Capstone 2 - Report

Heart Disease Risk Prediction using Framingham Data

Introduction

Heart disease remains a leading cause of mortality and a significant public health concern worldwide, with its development influenced by a multitude of risk factors. Its onset is influenced by a multitude of risk factors.

Epidemiology

USA

- According to the Centers for Disease Control and Prevention (CDC), heart disease is the leading cause of death for both men and women in the United States.
- In 2020, heart disease accounted for about 23% of all deaths in the US, with approximately 696,962 deaths attributed to heart disease.
- About 6.7% of adults aged 18 years and older (approximately 17.9 million people) have coronary heart disease, the most common type of heart disease.
- Each year, about 805,000 Americans have a heart attack, and about 655,000 Americans die from heart disease-related complications.

Worldwide

- The World Health Organization (WHO) reports that cardiovascular diseases (CVDs), including heart disease and stroke, are the leading cause of death globally.
- In 2019, an estimated 17.9 million people died from CVDs worldwide, representing 32% of all global deaths.
- Approximately 85% of these deaths are due to heart attacks and strokes.
- CVDs affect people of all ages and backgrounds, with the burden being particularly high in low- and middle-income countries.

The Framingham Study aims to address the pressing issue of heart disease, which remains the leading cause of death both in the United States and globally. In the US alone, nearly 700,000 deaths are attributed to heart disease annually, with millions more living with coronary heart disease. Globally, cardiovascular diseases claim the lives of almost 18 million people each year, with the burden disproportionately affecting low- and middle-income countries. Within this context, it is imperative to identify individuals at risk of developing heart disease within the next decade, as early detection allows for timely intervention and prevention strategies. By understanding the factors contributing to heart disease risk, the Framingham Study seeks to provide actionable insights that can inform targeted interventions and improve public health outcomes.

This project aims to create a strong screening model that effectively spots individuals with a high risk of developing heart disease within the next 10 years, while keeping false negatives to a minimum. Early detection is key for timely intervention. Like with any medical screening, we want to catch as many at-risk patients as possible, even if it means some false positives. We also aim to maximize true negatives and minimize false negatives to ensure we don't miss anyone who might be at risk. It's important to strike a balance so that the model doesn't raise too many false alarms, which could undermine trust from both patients and providers. Additionally, the model's ability to identify the most significant predictors of heart disease can help guide healthcare professionals in targeting their preventive efforts effectively.

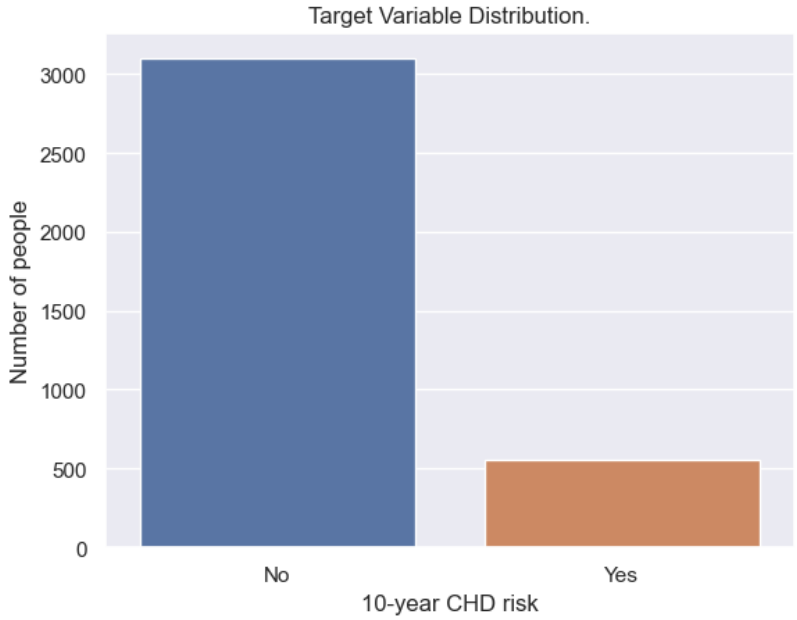
Data Overview

The primary data source for this project will be the Framingham Heart Study dataset, available for research purposes. This dataset includes demographic, behavioral, and medical risk factor information collected from participants over several decades, offering a comprehensive view of the risk factors associated with heart disease.

