.6	2.4
Age	<int>	2000	0	0.0	33.1	11.8	21.0	24.0	29.0	40.0	81.0
Outcome	<dbl>	2000	0	0.0	0.3	0.5	0.0	0.0	0.0	1.0	1.0

This figure shows how missing values patterns are distributed across groups of patients.



The first step towards handling missing data is to decide which variables to impute and which to ignore. The percentage of patients with missing values in ‘Glucose’ or ‘BMI’ is 2%. This means that, even if the missing values are MAR or MNAR, the bias would have to be extremely high for them to make a change in the result. Thus, we delete the patients that are missing either of them.

```r
diabetes_na = filter(diabetes_na,!is.na(diabetes_na$Glucose))
diabetes_na = filter(diabetes_na,!is.na(diabetes_na$BMI))
```

The second step towards handling missing data is to check its nature. An initial approach implies analyzing the dataset as a whole. We use the function *LittleMCAR()* from the package *BaylorEdPsyck*, whose null hypothesis is that the missing data is MCAR [5]. We conclude that missing data is not MCAR.

```r
temp1 = LittleMCAR(diabetes_na)
temp1 = data.frame(chi_square=temp1$chi.square,p_value=temp1$p.value)
```

chi_square	p_value
606.271	0

A further approach implies analyzing each variable individually. We use the function *missing_compare()* from the package *finalfit*, which takes one variable, divides the dataset in two (missing value vs not missing value) and compares both groups with each variable. When applying this to 'BloodPressure', 'SkinThickness' and 'Insulin', we obtain that all three variables missing values are affected by other variables. We conclude that missing data is either MAR or MNAR.

```
temp1 = diabetes_na %>% missing_compare("BloodPressure",variables)
temp2 = diabetes_na %>% missing_compare("SkinThickness",variables)
temp3 = diabetes_na %>% missing_compare("Insulin",variables)
```

	Missing data analysis: BloodPressure		Not missing	Missing	p
8	Pregnancies	Mean (SD)	3.7 (3.3)	3.0 (3.4)	0.016
4	Glucose	Mean (SD)	122.2 (30.8)	125.2 (27.4)	0.243
9	SkinThickness	Mean (SD)	29.2 (10.4)	47.4 (38.8)	0.561
5	Insulin	Mean (SD)	154.1 (111.4)	215.0 (0.0)	0.120
2	BMI	Mean (SD)	32.7 (7.2)	32.3 (8.5)	0.341
3	DiabetesPedigreeFunction	Mean (SD)	0.5 (0.3)	0.4 (0.3)	0.287
1	Age	Mean (SD)	33.2 (11.8)	32.2 (10.8)	0.968
6	Outcome	0	1249 (66.2)	35 (49.3)	0.003
7		1	639 (33.8)	36 (50.7)	

	Missing data analysis: SkinThickness		Not missing	Missing	p
9	Pregnancies	Mean (SD)	3.4 (3.3)	4.4 (3.3)	<0.001
5	Glucose	Mean (SD)	121.2 (30.8)	125.1 (30.2)	0.001
2	BloodPressure	Mean (SD)	71.7 (12.2)	74.5 (12.2)	<0.001
6	Insulin	Mean (SD)	154.1 (111.4)	215.0 (0.0)	0.120
3	BMI	Mean (SD)	33.1 (7.0)	31.4 (7.8)	<0.001
4	DiabetesPedigreeFunction	Mean (SD)	0.5 (0.3)	0.4 (0.3)	<0.001
1	Age	Mean (SD)	31.5 (10.6)	37.6 (13.4)	<0.001
7	Outcome	0	947 (67.3)	337 (61.2)	0.011
8		1	461 (32.7)	214 (38.8)	

	Missing data analysis: Insulin		Not missing	Missing	p
8	Pregnancies	Mean (SD)	3.2 (3.2)	4.3 (3.4)	<0.001
5	Glucose	Mean (SD)	123.0 (30.8)	121.4 (30.6)	0.377
2	BloodPressure	Mean (SD)	70.8 (12.3)	74.4 (11.9)	<0.001
9	SkinThickness	Mean (SD)	29.4 (11.1)	29.2 (10.0)	0.762
3	BMI	Mean (SD)	33.4 (7.4)	31.8 (7.0)	<0.001
4	DiabetesPedigreeFunction	Mean (SD)	0.5 (0.3)	0.4 (0.3)	<0.001
1	Age	Mean (SD)	30.7 (10.0)	36.0 (12.8)	<0.001
6	Outcome	0	702 (67.6)	582 (63.3)	0.045
7		1	337 (32.4)	338 (36.7)	

The most important thing is whether missing values are related to the outcome. To be sure about this, we perform a chi-square analysis with the number of patients with or without type 2 DM and with or without missing values in the studied variable. In all three cases, the percentage of patients with type 2 DM when the values of a variable are missing is greater. This means that probably, our data is close to being MNAR, because the missing values make a difference in the outcome. These are not good results, because they indicate that there is a bias in our data that cannot be eliminated without further investigation. To get some information on how data was retrieved, we accessed the paper in which the data was collected [6], but missing values are not even mentioned. Therefore, we have no clue why these values are missing.

```r
temp1 = matrix(nrow=2,ncol=2,c(
nrow(filter(diabetes_na,is.na(diabetes_na$BloodPressure)&diabetes_na$Outcome==0)),
nrow(filter(diabetes_na,!is.na(diabetes_na$BloodPressure)&diabetes_na$Outcome==0)),
nrow(filter(diabetes_na,is.na(diabetes_na$BloodPressure)&diabetes_na$Outcome==1)),
nrow(filter(diabetes_na,!is.na(diabetes_na$BloodPressure)&diabetes_na$Outcome==1))))
rownames(temp1) = c("BloodPressureNo","BloodPressureYes")
colnames(temp1) = c("DiabetesNo","DiabetesYes")

temp2 = matrix(nrow=2,ncol=2,c(
nrow(filter(diabetes_na,is.na(diabetes_na$SkinThickness)&diabetes_na$Outcome==0)),
nrow(filter(diabetes_na,!is.na(diabetes_na$SkinThickness)&diabetes_na$Outcome==0)),
nrow(filter(diabetes_na,is.na(diabetes_na$SkinThickness)&diabetes_na$Outcome==1)),
nrow(filter(diabetes_na,!is.na(diabetes_na$SkinThickness)&diabetes_na$Outcome==1))))
rownames(temp2) = c("SkinThicknessNo","SkinThicknessYes")
colnames(temp2) = c("DiabetesNo","DiabetesYes")

temp3 = matrix(nrow=2, ncol=2, c(
nrow(filter(diabetes_na,is.na(diabetes_na$Insulin)&diabetes_na$Outcome==0)),
nrow(filter(diabetes_na,!is.na(diabetes_na$Insulin)&diabetes_na$Outcome==0)),
nrow(filter(diabetes_na,is.na(diabetes_na$Insulin)&diabetes_na$Outcome==1)),
nrow(filter(diabetes_na,!is.na(diabetes_na$Insulin)&diabetes_na$Outcome==1))))
rownames(temp3) = c("InsulinNo","InsulinYes")
colnames(temp3) = c("DiabetesNo","DiabetesYes")

temp1 = chisq.test(temp1)
temp2 = chisq.test(temp2)
temp3 = chisq.test(temp3)

temp4 = data.frame(chi_square=c(temp1$statistic[[1]],temp2$statistic[[1]],temp3$statistic[[1]]),
p_value=c(temp1$p.value,temp2$p.value,temp3$p.value))
rownames(temp4) = c("BloodPressure", "SkinThickness", "Insulin")
```

	chi_square	p_value
BloodPressure	7.881	0.005
SkinThickness	6.251	0.012
Insulin	3.814	0.051

6.4 Data imputation in our dataset

Based on the results obtained, it looks like CCA is not a good methodology. We should then rely on some imputation methods. In the package *VIM*, we can implement HD imputation with the function *hotdeck(),* KNN imputation with the function *kNN()* and EM imputation with the function *irmi()*. As our data do not usually follow a probability distribution (see next section), non-parametric methods (which do not make assumptions about the parameters of the population distributions) should work better. In theory, we would expect EM to perform the best, followed by HD and eventually by KNN. However, EM is not reaching convergence for 'Insulin', even when increasing the number of iterations. Therefore, the results obtained with the EM imputation might not be as good as expected.

