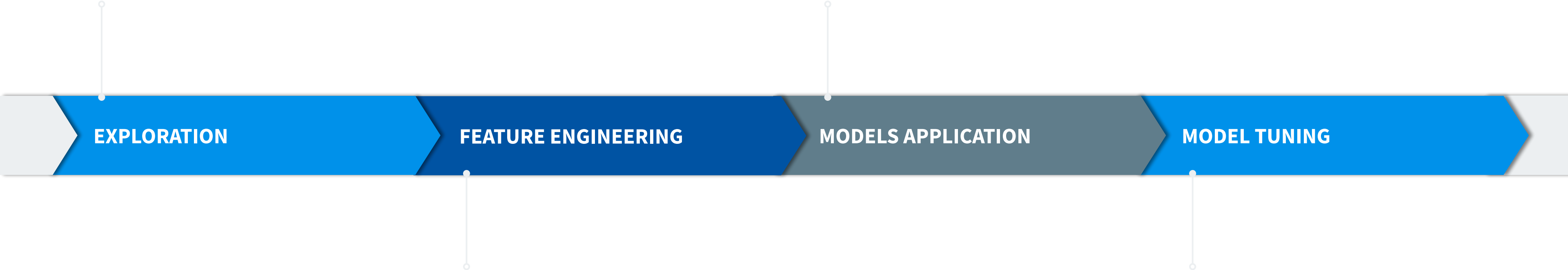
Diabetes Prediction

Machine Learning and Data Analysis, 2022-2023

La Corte Lorenzo (S4784539)

Timeline



EXPLORATION

The first step was understand, clean and analyse the dataset

FEATURE ENGINEERING

Then external knowledge deriving from academic articles and studies has been used in order to create new features, which are used to get the best results out of the dataset

MODELS APPLICATION

Then four different algorithms have been applied through cross-validation

MODEL TUNING

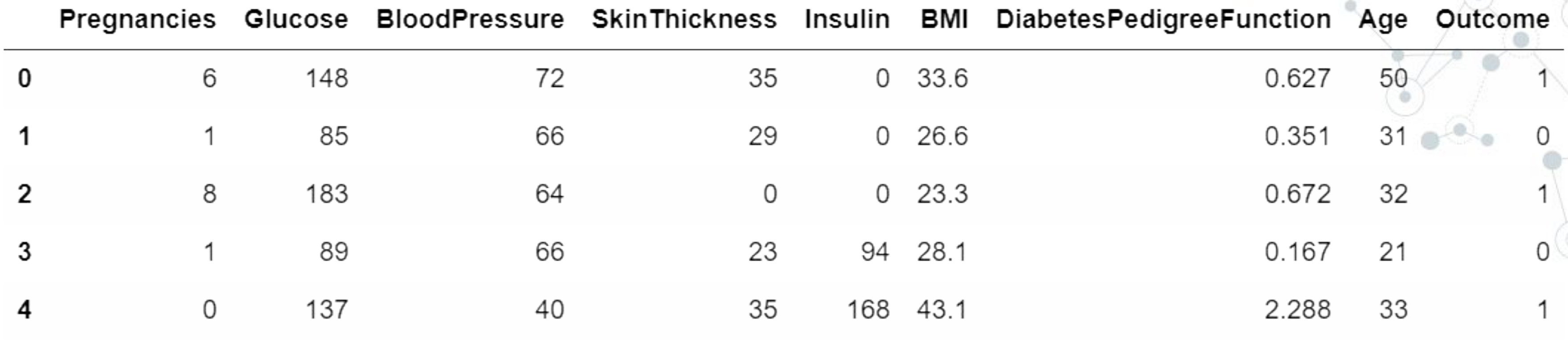
Finally, the best model is tuned in order to improve the final results

1.