Features of the data: - Demographic: Patient's age and sex - Behavioral features: smoking status, number of cigarettes per day - Medical history: blood pressure medications, hypertension, diabetes, stroke history - Test results: total cholesterol and glucose concentrations, systolic and diastolic blood pressure, heart rate and body mass index. - Target feature: 10-year risk of coronary heart disease CHD

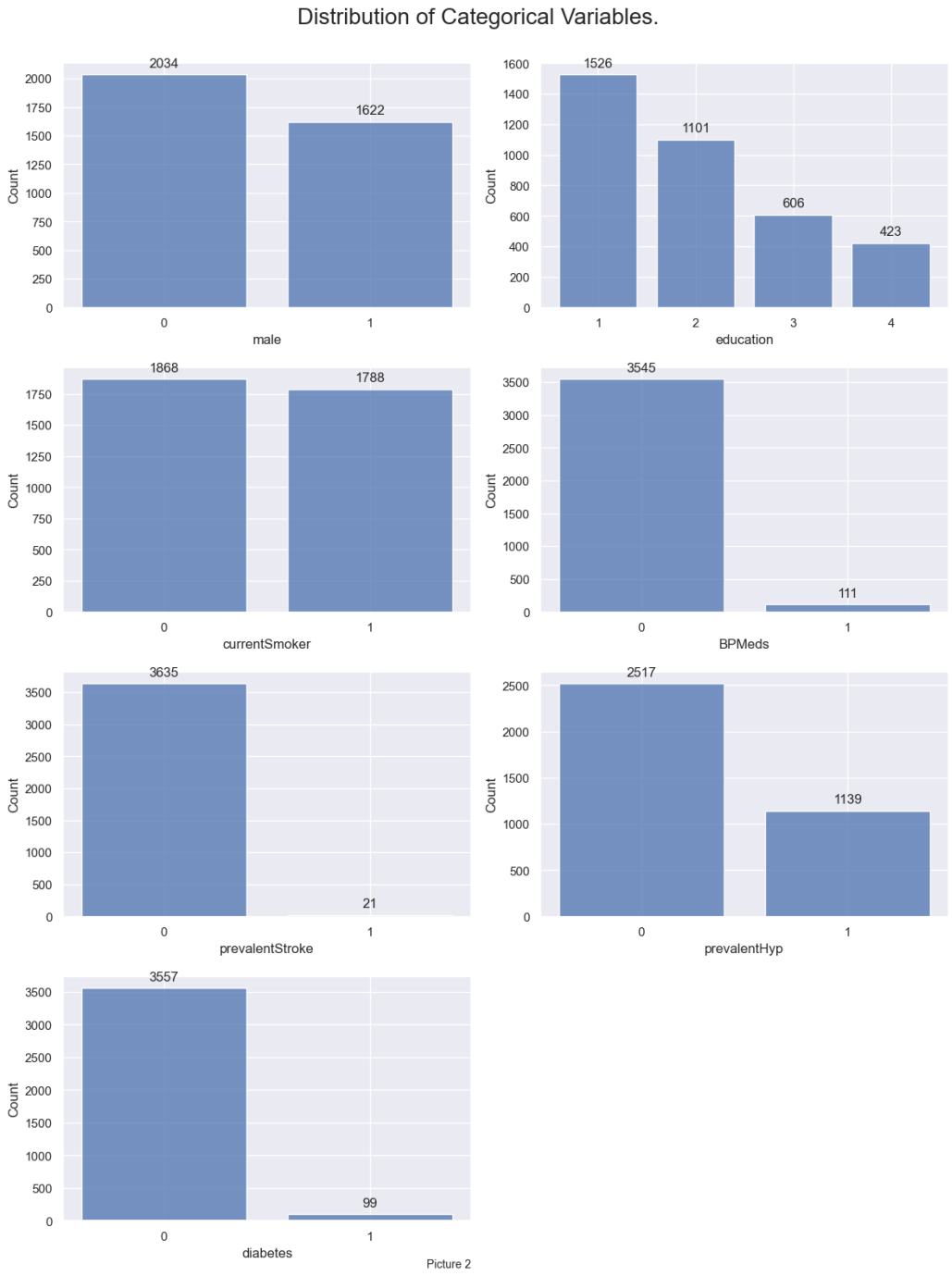
Exploratory Data Analysis

First, let's see the distribution of the target variable TenYearCHD, which is a patient's risk of having a heart disease within the next 10 years.