We are going to create 5 different datasets. The first is the original dataset with the a few modifications (numerical outputs, missing data identified as NA and 2% of the rows removed). The remaining datasets are the first one after applying CCA, HD imputation, KNN imputation and EM imputation, respectively. These datasets are going to be used for two purposes. For the exploratory and statistical analysis of the variables we need the whole datasets. For the modelling, we need to consider a division between training and testing dataset; this will allow us to build different models based on the training data and evaluation such models in the test. Note that the rows with missing data in the training datasets will be imputed (except in CCA), while the rows with missing data in the test datasets will be removed.

```
# sample=sample(nrow(diabetes_na),0.8*nrow(diabetes_na),replace=FALSE)
# diabetes_na_train = diabetes_na[sample,]
# diabetes_na_test = diabetes_na[-sample,]
# diabetes_na_test = diabetes_na_test[complete.cases(diabetes_na_test),]
# write.csv(diabetes_na,"diabetes_na.csv",row.names=FALSE)
# write.csv(diabetes_na_train,"diabetes_na_train.csv",row.names=FALSE)
# write.csv(diabetes_na_test,"diabetes_na_test.csv",row.names=FALSE)
#
# diabetes_cca = diabetes_na[complete.cases(diabetes_na),][1:9]
# diabetes_cca_train = diabetes_na_train[complete.cases(diabetes_na_train),][1:9]
# write.csv(diabetes_cca,"diabetes_cca.csv",row.names=FALSE)
# write.csv(diabetes_cca_train,"diabetes_cca_train.csv",row.names=FALSE)
#
# diabetes_hd = hotdeck(diabetes_na,
# c("Insulin","BloodPressure","SkinThickness"),domain_var="Outcome")[1:9]
# diabetes_hd_train = hotdeck(diabetes_na_train,
# c("Insulin","BloodPressure","SkinThickness"),domain_var="Outcome")[1:9]
# write.csv(diabetes_hd,"diabetes_hd.csv",row.names=FALSE)
# write.csv(diabetes_hd_train,"diabetes_hd_train.csv",row.names=FALSE)
#
# diabetes_knn = kNN(diabetes_na,dist_var=variables,
# k=5,numFun=median,weights=c(rep(1,8),3))[1:9]
# diabetes_knn_train = kNN(diabetes_na_train,dist_var=variables,
# k=5,numFun=median,weights=c(rep(1,8),3))[1:9]
# write.csv(diabetes_knn,"diabetes_knn.csv",row.names=FALSE)
# write.csv(diabetes_knn_train,"diabetes_knn_train.csv",row.names=FALSE)
#
# diabetes_em = irmi(diabetes_na,maxit=10000,robust=TRUE)[1:9]
# diabetes_em_train = irmi(diabetes_na_train,maxit=10000,robust=TRUE)[1:9]
# write.csv(diabetes_em,"diabetes_em.csv",row.names=FALSE)
# write.csv(diabetes_em_train,"diabetes_em_train.csv",row.names=FALSE)

imputed_variables = c("BloodPressure","SkinThickness","Insulin")
imputations = c("na","cca","knn","hd","em")
```

7 Dataset: Exploring the variables behavior (3rd week)

7.1 Procedure explanation

In this section, we will perform an exploratory and statistical analysis for each of our variables. The exploratory data analysis is an approach of data analysis that summarizes its characteristics without using statistical models or having formulated prior hypothesis; the motivation behind is to select the right tools for processing data and to exploit humans' abilities to recognize patterns not captured by automatic tools. The statistical data analysis is an approach of data analysis that summarizes its characteristics by using statistical models and having formulated a prior hypothesis; the motivation behind is to be more accurate in our conclusions. For each variable, our analysis will include:

1. **Explanation**
2. **Type**
3. **Exploratory data analysis**: we will divide the dataset in 2 populations (healthy/no_diabetes/P0 and sick/yes_diabetes/P1). For each population, we will build a density plot comparing the distribution shape, a boxplot comparing the quartiles, and a barplot comparing the mean. This will give us an idea on how each variable contributes to the outcome.
4. **Statistical data analysis**: for each population (P0 and P1), we will perform a distribution comparison test to confirm whether each variable contributes to the outcome. We first check the normality with the Shapiro test, whose null hypothesis is that a given data fits a normal distribution, through the function *shapiro.test()*. We then check the homoscedasticity with the Breusch-Pagan test, whose null hypothesis is that a given data is homoscedastic, through the function *ncvTest()* from the package *car*. If normality and homoscedasticity assumptions are met, we implement Tukey test (multiple comparison method based on the T Student parametric test, which declares two distributions as different when the difference between there mean is lower than HSD quantity) with the function *TukeyHSD()*. If normality and homoscedasticity assumptions are not met, we implement Pairwise Wilcoxon Rank Sum test (multiple comparison method based on the U Mann-Whitney non-parametric test, which declares distributions X, Y as different when the probability that one observation in X exceeds one observation in Y is different from the probability that one observation in Y exceed one observation in X), with the function *pairwise.wilcox.test()*.
5. **Literature based relations**: we will study the relationships that, according to the scientific literature, exist between each variable and diabetes.
6. **Conclusion**: based on both data and literature, we will state each variable effect on type 2 DM.

7.2 Preparation and selection of the datasets

We prepare the datasets for their proper visualization.

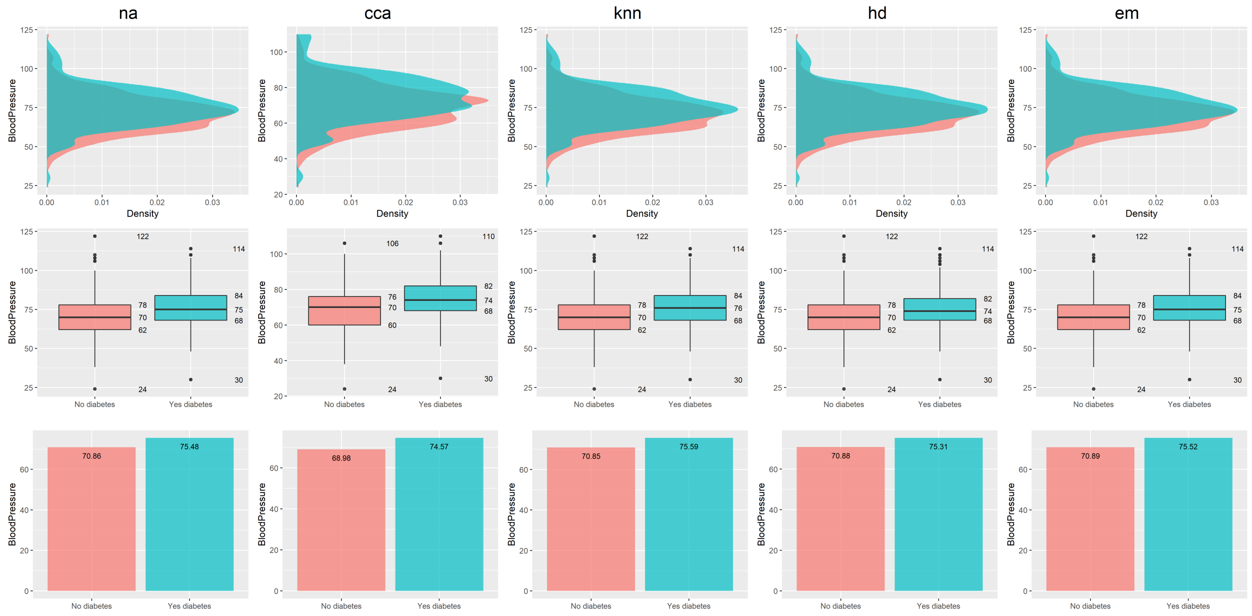
```r
#We load the data and factorize the output

for (imputation in imputations)
{
temp1 = read.csv(paste0("diabetes_",imputation,".csv"),header=T,sep=",")
temp1$Outcome = temp1$Outcome %>% replace(.==0,"No diabetes")
temp1$Outcome = temp1$Outcome %>% replace(.==1,"Yes diabetes")
temp1$Outcome = factor(temp1$Outcome)
assign(paste0("diabetes_",imputation),temp1)
}
```

```r
#We create the dataframes for the exploratory analysis

for (imputation in imputations)
{
for (variable in predictive_variables)
{
temp1 = get(paste0("diabetes_",imputation))
temp2 = data.frame(Variable=temp1[[variable]],Outcome=temp1$Outcome)
temp3 = temp2 %>% group_by(Outcome) %>% summarise(Variable=mean(Variable,na.rm=TRUE))
temp4 = temp2 %>% group_by(Outcome) %>% summarise(Variable=list(fivenum(Variable))) %>% unnest()
assign(paste0(variable,"_",imputation,"_df"),temp2)
assign(paste0(variable,"_",imputation,"_mean"),temp3)
assign(paste0(variable,"_",imputation,"_quantiles"),temp4)
}
}
```