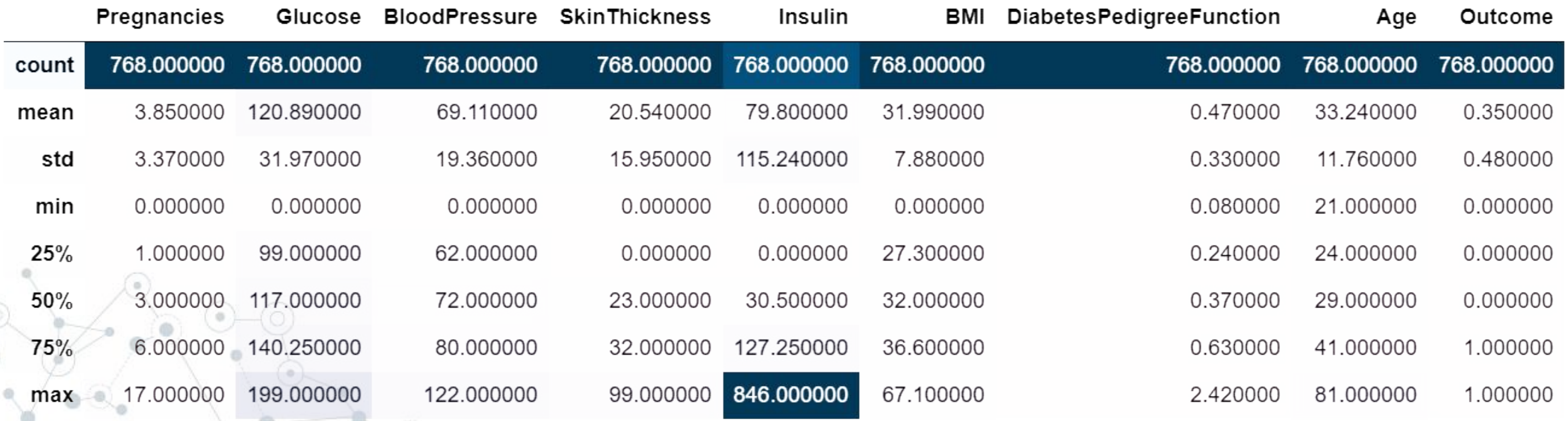
Dataset Exploration

Cleaning and Feature Analysis

Dataset Exploration

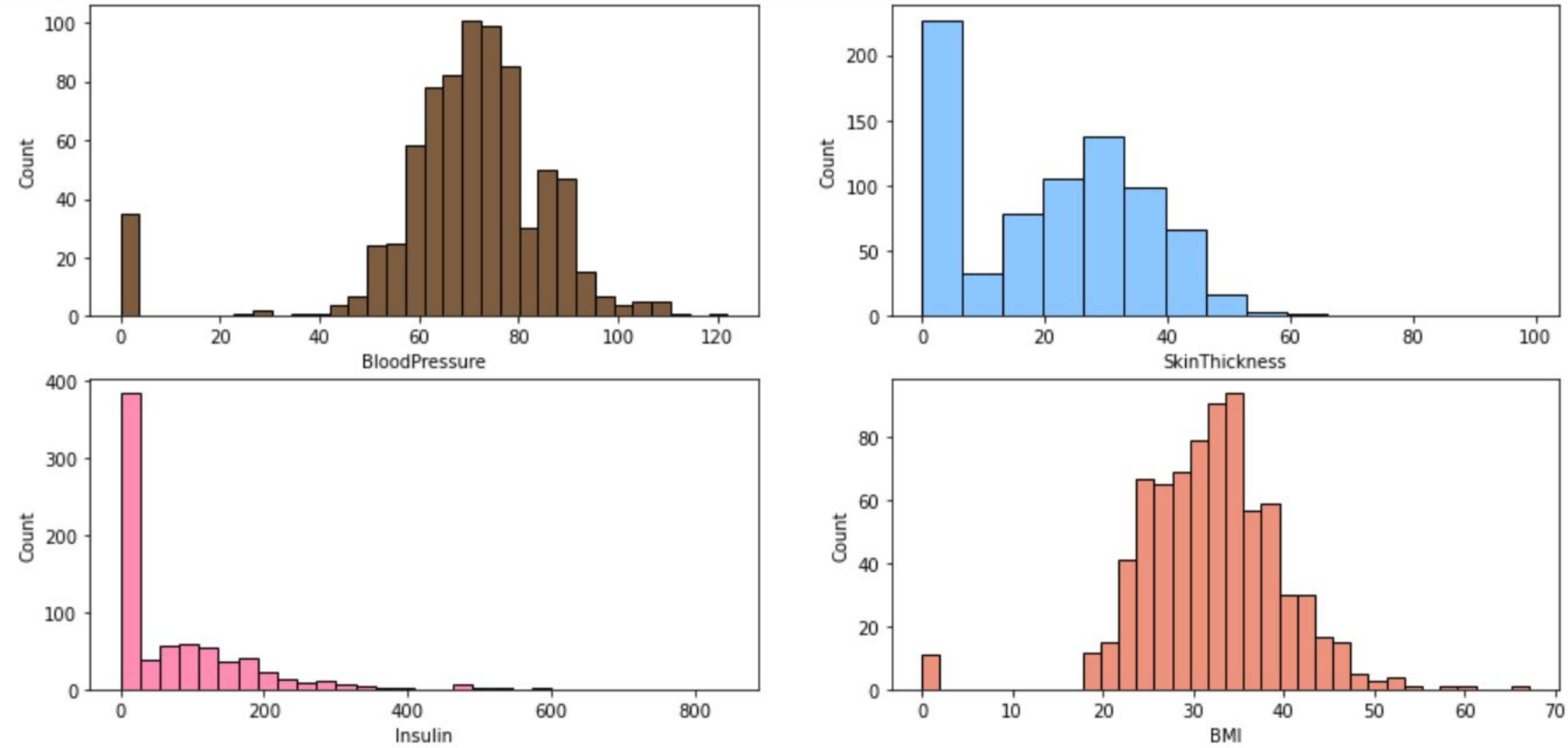


	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1



	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.850000	120.890000	69.110000	20.540000	79.800000	31.990000	0.470000	33.240000	0.350000
std	3.370000	31.970000	19.360000	15.950000	115.240000	7.880000	0.330000	11.760000	0.480000
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.080000	21.000000	0.000000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.240000	24.000000	0.000000
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	0.370000	29.000000	0.000000
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	0.630000	41.000000	1.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	81.000000	1.000000