Picture 1

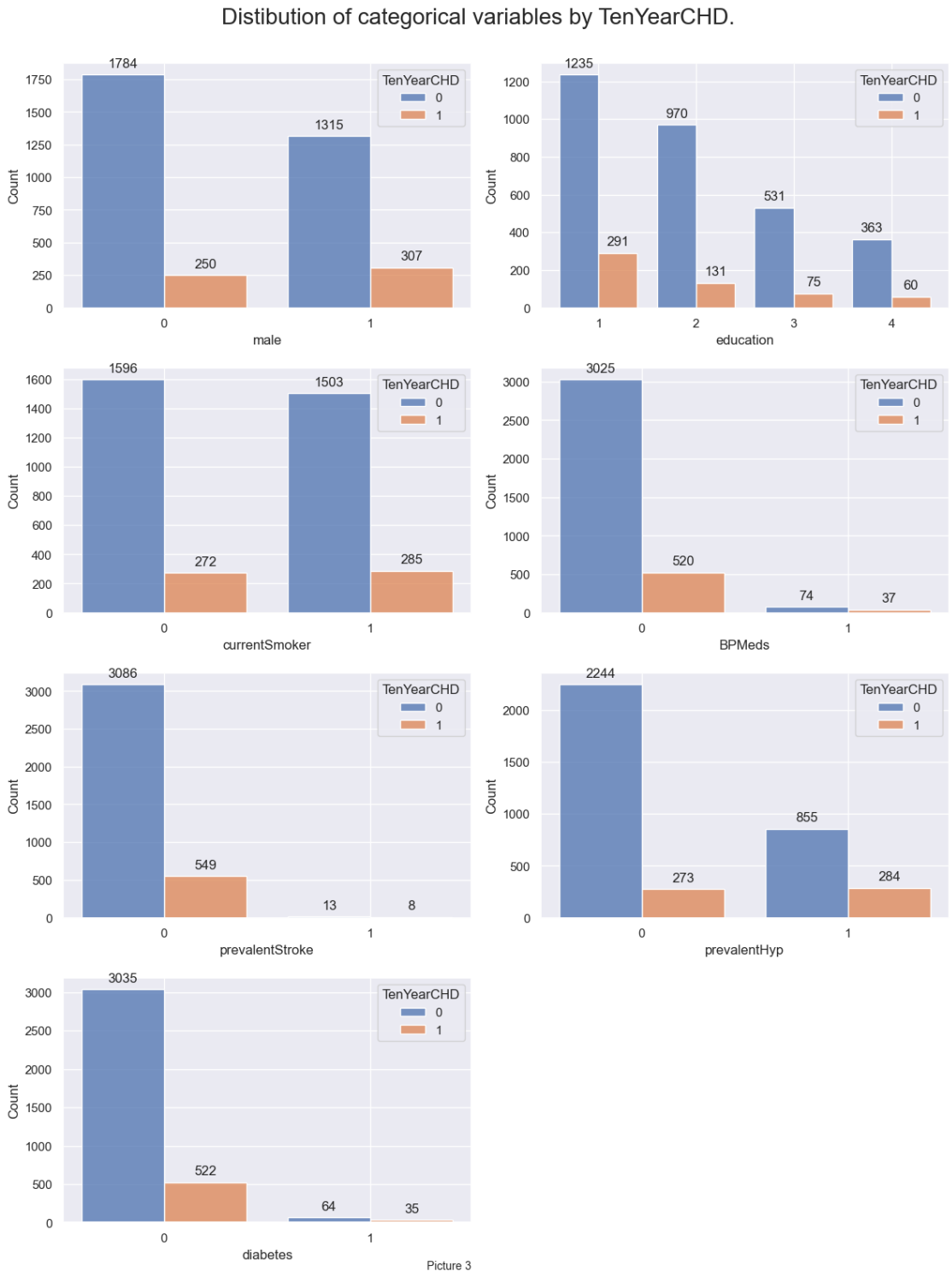
There is a high imbalance within our target value with ratio of 5.6 / 1 (No Risk / Risk). The data will have to be balanced before training models or parameters for balanced class weights should be used when configuring the model.