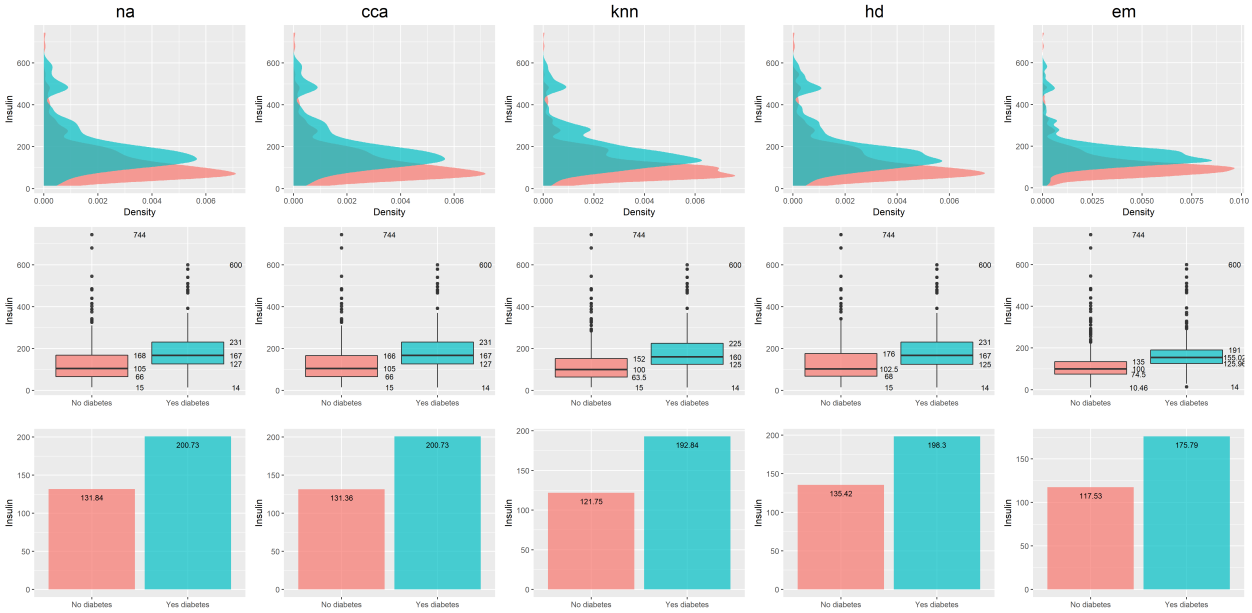
In order to see how the imputations modified our dataset, we draw the data distributions for the 3 variables that were imputed.

```r
#We iterate over each imputed variable and over each imputation

for (variable in imputed_variables)
{
for (imputation in imputations)
{
temp1 = get(paste0(variable,"_",imputation,"_df"))
temp2 = get(paste0(variable,"_",imputation,"_quantiles"))
temp3 = get(paste0(variable,"_",imputation,"_mean"))

#We create the density plot, boxplot and column plot

p1 = ggplot(data=temp1, aes(x=Variable,fill=Outcome)) +
     geom_density(alpha=0.7,color=NA) + coord_flip() +
     labs(x=variable,y="Density",fill="Legend") +
     theme(legend.position="none")
p2 = ggplot(data=temp1, aes(x=Outcome,y=Variable,fill=Outcome)) +
     geom_boxplot(alpha=0.7) +
     geom_text(data=temp2,aes(x=Outcome,y=Variable,label=round(Variable,2)),nudge_x=0.5,size=3) +
     labs(x="", y=variable) +
     theme(legend.position="none")

p3 = ggplot(data=temp3, aes(x=Outcome,y=Variable,fill=Outcome)) +
     geom_col(alpha=0.7) +
     geom_text(aes(label=round(Variable,2)),vjust=2,size=3) +
     labs(x="",y=variable) +
     theme(legend.position="none")

#We create the composed plot

plot = arrangeGrob(p1,p2,p3,heights=c(1,1,1),top=textGrob(imputation,gp=gpar(fontsize=20)))
assign(paste0(imputation,"_",variable),plot)
}

#We create and save the comparison plot

comparison_plot = arrangeGrob(get(paste0("na_",variable)),get(paste0("cca_",variable)),
get(paste0("knn_",variable)),get(paste0("hd_",variable)),get(paste0("em_",variable)),
ncol=5,nrow=1)

ggsave(filename=paste0("All_imputations_",variable,".png"),
plot=comparison_plot,width=20,height=10,path="Figures")
}
```

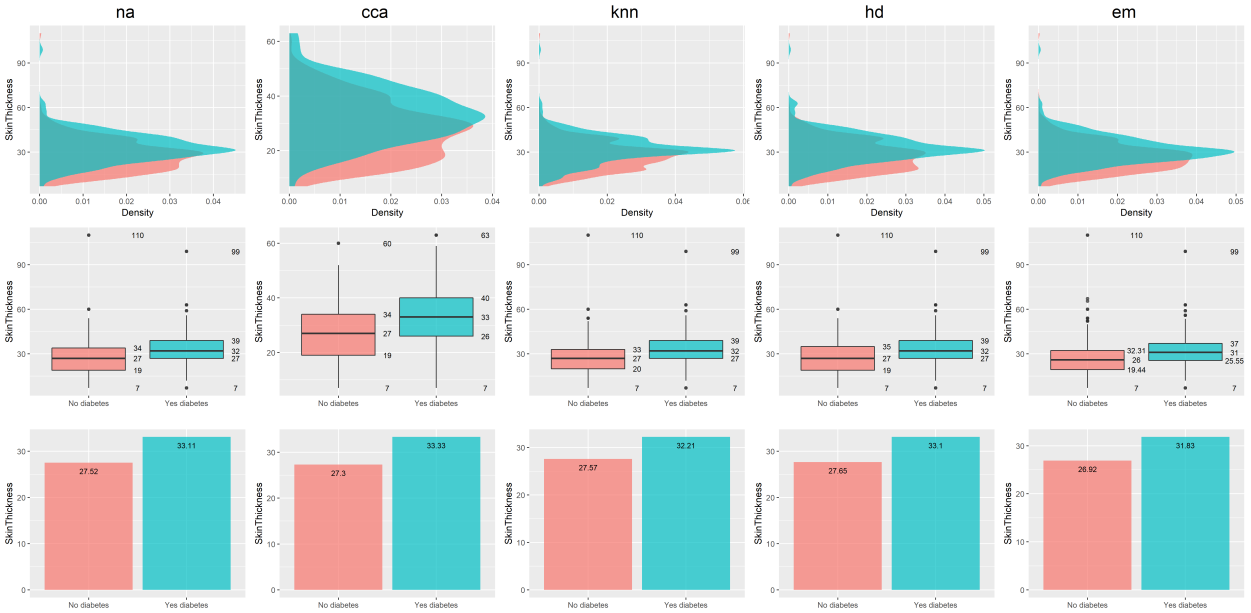
This figure shows how different imputations work for ‘BloodPressure’.