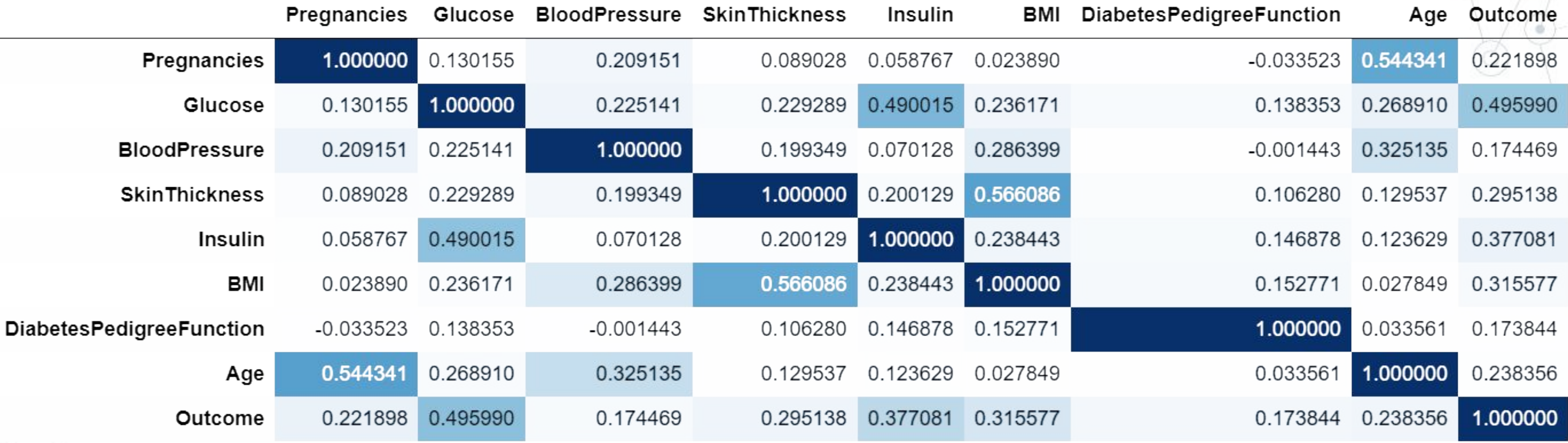
Dataset Cleaning



Feature Analysis

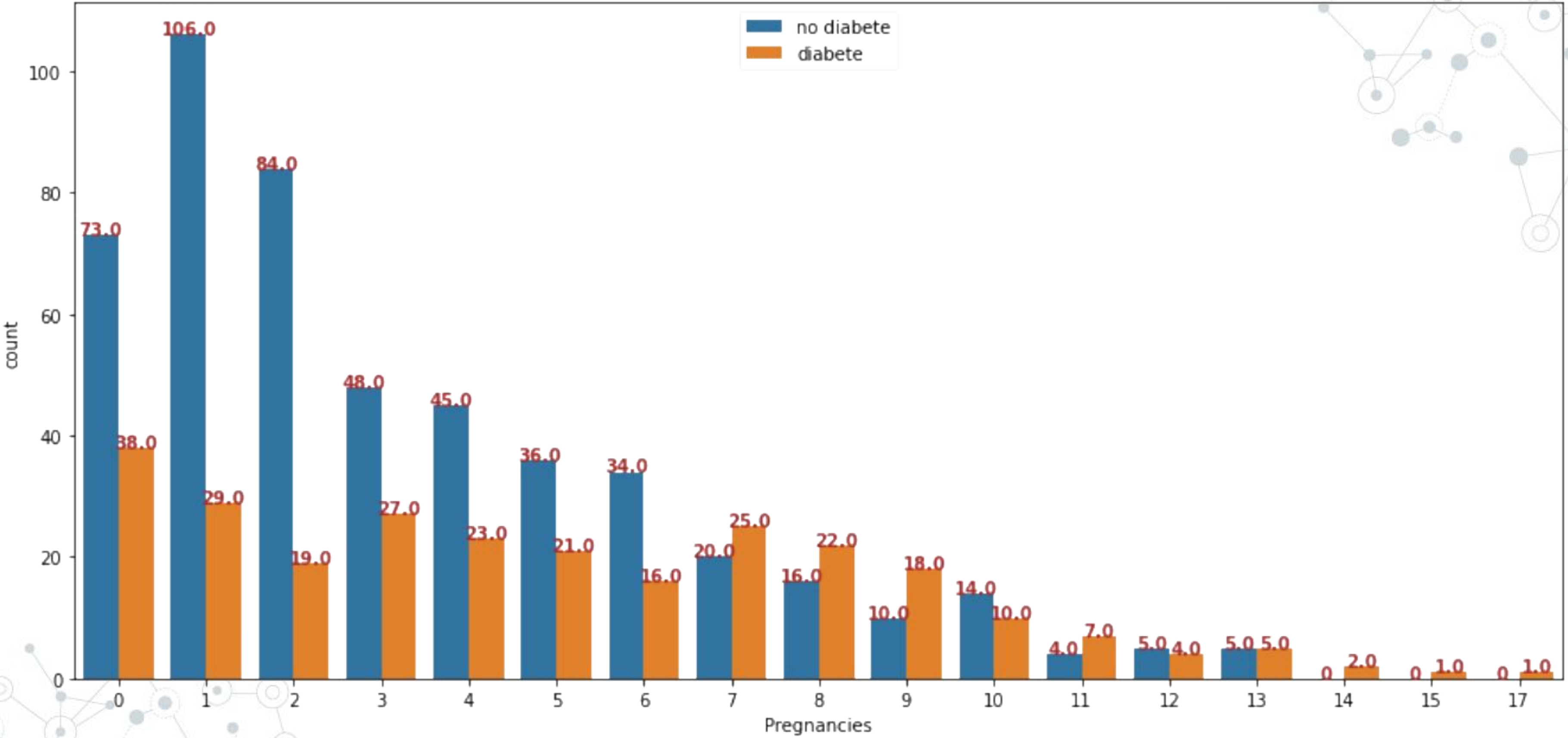
Some interesting correlations can be noticed, in particular observing the ***Outcome* column.**

These are reported in the following table of correlations:



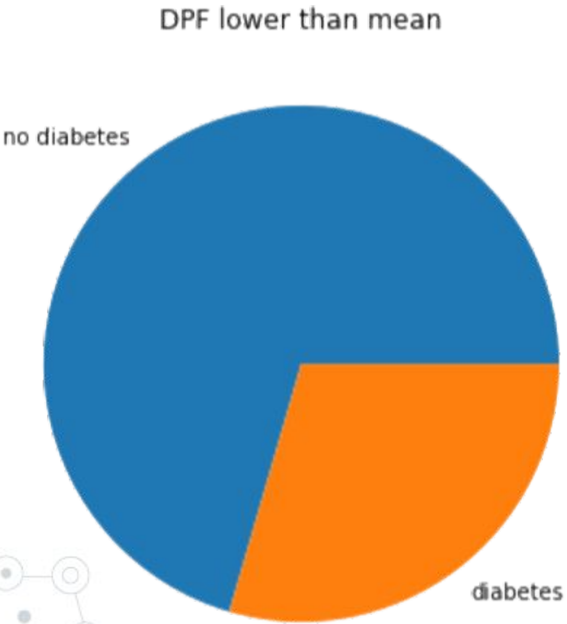
	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
Pregnancies	1.000000	0.130155	0.209151	0.089028	0.058767	0.023890	-0.033523	0.544341	0.221898
Glucose	0.130155	1.000000	0.225141	0.229289	0.490015	0.236171	0.138353	0.268910	0.495990
BloodPressure	0.209151	0.225141	1.000000	0.199349	0.070128	0.286399	-0.001443	0.325135	0.174469
SkinThickness	0.089028	0.229289	0.199349	1.000000	0.200129	0.566086	0.106280	0.129537	0.295138
Insulin	0.058767	0.490015	0.070128	0.200129	1.000000	0.238443	0.146878	0.123629	0.377081
BMI	0.023890	0.236171	0.286399	0.566086	0.238443	1.000000	0.152771	0.027849	0.315577
DiabetesPedigreeFunction	-0.033523	0.138353	-0.001443	0.106280	0.146878	0.152771	1.000000	0.033561	0.173844
Age	0.544341	0.268910	0.325135	0.129537	0.123629	0.027849	0.033561	1.000000	0.238356
Outcome	0.221898	0.495990	0.174469	0.295138	0.377081	0.315577	0.173844	0.238356	1.000000

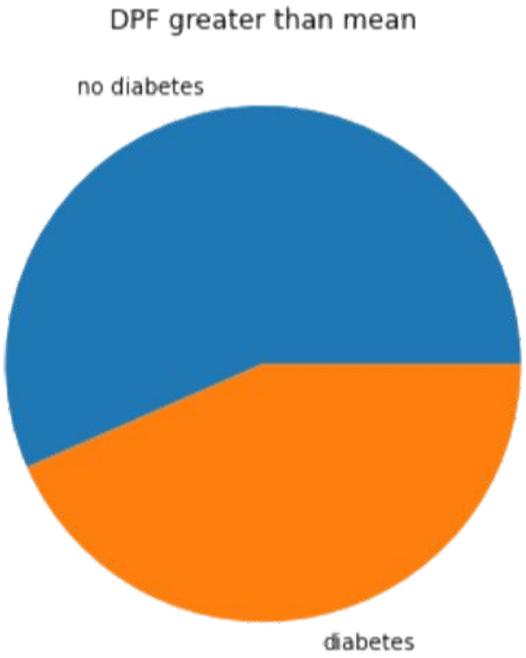
Pregnancies and Outcome



Diabetes Pedigree Function and Outcome

Diabetes Pedigree is the function which indicates likelihood of diabetes based on **family history**: these plots analyze the percentage of cases of diabetes for the classes which contains a value under or over the mean for this metric.