This figure shows how different imputations work for ‘Insulin’.



This figure shows how different imputations work for ‘SkinThickness’.



As we expected, CCA is the method that worst reproduces the original dataset in terms of distribution. The imputation methods seem to do a good job, but it is HD, and not EM, the one that apparently gives closer distributions to the ones of our original data. Due to this, we will be doing our analysis with ‘diabetes_hd.csv’.

7.3 Exploratory data analysis

We perform the exploratory data analysis.

```r
#We iterate over each predictive variable

for (variable in predictive_variables)
{
temp1 = get(paste0(variable,"_hd_df"))
temp2 = get(paste0(variable,"_hd_quantiles"))
temp3 = get(paste0(variable,"_hd_mean"))

#We create the density plot, boxplot and column plot

p1 = ggplot(data=temp1, aes(x=Variable,fill=Outcome)) +
     geom_density(alpha=0.7,color=NA) + coord_flip() +
     labs(x=variable,y="Density",fill="Legend") +
     theme(legend.position="none")

p2 = ggplot(data=temp1, aes(x=Outcome,y=Variable,fill=Outcome)) +
     geom_boxplot(alpha=0.7) +
     geom_text(data=temp2,aes(x=Outcome,y=Variable,label=round(Variable,2)),nudge_x=0.5,size=3) +
     labs(x="", y=variable) +
     theme(legend.position="none")

p3 = ggplot(data=temp3, aes(x=Outcome,y=Variable,fill=Outcome)) +
     geom_col(alpha=0.7) +
     geom_text(aes(label=round(Variable,2)),vjust=2,size=3) +
     labs(x="",y=variable) +
     theme(legend.position="none")

#We create and save the composed plot

plot = arrangeGrob(p1,p2,p3,widths=c(1,1,1))
ggsave(filename=paste0(variable,".png"),plot=plot,width=10,height=4,path="Figures")
}
```

The results of the analysis will be shown and commented in the ‘Results’ section.

7.4 Statistical data analysis

We perform the statistical data analysis.

```r
#We prepare the inputs for the statistical analysis

temp1 = read.csv("diabetes_hd.csv",header=T,sep=",")
temp2 = filter(temp1,temp1$Outcome==0)
temp3 = filter(temp1,temp1$Outcome==1)

comparisons = data.frame(Variable=character(),Normality_P0=numeric(),
Normality_P1=numeric(),Homoscedasticity=numeric(),Tukey=numeric(),Wilcoxon=numeric())

for (variable in predictive_variables)
{

P0 = temp2[[variable]][!is.na(temp2[[variable]])]
P1 = temp3[[variable]][!is.na(temp3[[variable]])]
factors = as.factor(temp1$Outcome)
values=temp1[[variable]]
model = lm(values~factors,data=data.frame(factors,values))
anova = aov(model)

#We perform the statistical analysis

normality_P0 = shapiro.test(P0)$p.value
normality_P1 = shapiro.test(P1)$p.value
wilcoxon = pairwise.wilcox.test(values,factors)$p.value[[1]]
homoscedasticity = ncvTest(model)$p
tukey = TukeyHSD(anova,"factors")$factors[[4]]

#We save the results from the statistical analysis

comparison = data.frame(Variable=variable,Normality_P0=normality_P0,
Normality_P1=normality_P1,Homoscedasticity=homoscedasticity,
Tukey=tukey,Wilcoxon=wilcoxon)

comparisons = bind_rows(comparisons,comparison)
}
```

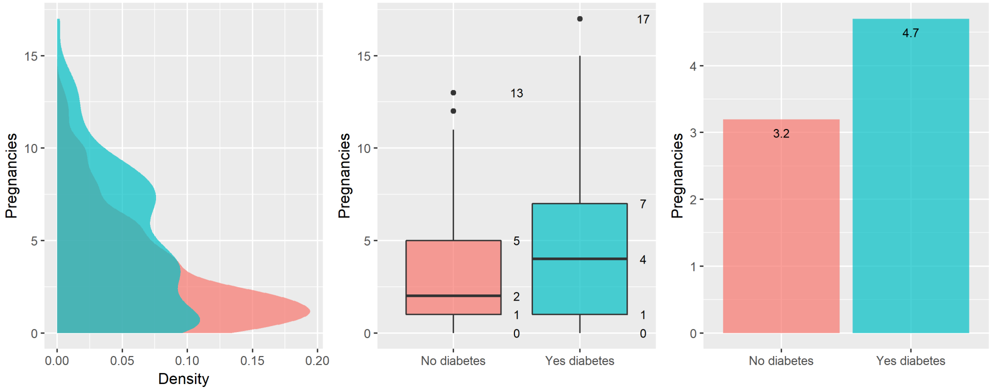
These are the results of the analysis, that will be commented in the 'Results' section.

Variable	Normality_P0	Normality_P1	Homoscedasticity	Tukey	Wilcoxon
Pregnancies	1.390000e-30	1.571583e-16	3.862764e-15	3.769907e-11	1.757980e-16
Glucose	2.825228e-15	2.404834e-09	9.749397e-06	3.768019e-11	0.000000e+00
BloodPressure	3.592894e-08	2.686546e-07	9.821899e-01	3.771172e-11	2.575028e-15
SkinThickness	1.483288e-20	2.602464e-15	5.150222e-02	3.769174e-11	7.117000e-29
Insulin	0.000000e+00	3.935253e-25	2.238725e-05	3.768019e-11	0.000000e+00
BMI	6.897149e-21	4.020314e-15	2.939529e-01	3.768019e-11	0.000000e+00
DiabetesPedigreeFunction	0.000000e+00	1.313256e-23	3.308536e-13	3.912504e-11	6.864122e-11
Age	0.000000e+00	1.031258e-13	8.508669e-02	3.769907e-11	0.000000e+00

7.5 Results

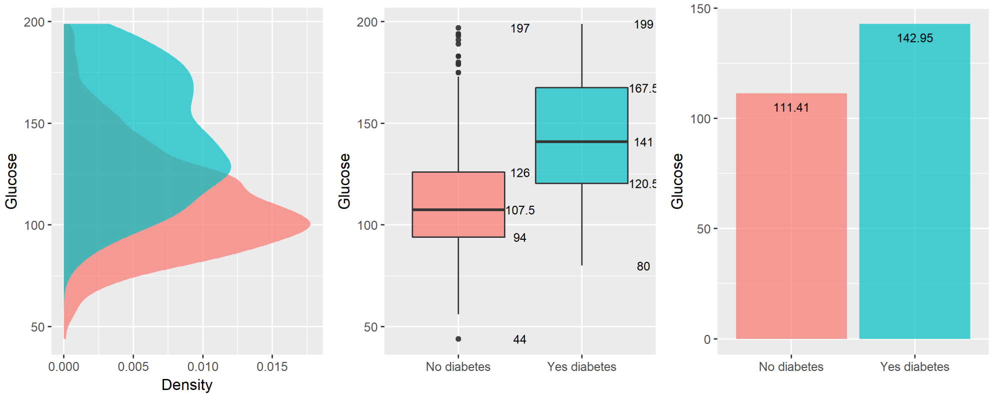
7.5.1 Pregnancies

‘Pregnancies’ refers to the number of times each woman got pregnant. It is a numeric, discrete and ordinal variable. From the exploratory data analysis, we can deduce that P1 has greater values of ‘Pregnancies’. The distributions are the followings: P0 has many instances with low values and few instances with high values; although this tendency is shared with P1, instances are not as grouped around lower values and there are more instances with high values. From the statistical analysis, we can verify that both populations are statistically different. What does literature say? Pregnancy can sometimes lead to gestational diabetes, which can itself become chronic and turn into type 2 DM [7, 8]. The data and the literature point to the same conclusion, having type 2 DM is associated with high values of ‘Pregnancies’.



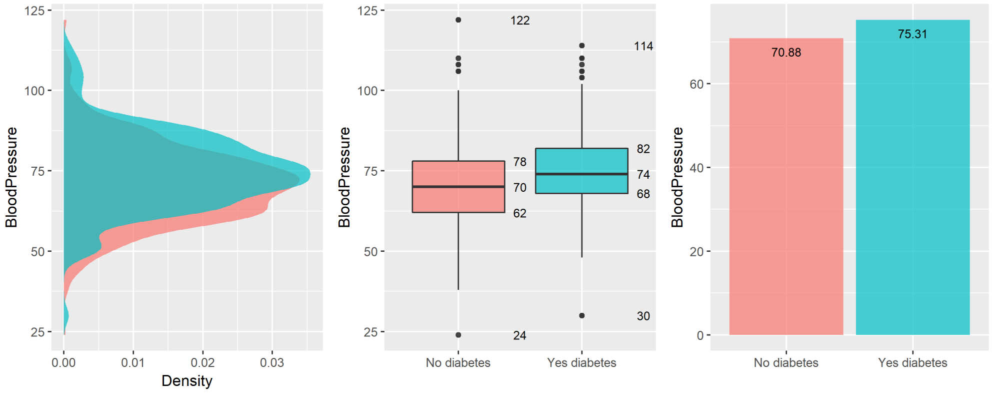
7.5.2 Glucose

‘Glucose’ refers to the plasma glucose concentration 2 hours after an OGTT in mg/dl. It is a numeric and continuous variable. From the exploratory data analysis, we can deduce that P1 has greater values of ‘Glucose’. The distributions are the followings: P0 follows a distribution with most instances grouped around the peak and longer tail at higher values; P1 is moved toward higher values and has a distribution that is more uniform in how instances are spread. From the statistical analysis, we can verify that both populations are statistically different. What does literature say? Type 2 DM is characterized by the presence of high levels of plasmatic glucose (due to flaws in the action of insulin over its target tissues). It is precisely glucose levels that are measured for diagnosing diabetes. The data and the literature point to the same conclusion, having type 2 DM is associated with high values of ‘Glucose’.



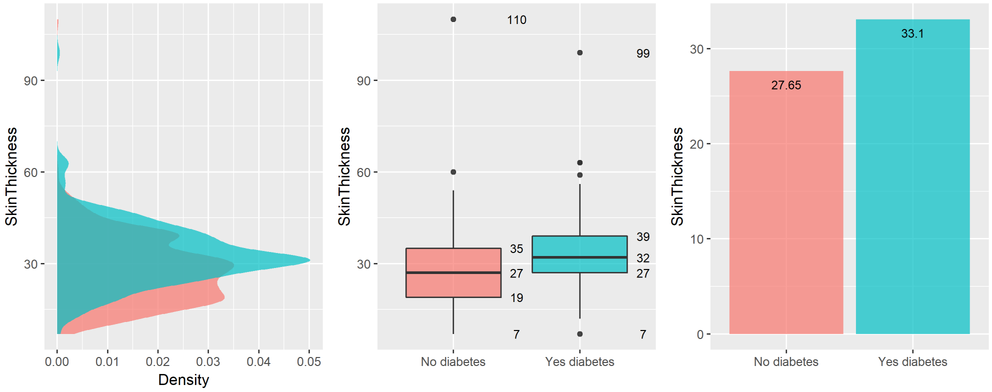
7.5.3 BloodPressure

‘BloodPressure’ refers to the diastolic blood pressure in mmHg. It is a numeric and continuous variable. From the exploratory data analysis, we can deduce that P1 has greater values of ‘BloodPressure’, but such difference do not look significant. Both P0 and P1 follow similar distributions, with most instances grouped around a short range of values. From the statistical analysis, we can verify that both populations are statistically different. What does literature say? Hypertension and type 2 DM share risk factors (obesity, inflammation, oxidative stress, insulin resistance), worsen each other’s symptoms and can apparently be cause of the other [9]. Diabetes causes hypertension by damaging blood vessels and kidneys with high levels of blood glucose; hypertension causes diabetes through trigger of certain processes. The data and the literature point to the same conclusion, having type 2 DM is associated with high values of ‘BloodPressure’.



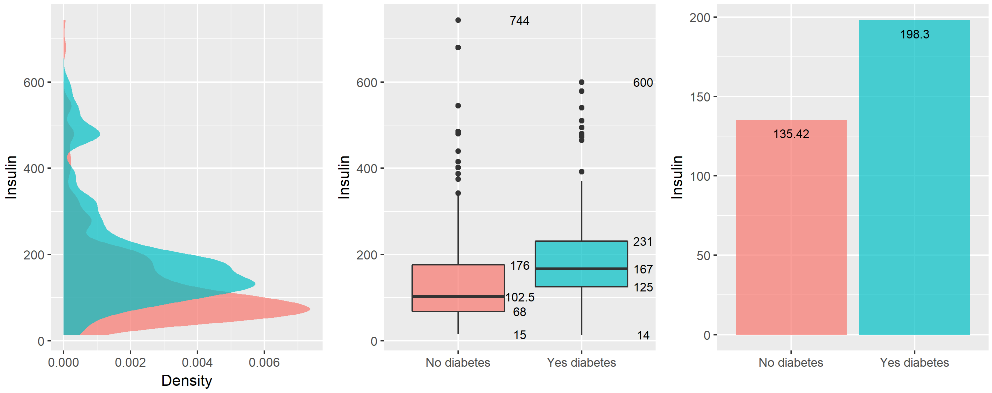
7.5.4 SkinThickness

‘SkinThickness’ refers to the triceps skin fold thickness in mm. It is a numeric and continuous variable. From the exploratory data analysis, we can deduce that P1 has greater values of ‘SkinThickness’. Both P0 and P1 follow similar distributions, with most instances grouped around a short range of values (although some instances have values far away from the mean); the difference is that P1 is moved towards higher values. From the statistical analysis, we can verify that both populations are statistically different. What does literature say? Several dermatological diseases associated to type 2 DM thicken the skin [10], which has particularly been tested in hands and feet. The data and the literature point to the same conclusion, having type 2 DM is associated with high values of ‘SkinThickness’.