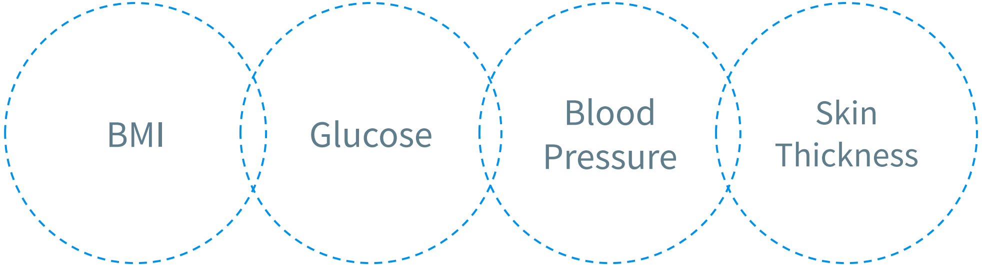


2.

Feature Engineering

Applying medical knowledge to collect more information

The Idea



BMI

Glucose

Blood Pressure

Skin Thickness

Classify Features

Explore and analyse medical academic articles in order to get a classification for these features.

Transform the new columns in matrices

Take the new categorical features and make them numerical through the creation of matrices, which contain a column for each class of each feature.

Use the new features in the *Outcome* Prediction

Append these new matrices to the dataset in order to exploit the information and achieve more accurate predictions.

Feature Engineering

BMI	
BMI <= 18.5	**Underweight**
18.5 < BMI <= 24.9	**Normal**
24.9 < BMI <= 29.9	**Overweight**
29.9 < BMI <= 34.9	**Obesity 1**
34.9 < BMI <= 39.9	**Obesity 2**
BMI > 39.9	**Obesity 3**

Glucose	
Glucose <= 140	**Normal**
140 < Glucose <= 199	**Prediabetes**
Glucose > 199	**Diabetes**

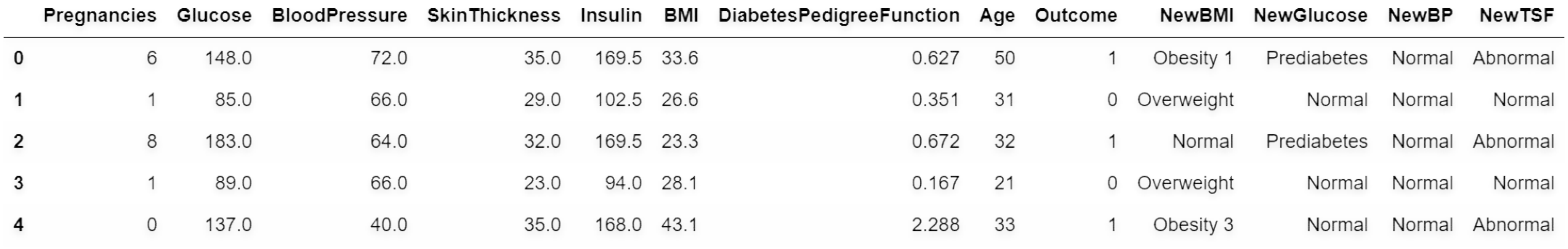
Feature Engineering

Blood Pressure	
BP <= 80	**Normal**
80 < BP <= 89	**Prehypertension**
BP > 89	**Hypertension**

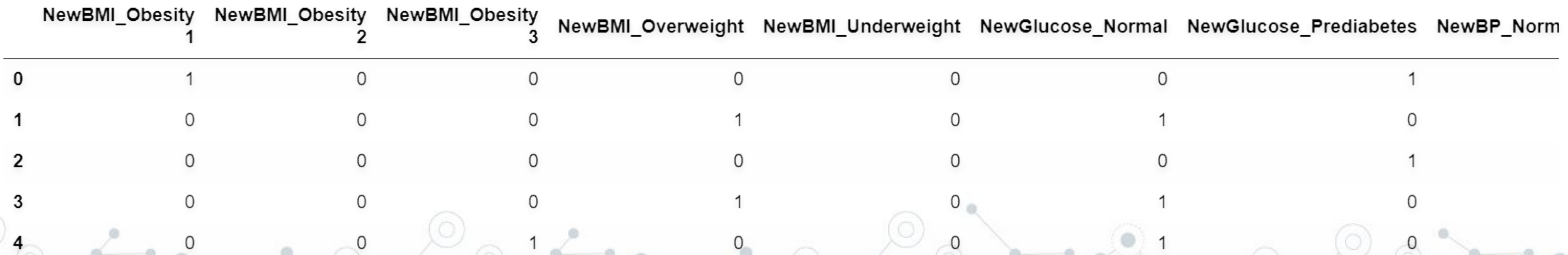
Skin Thickness	
16.1 <= TSF <= 31.1	**Normal**
TSF < 16.1 or TSF > 31.1	**Abnormal**

Feature Engineering

The new categorical features are then converted into matrices, in order to fit the ***RobustScaler*** function used in the dataset preprocessing before model application.



	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome	NewBMI	NewGlucose	NewBP	NewTSF
0	6	148.0	72.0	35.0	169.5	33.6	0.627	50	1	Obesity 1	Prediabetes	Normal	Abnormal
1	1	85.0	66.0	29.0	102.5	26.6	0.351	31	0	Overweight	Normal	Normal	Normal
2	8	183.0	64.0	32.0	169.5	23.3	0.672	32	1	Normal	Prediabetes	Normal	Abnormal
3	1	89.0	66.0	23.0	94.0	28.1	0.167	21	0	Overweight	Normal	Normal	Normal
4	0	137.0	40.0	35.0	168.0	43.1	2.288	33	1	Obesity 3	Normal	Normal	Abnormal



	NewBMI_Obesity 1	NewBMI_Obesity 2	NewBMI_Obesity 3	NewBMI_Overweight	NewBMI_Underweight	NewGlucose_Normal	NewGlucose_Prediabetes	NewBP_Norm
0	1	0	0	0	0	0	1	
1	0	0	0	1	0	1	0	
2	0	0	0	0	0	0	1	
3	0	0	0	1	0	1	0	
4	0	0	1	0	0	1	0	

3.

Outcome Prediction

The application and tune of
Supervised Learning algorithms

Supervised Learning Algorithms

Logistic Regression

is a model for analyzing a dataset in which there are one or more independent variables that determine an outcome.