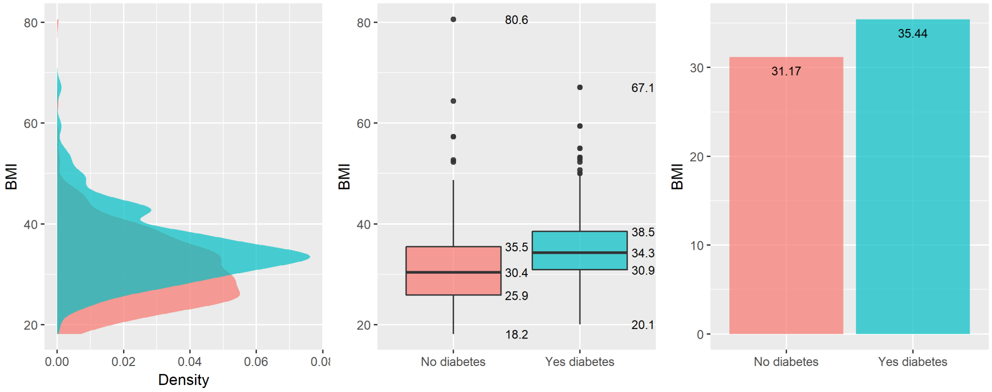
7.5.5 Insulin

'Insulin' refers to the 2 hour serum insulin in mU/ml. It is a numeric and continuous variable. From the exploratory data analysis, we can deduce that P1 has greater values of 'Insulin'. Both P0 and P1 follow similar distributions, with most instances grouped around a very short range of values (although several instances have values far away from the mean); the difference is that P1 is moved towards higher values. From the statistical analysis, we can verify that both populations are statistically different. What does literature say? Type 2 DM is initially caused by a flaw in the action of insulin over its target tissues. Such resistance, which is explained by the receptors incapability to recognize insulin, makes the body think that no insulin is being produced at all, so the pancreatic β cells keep producing even when insulin is at high levels. So at early stages at least, high levels of glucose coexist with high levels of insulin. The data and the literature point to the same conclusion, having type 2 DM is associated with high values of 'Insulin'.



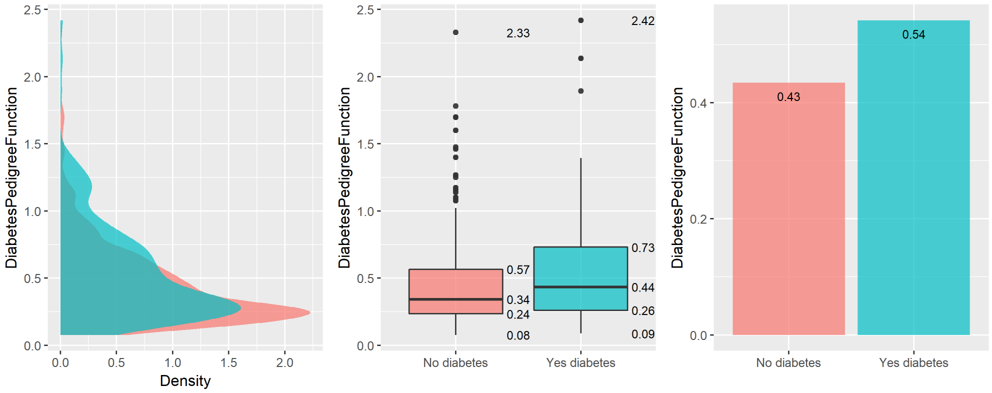
7.5.6 BMI

'BMI' refers to the body mass index in kg/m^2. It is a numeric and continuous variable. From the exploratory data analysis, we can deduce that P1 has greater values of 'BMI'. Both P0 and P1 follow similar distributions, with most instances grouped around a short range of values (although some instances have values far away from the mean); the difference is that P1 is peakier and moved to higher values. From the statistical analysis, we can verify that both populations are statistically different. What does literature say? An increase in body mass leads to an increased risk of having type 2 DM [11, 12]; this is actually one of the most well-known risk factors of the disease. The data and the literature point to the same conclusion, having type 2 DM is associated with high values of 'BMI'.



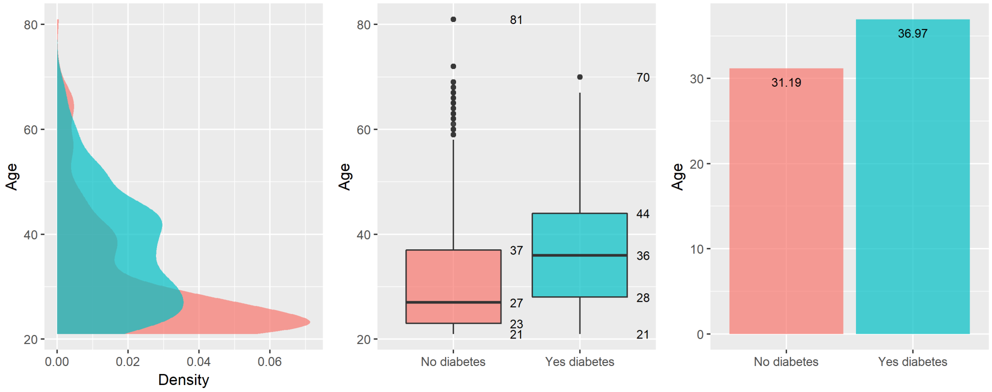
7.5.7 DiabetesPedigreeFunction

‘DiabetesPedigreeFunction’ refers to the likelihood of diabetes based on family history. It is a numeric and continuous variable. From the exploratory data analysis, we can deduce that P1 has greater values of ‘DiabetesPedigreeFunction’, but such differences do not look significant. Both P0 and P1 follow similar distributions, with most instances grouped around a short range of values (although some instances have values far away from the mean); the difference is that the tail decreases faster in P0. From the statistical analysis, we can verify that both populations are statistically different. What does literature say? Type 2 DM is greatly influenced by genetic factors. Apparently, 40% of type 2 DM patients have a parent with the disease and the probability to develop the disease is 5-10 times higher for those with family history. The data and the literature point to the same conclusion, having type 2 DM is associated with high values of ‘DiabetesPedigreeFunction’.



7.5.8 Age

‘Age’ refers to the patient age in years. It is a numeric, discrete and ordinal variable. From the exploratory data analysis, we can deduce that P1 has greater values of ‘Age’. The distributions are the followings: P0 follows has most instances grouped around the peak and longer tail at higher values; P1 is moved toward higher values and has a distribution that is more uniform in how instances are spread. From the statistical analysis, we can verify that both populations are statistically different. What does literature say? Not only type 2 DM is more common over 45, but the prevalence among elder is eight times higher than among adults, which might result from an age-related decline in β cells function [13]. The data and the literature point to the same conclusion, having type 2 DM is associated with high values of ‘Age’.



8 Dataset: Exploring the variables interactions (3rd week)

8.1 Correlation test theory

Correlation tests are used to evaluate the association between two or more variables. To perform a correlation analysis, there are two main types of methods. The parametric correlation tests like the Pearson measure linear dependence between two variables and assume normality in both variables compared. The non-parametric correlation tests like the Spearman rho use rank-based correlation coefficients which do not depend on the distribution of the data. Since our variables do not fit normal distributions, the Spearman rho method seems be appropriate. To implement it, we will use the function *rcorr()*.

8.2 Correlation test in our dataset

We perform the correlation analysis.

```r
temp1 = read.csv("diabetes_hd.csv",header=T,sep=",")
temp1 = rename(temp1,DiabPed=DiabetesPedigreeFunction,BloodP=BloodPressure,SkinT=SkinThickness)
temp2 = rcorr(as.matrix(temp1),type="spearman")$r; temp2[upper.tri(temp2)] = NA
temp2 = temp2 %>% melt(na.rm=TRUE) %>% arrange(desc(abs(value)))
temp3 = temp2 %>% filter(Var1!=Var2 & (Var1=="Outcome" | Var2=="Outcome"))
temp4 = temp2 %>% filter(Var1!=Var2 & (Var1!="Outcome" & Var2!="Outcome"))
```

This table shows the correlations of the variables with the outcome, which can give us an idea of which variable will more relevant in a predictive scenario. While the past analysis was only capable of showing each variable relationship with the outcome, this analysis puts such relations into context, enabling comparison. The variables with a clearer relation with the outcome are ‘Glucose’ and ‘Insulin’, which does not come as a surprise. Other important variables are ‘BMI’, a very well-known risk factor, and ‘Age’. It is surprising, however, that ‘DiabetesPedigreeFunction’ appears so low in the hierarchy given the importance of genetics in the disease. Even though some variables are clearly more important than others, we will not do a feature selection for two reasons. The first is that we have already proved that all variables are relevant. The second is that the number of variables is small, so removing some of them seems unnecessary.

Var1	Var2	value
Outcome	Glucose	0.480
Outcome	Insulin	0.332
Outcome	Age	0.303
Outcome	BMI	0.291
Outcome	SkinT	0.252
Outcome	Pregnancies	0.186
Outcome	BloodP	0.179
Outcome	DiabPed	0.147

This table shows the main correlations between the variables that are not the outcome.

Var1	Var2	value
Age	Pregnancies	0.597
BMI	SkinT	0.501
Insulin	Glucose	0.425
Age	BloodP	0.358
BMI	BloodP	0.271

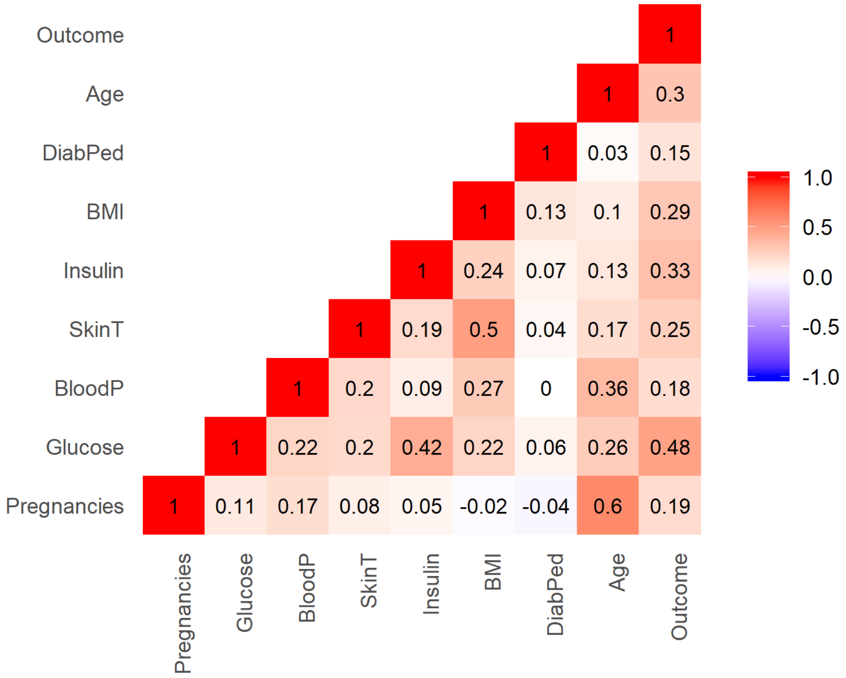
Their explanation is the following:

- 'Age' and 'Pregnancies' → the older a woman, the more pregnancies it has had.
- 'BMI' and 'SkinThickness' → higher body mass are associated with thicker skin.
- 'Insulin' and 'Glucose' → this correlation makes sense in the context of the disease.
- 'Age' and 'BloodPressure' → with the age, blood pressure tends to increase.
- 'BMI' and 'BloodPressure' → higher body mass are associated with increased blood pressure.

We prepare the correlation plot.

```
plot = ggplot(temp2,aes(Var1,Var2,fill=value)) +
       geom_tile() +
       scale_fill_gradient2(low="blue",high="red",mid="white",limit=c(-1,1),name=NULL) +
       geom_text(aes(Var1,Var2,label=round(value,2)),color="black",size=3) +
       theme_minimal() +
       theme(axis.text.x=element_text(angle=90,vjust=1,hjust=1),axis.title.x=element_blank(),
       axis.title.y=element_blank(),axis.ticks=element_blank(),panel.grid.major=element_blank(),
       panel.border=element_blank(),panel.background=element_blank())

ggsave(filename="Correlation matrix.png",plot=plot,width=5,height=4,path="Figures")
```

This figure shows the correlation between all values.



9 Dataset: Modelling (4th week)

The problem we are facing is an example of supervised machine learning (the data that is given to the algorithm contains the results). Moreover, we are talking about a classification problem, where our algorithm needs to distinguish between the combination of values that belong to a diabetic person and the combination of values that belong a non-diabetic person. In other words, the final objective is to assign a category or label to each observation. Although there are many classifiers that could be used for such purpose, we should filter out black boxes that do not allow for interpretability. Clinicians do not only demand for good predictors; they want to understand how the algorithm is taking the decisions. This is why we have tried to build models that allow for some interpretability.

9.1 Tools used

9.1.1 WEKA

WEKA (Waikato Environment for Knowledge Analysis) is a data mining and machine learning tool developed by the Department of Computer Science from the University of Waikato. It includes a collection of algorithms that cover task like preprocessing, classification, clustering, association, feature selection and Visualization.

9.1.2 MLlib

Apache Spark is a unified analytics engine for big data processing that can be implemented in Python. Some interesting features about Spark are the followings: it is parallel (each executer operates on data which is local to it), lazy (execution plans are built but only when the result is needed the computation starts), chained (operations can be concatenated) and reduces the access to memory. Spark is based on DataFrames, two dimensional structures where each column has a specific datatype and each row contains a record.

MLlib is a Spark library that makes machine learning scalable and easy. In machine learning, it is common to run a sequence of algorithms to learn from data. MLlib represents such workflow as a pipeline, which consists of a sequence of stages, applied to DataFrames, to be run in a specific order. In transformer stages, the method *transform()* converts a DataFrame into another, generally by appending one or more columns. In the estimator stages, the method *fit()* takes a DataFrame and generates a model, which is itself a transformer that becomes part of the pipeline. A whole pipeline can be regarded as an estimator.

An important task in machine learning is model selection (finding the best model or the best parameters for a task). In MLlib, the model selection tools work in the following way. First, training data is split into training and validation datasets. Then, there is an iteration over the set of parameters (for each combination, they estimator is fitted with training data and the model performance is evaluated with an evaluator). Eventually, the model with the best-performing set of parameters is selected. The most efficient model selection tool is cross-validation, where the dataset is split into folds that are used to separate training-validation pairs.

9.2 Methodologies used

9.2.1 Classification in WEKA

Although there are many classifiers in WEKA, we have opted for Rule Induction models. Rule Induction is an area of machine learning in which formal rules are extracted from a set of observations. Usually, rules are expressions of the form *if (attribute A has value X1) and (attribute B has value X2) . . . then (decision has value Y)* which are obtained through sequential coverage (rules are induced from examples, then all the positive examples covered by this rule are removed, this is repeated until the whole dataset has been covered, and finally rules are post-processed). We will be using Repeated Incremental Pruning to Produce Error Reduction (RIPPER).

9.2.2 Clasification in MLlib

Although there are many classifiers in MLlib, we have opted for Decision Tree based models. This is basically for two reasons. The first is that we measured several classifiers accuracy in predicting type 2 DM from our dataset; Naive Bayes performed badly, Multilayer Perceptron and Logistic Regressions did a decent job, but Decision Trees and its derivated, Random Forests and Gradient Boosted Trees, clearly outperformed the others (the configuration of all these classifiers is available at *DiabetesPredictiveModels.ipynb*). The second is that they allow for interpretation. Decision trees are machine learning algorithms that do recursive binary partitioning of the feature space to map observations into labels. Both Random Forests and Gradient Boosted Trees are ensembles of decision trees that predict the preferred label to reduce overfitting risk or perform iterative training to minimize loss function.