Random Forest

is a model that uses multiple decision trees and aggregates their individual predictions to produce a final output.

KNeighbors Classifier

is an algorithm that classifies data points based on their proximity to their k nearest neighbors in the feature space.

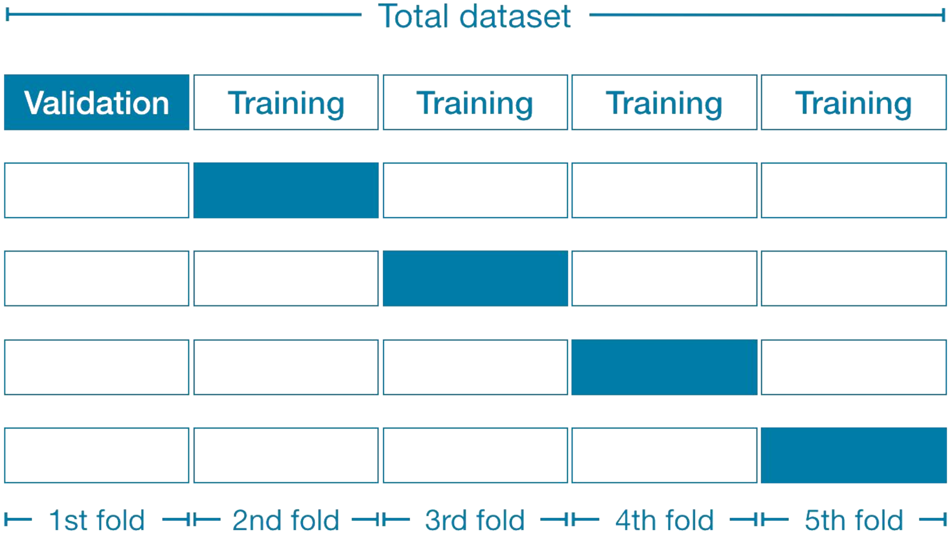
SVM

is an algorithm that classifies data by finding the optimal hyperplane that maximally separates the data into classes.

Cross Validation

Splits the data in *k* bins and runs *k* separate experiments, where each:

- Picks a bin as validation set
- Uses the other bins as training set
- Trains the model



Total dataset

Validation Training Training Training Training

1st fold 2nd fold 3rd fold 4th fold 5th fold

Then collects the chosen metrics as **the average** of the results from those *k* experiments.

Chosen Metrics

Balanced Accuracy

Is similar to accuracy, but it takes into account the imbalance in the dataset by calculating the average of recall for each class.

Recall

it is the number of true positive predictions divided by the sum of the true positive predictions and false negative predictions.

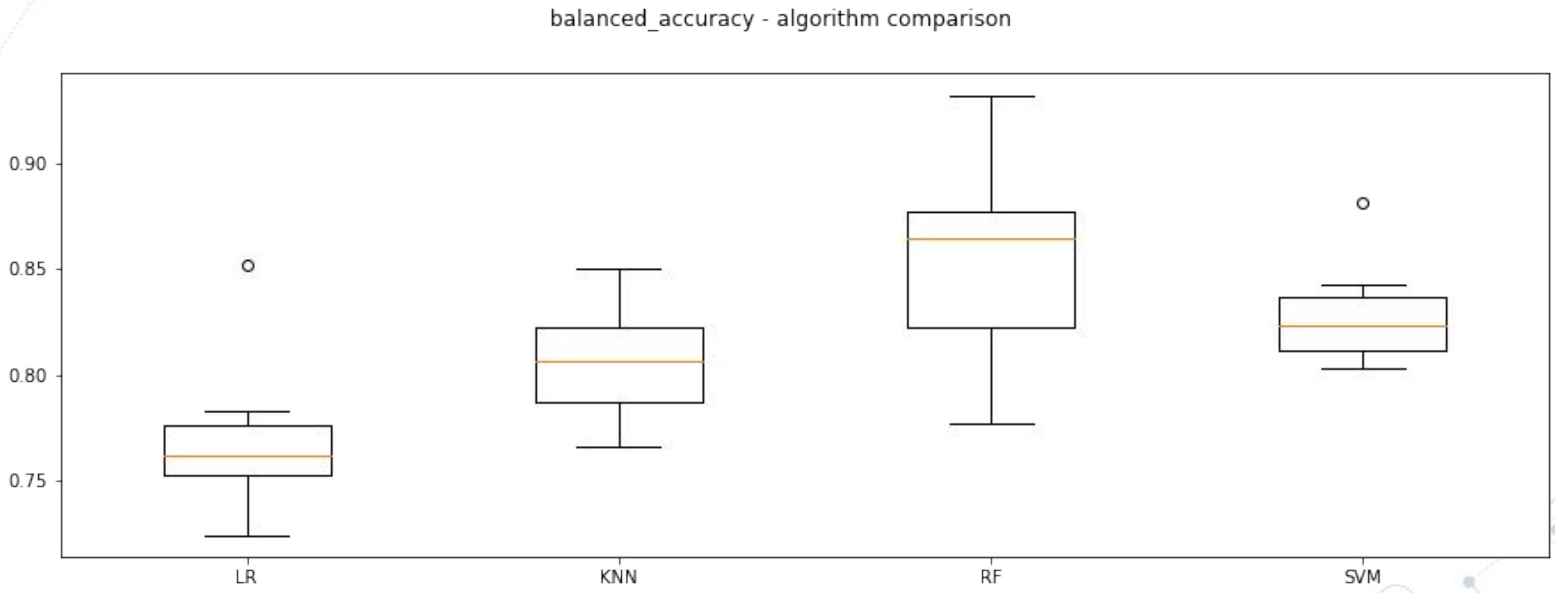
ROC/AUC

It indicates how much the model is able to distinguishing between positive and negative cases.