9.3 Results

9.3.1 Interpretations

From the results obtained in the Rule Induction and Decision Trees model, we can leap to some conclusions:

- ‘Pregnancies’: the rules usually suggest that high values are associated with having type 2 DM; however, some other rules suggest different scenarios. In the trees, ‘Pregnancies’ appears at bottom branches and with changeable patterns in each branch.
- ‘Glucose’: the rules almost always suggest that values higher than the median are associated with having type 2 DM. In the trees ‘Glucose’ appears at top branches and with clear patterns (for instance, there seems to clear separation around 125 mg/ml and a definite threshold between 140 and 170 mg/ml, depending on the imputation).
- ‘Blood pressure’: the rules do not suggest anything relevant as the values associated with having type 2 DM are close to the median. In the trees, ‘BloodPressure’ appears at bottom branches and with changeable patterns in each branch.
- ‘SkinThickness’: the rules usually suggest that high values are associated with having type 2 DM; however, some other rules suggest different scenarios. In the trees, ‘SkinThickness’ appears at bottom branches and with changeable patterns in each branch.
- ‘Insulin’: the rules almost always suggest that values higher than the median are associated with having type 2 DM. In the trees, ‘Insulin’ appears at top branches and with clear patterns (for instance, there seems to be a clear separation aroung 120 mU/ml, but no definite threshold is seen). Interestingly, CCA gives less importance to ‘Insulin’; this shows that imputations turned this variable, with lots of missing data, into an important model feature.
- ‘BMI’: the rules almost always suggest that values higher than the median are associated with having type 2 DM. In the trees, ‘BMI’ appears at top branches and with clear patterns (for instance, there seems to be a clear separation around 30 kg/m^2, and a definite threshold around 45 kg/m^2).
- ‘DiabetesPedigreeFunction’: the rules associate very diverse values to having type 2 DM. In the trees, ‘DiabetesPedigreeFunction’ appears at both top and bottom branches but with changeable patterns in each branch.
- ‘Age’: the rules usually suggest that high values are associated with having type 2 DM; however, some other rules suggest different scenarios. In the trees, ‘Age’ appears at bottom branches and with changeable patterns in each branch.

It is interesting to see how this conclusions look quite similar to the ones we reached in the previous section. Just as before, ‘Glucose’, ‘Insulin’ and ‘BMI’ appear as key variables in determining the outcome; ‘Age’, however, does not look as relevant. It is important to note that the rules and the tree paths can be of greater use than the simple correlation values because they draw profiles of patients with greater risk of having type 2 DM In other words, this gives the doctor a combination of variables to look at when trying to diagnose.

9.3.2 Performance

Although we were esentially pursuing for explainable models, it is interesting to see the performance of both Rule Induction and Decision Tree models in comparison to other classifiers that were tested. What looks obvious from the results is that Decision Trees clearly outperform other methods. Not only this, but ensembles of trees (Random Forests and Gradient Boosted Trees) are not significantlly better than single trees predicting on their own. This comes to show that the diabetes problem (at least the one we are facing) can be easily classified through a simple partitioning of the feature space, yielding accuracies of around 95%. It would be unfair, however, to take these results as totally valid. Most of the classifiers were trained at a very basic level; for instance, the Multilayer Perceptron was given relatively simple structures and it could perfectly be scaled up. In contrast, Decision Tree based algorithms were implemented with high depth and iterations.

This comparison is also relevant for determining the imputation method that worked best. Although there is no method with clearly better results, it looks like CCA is producing models which are bit more precise. This comes as a surprise to us, since theory was pointing in the opposite direction and the data distribution, in the imputed variables, that CCA draw looked quite different from the original one. In any case, it is important to note that CCA by itself cannot be a solution as it is unable to work with data that has missing values (patients for which not all the the variables have been measured). We would either need a methodology that builds such values based on already stored profiles (this is what imputation is basically about) or a predictor that is capable of dealing with such missing values. This would probably be the next step to take.

These are the accuracies obtained, for each imputation and algortihm, in the test dataset.

Methods	CCA	KNN	HD	EM
Naive Bayes	0.659	0.677	0.645	0.668
Multilayer Perceptron	0.682	0.696	0.691	0.728
Logistic Regression	0.783	0.774	0.770	0.774
Rule Induction	0.865	0.855	0.885	0.865
Decision Tree	0.959	0.940	0.972	0.968
Random Forest	0.972	0.963	0.945	0.959
Gradient Boosted Tree	0.982	0.926	0.931	0.912

10 Conclusions and Proposals (4th week)

This work was not aiming for specific results. Our main purpose was to gain knowledge on diabetes through a specific dataset, and this was achieved successfully. From the technical point of view, we had to deal with a very realistic data science problem, which means receiving raw information, and dealing with problems like missing data in order to continue with the variable analysis, approached from both an exploratory and statistical perspective. This was completed with an explanation orientated modelling that could corroborate the conclusions we had already reached. From the content point of view, we had to go from a very general knowledge in diabetes, to a deep understanding on how each variable contributes to having type 2 DM.

We were able to see, in our dataset, the same relations that had already been described in the literature. It would be useful, however, to put greater emphasis on which variables are more relevant and to build clearer patient profiles (one variable might not very important on its own, but combined with another one, it could turn definitive).

We also developed models that gave a high accuracy in our dataset. However, the imputations used turned out not to be very useful and new approaches could be considered. In addition, our models are uncapable of working with missing data, which could be an inconvenient in the clinical world. Although one could impute the upcoming missing data with a pre-existing dataset, it would be more appropriate the build models that could deal with such inconsistencies.

References

[1] American Diabetes Association, "Classification and Diagnosis of Diabetes", Diabetes Care 2017, vol.40, no.1, pp.11-24, 2017

[2] M. R. Stavseth, T. Clausen, and J. Røislien, "How handling missing data may impact conclusions: A comparison of six different imputation methods for categorical questionnaire data," SAGE Open Med., vol. 7, p. 205031211882291, Jan. 2019.

[3] R. R. Andridge and R. J. A. Little, "A review of hot deck imputation for survey non-response," International Statistical Review, vol. 78, no. 1. pp. 40-64, Apr-2010.

[4] M. Templ, A. Kowarik, and P. Filzmoser, "Iterative stepwise regression imputation using standard and robust methods," Comput. Stat. Data Anal., vol. 55, no. 10, pp. 2793-2806, Oct. 2011.

[5] R. J. A. Little, "A test of missing completely at random for multivariate data with missing values," J. Am. Stat. Assoc., vol. 83, no. 404, pp. 1198-1202, 1988.

[6] J. W. Smith, J. E. Everhart, W. C. Dicksont, W. C. Knowler, and R. S. Johannes, "Using the ADAP Learning Algorithm to Forecast the Onset of Diabetes Mellitus."

[7] C. Kim, K. M. Newton, and R. H. Knopp, "Gestational diabetes and the incidence of type 2 diabetes: a systematic review.," Diabetes care, vol. 25, no. 10. pp. 1862-1868, 2002.

[8] L. Bellamy, J. P. Casas, A. D. Hingorani, and D. Williams, "Type 2 diabetes mellitus after gestational diabetes: a systematic review and meta-analysis," Lancet, vol. 373, no. 9677, pp. 1773-1779, 2009.

[9] B. M. Y. Cheung and C. Li, "Diabetes and hypertension: Is there a common metabolic pathway?," Current Atherosclerosis Reports, vol. 14, no. 2. pp. 160-166, Apr-2012.

[10] A. C. Huntley and R. M. Walter, "Quantitative determination of skin thickness in diabetes mellitus: relationship to disease parameters.," J. Med., vol. 21, no. 5, pp. 257-64, 1990.

[11] M. L. Ganz, N. Wintfeld, Q. Li, V. Alas, J. Langer, and M. Hammer, "The association of body mass index with the risk of type 2 diabetes: a case-control study nested in an electronic health records system in the United States," Diabetol. Metab. Syndr., vol. 6, no. 1, p. 50, 2014.

[12] H. E. Bays, R. H. Chapman, and S. Grandy, "The relationship of body mass index to diabetes mellitus, hypertension and dyslipidaemia: comparison of data from two national surveys," Int. J. Clin. Pract., vol. 61, no. 5, pp. 737-747, Apr. 2007.

[13] E. Selvin and C. M. Parrinello, "Age-related differences in glycaemic control in diabetes," Diabetologia, vol. 56, no. 12. pp. 2549-2551, 2013.