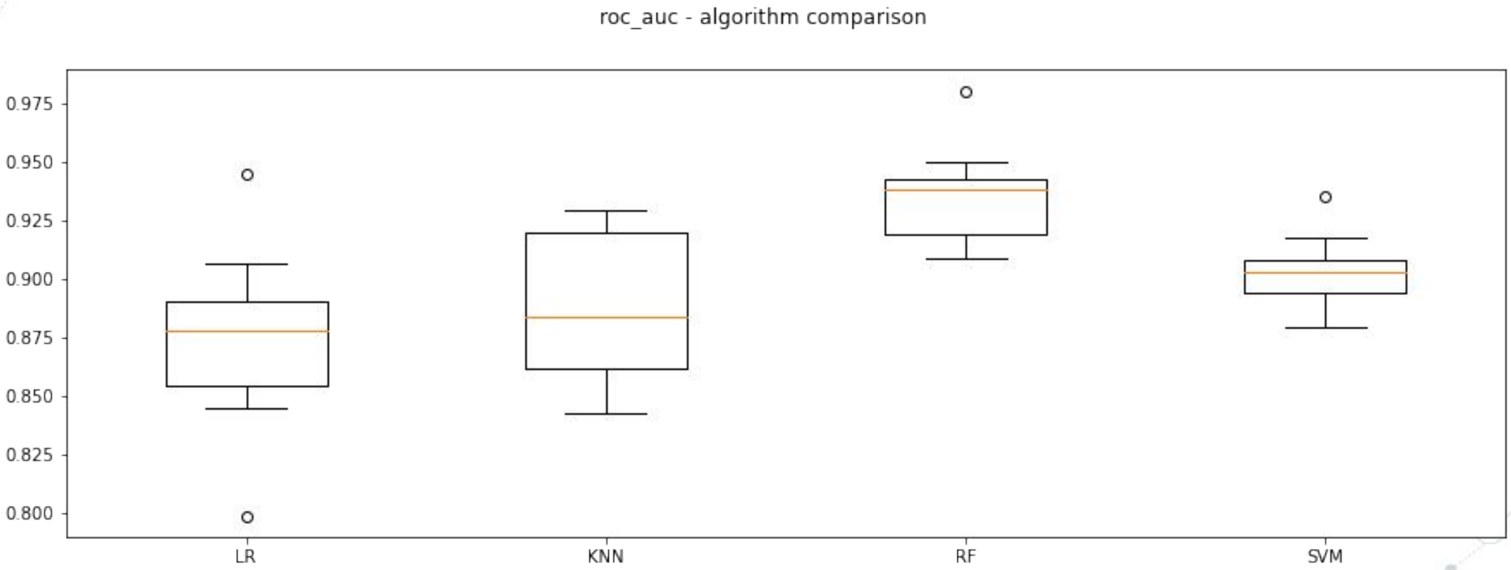
F1

Is the harmonic mean of precision and recall, often used when data is unbalanced.

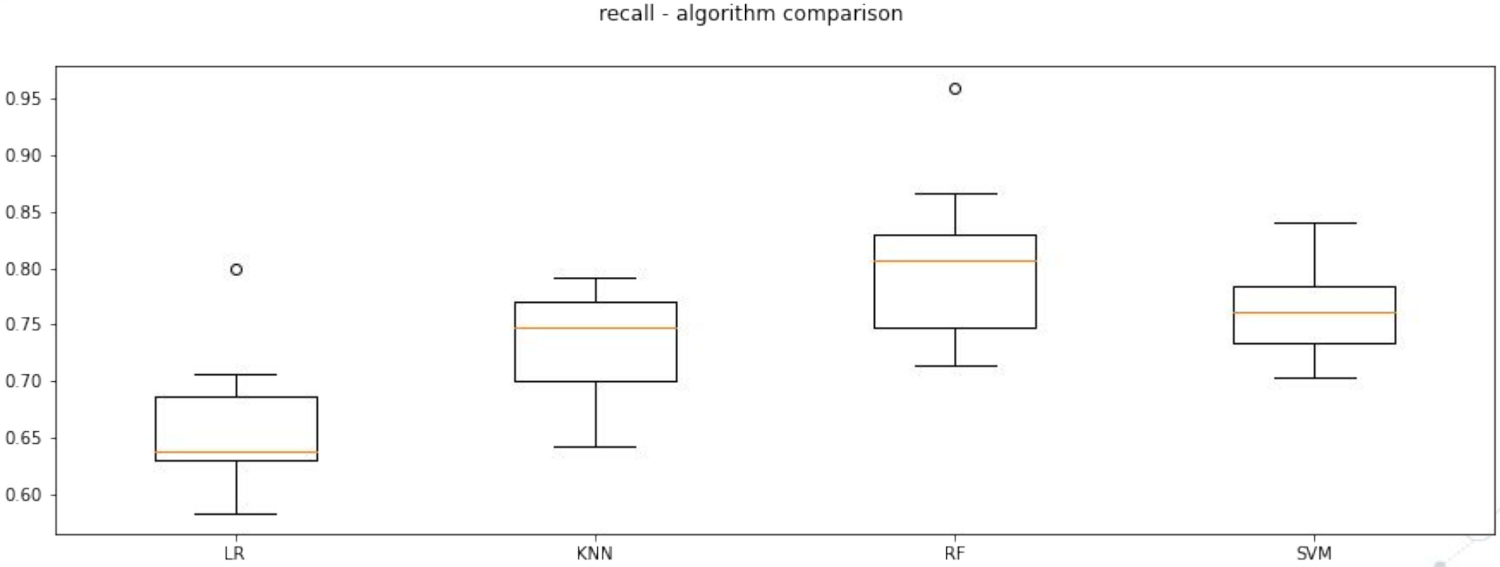
Balanced Accuracy



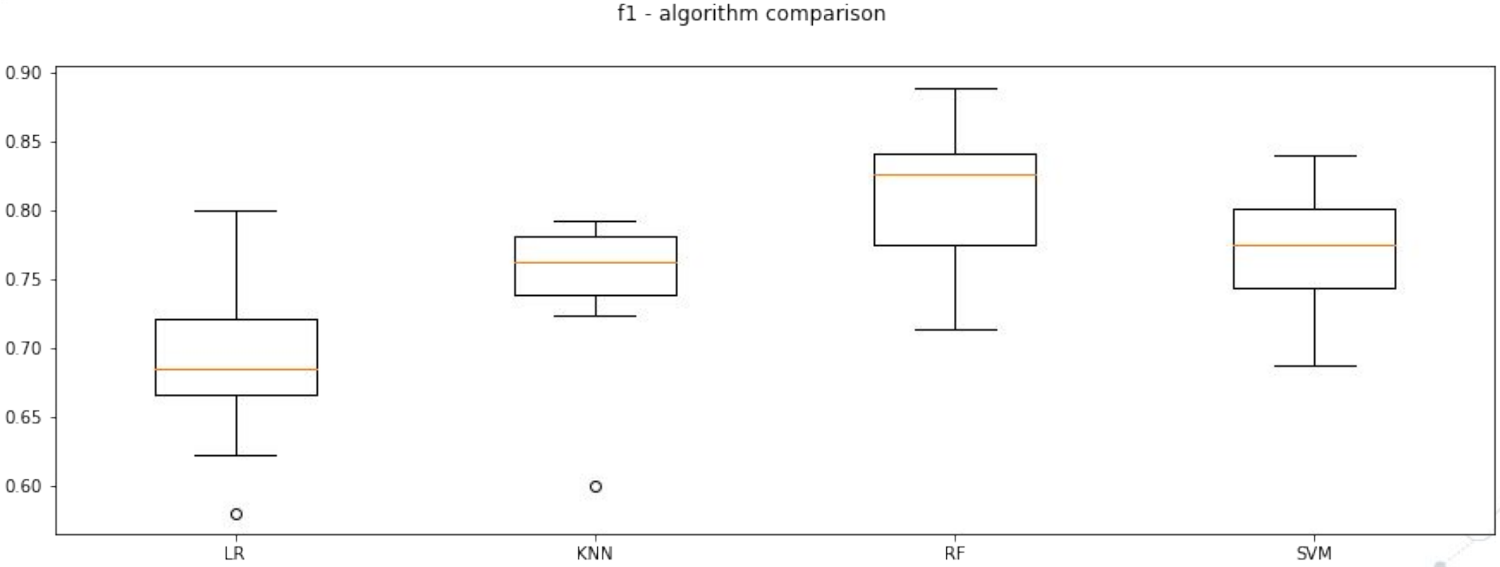
ROC/AUC


roc_auc - algorithm comparison
0.975
0.950
0.925
0.900
0.875
0.850
0.825
0.800
LR
KNN
RF
SVM

Recall


recall - algorithm comparison
0.95
0.90
0.85
0.80
0.75
0.70
0.65
0.60
LR
KNN
RF
SVM

F1


f1 - algorithm comparison
0.90
0.85
0.80
0.75
0.70
0.65
0.60
LR
KNN
RF
SVM

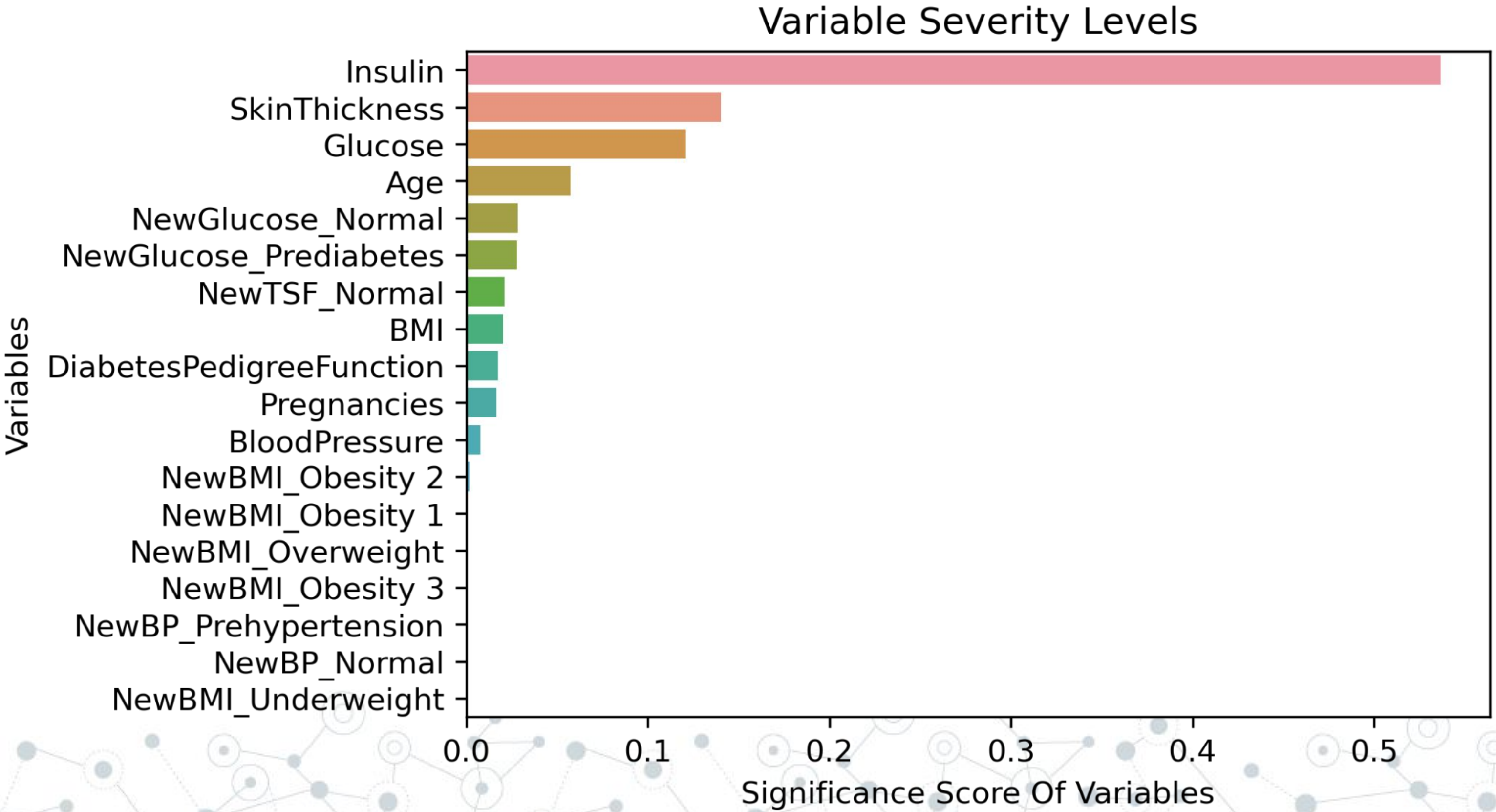
Random Forest Tuning

Hyperparameter	Description	Possible Values	Selected Value
n_estimators	number of trees in the forest	[100, 500, 1000]	100
max_features	max number of features considered for splitting a node	[2, 5, 7]	7
min_samples_split	minimum number of data points placed in a node before it’s split	[2, 5, 10]	2
max_depth	max number of levels in each decision tree	[None, 5, 25]	None
min_samples_leaf	minimum number of data points allowed in a leaf node	[1, 5, 15]	5

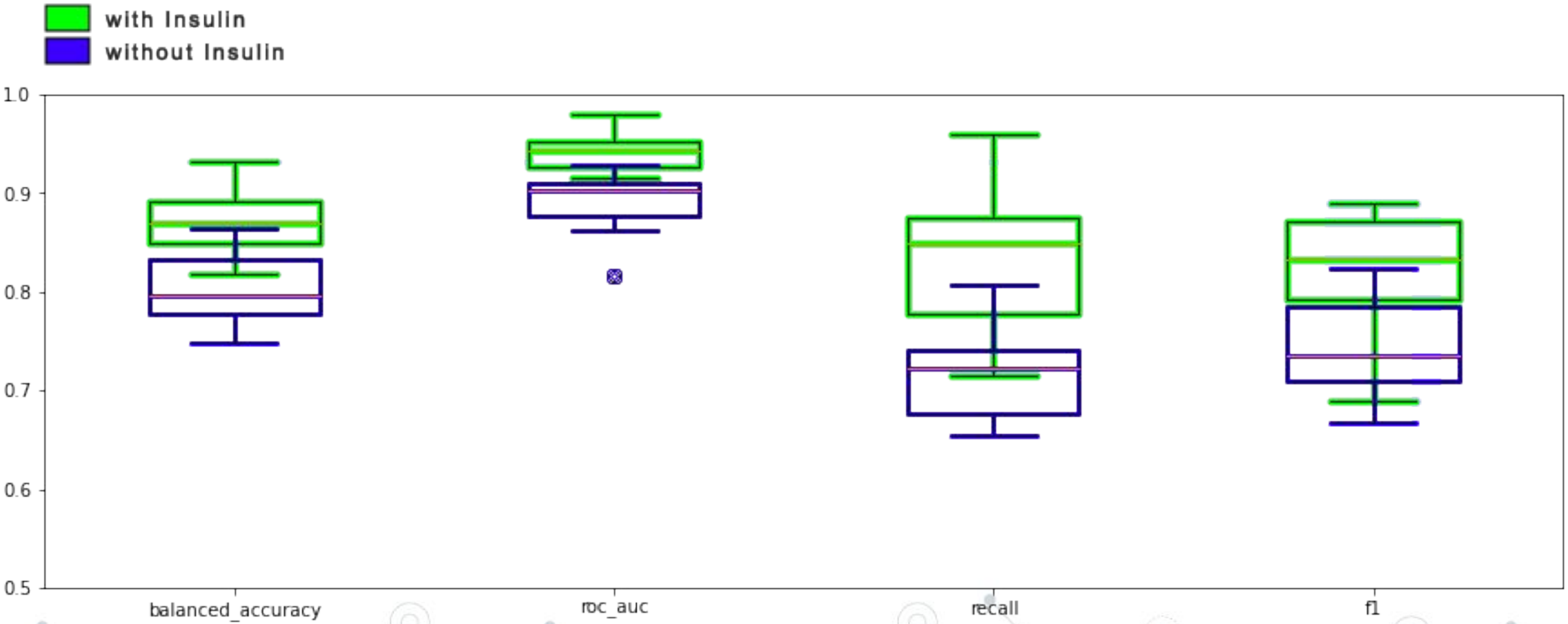
Final Results

Metric	Score Before Tuning	Score After Tuning	Improvement
Accuracy	0.853605	0.862414	0.008809
ROC/AUC	0.935069	0.945228	0.010159
Recall	0.804767	0.811582	0.006815
F1	0.808100	0.818264	0.010164

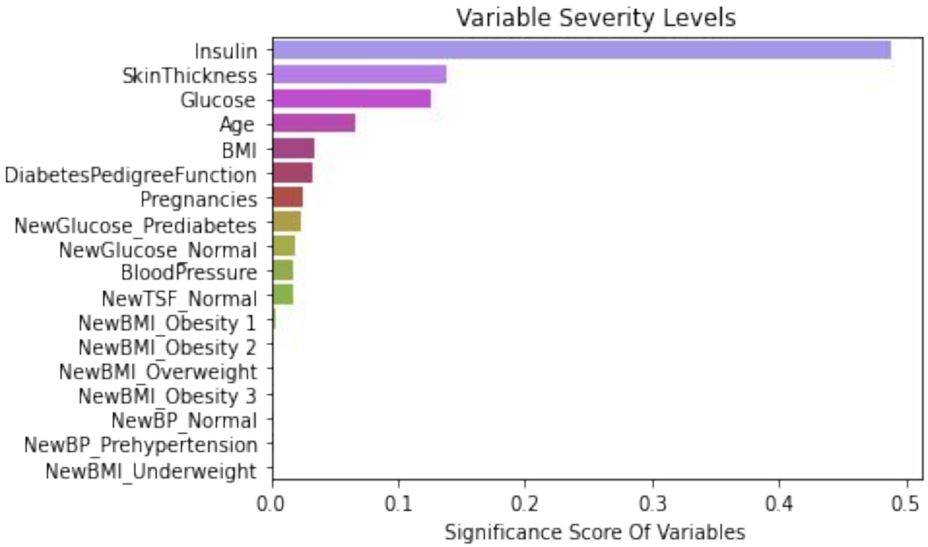
Variable Severity Levels

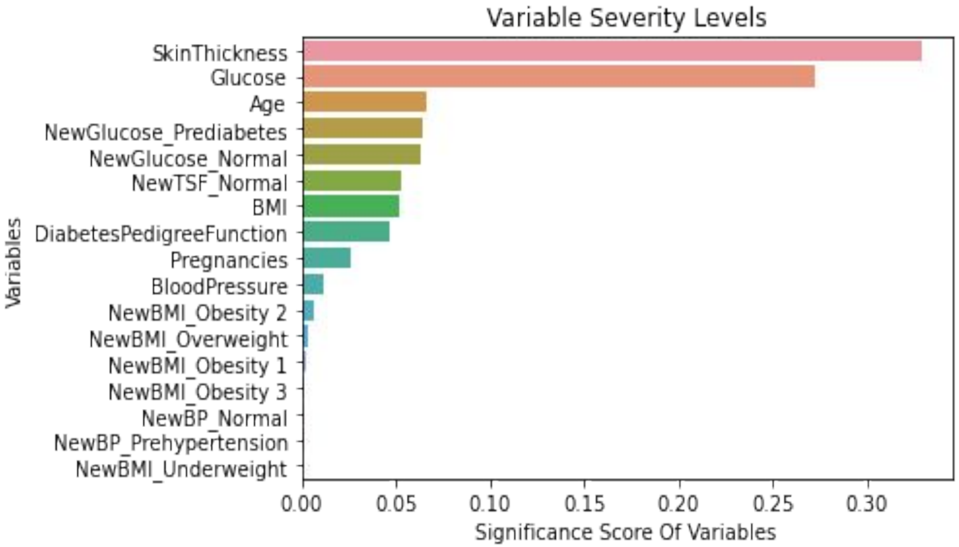


Final Results with and without Insulin



Severity Levels with and without Insulin


Variable Severity Levels


Variable Severity Levels

Thanks!

Lorenzo La Corte

STUDENT

S4784539@studenti.unige.it