Picture 2

Out of all categorical columns the most balanced ones are sex (male) and smoking status (currentSmoker). Education column shows the expected distribution of education levels with more people at level 1, gradually degreasing with each step up. Prevalent hypertension column shows that approximately 1/3 of all patients (1139) have diagnosis of hypertension, but only 111 out of them take any blood pressure medications. In the diabetes column there are only 99 patients with established diagnosis. This column can technically be compared to the blood glucose The prevalentStroke column has with only 21 positive patients) cannot be compared with any other column.

Let’s see how the target value varies within groups of these categorical values.



Picture 3

Target value categorical influencers.

1. Being male seems to increase the risk.

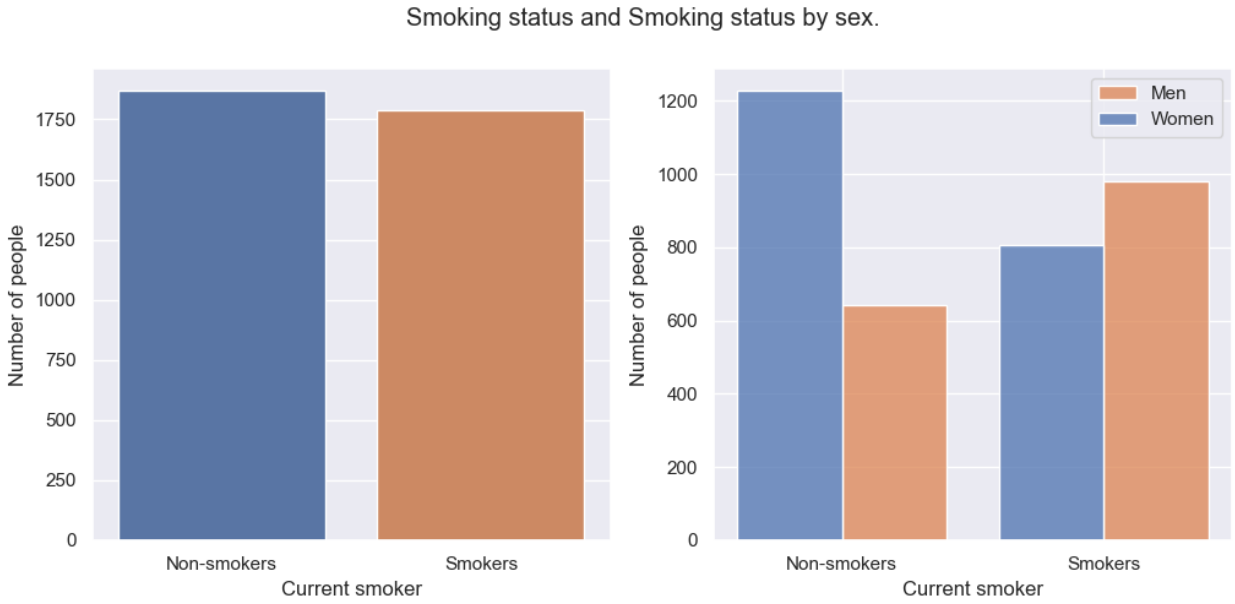
2. Education does not seem to have any influence of target value.

3. Being a smoker increases the risk, but not much really.

4. BP Medications, Prevalent Stroke and Diabetes have not enough positive entries to draw any reasonable conclusion (might need to engineer new features in these categories).

5. Prevalent hypertension, although being quite misbalanced, seems to play a role in increasing the risk (might also engineer a new feature for this category). Statistically (especially in the second half of 20th century) men tend to smoke more than women.

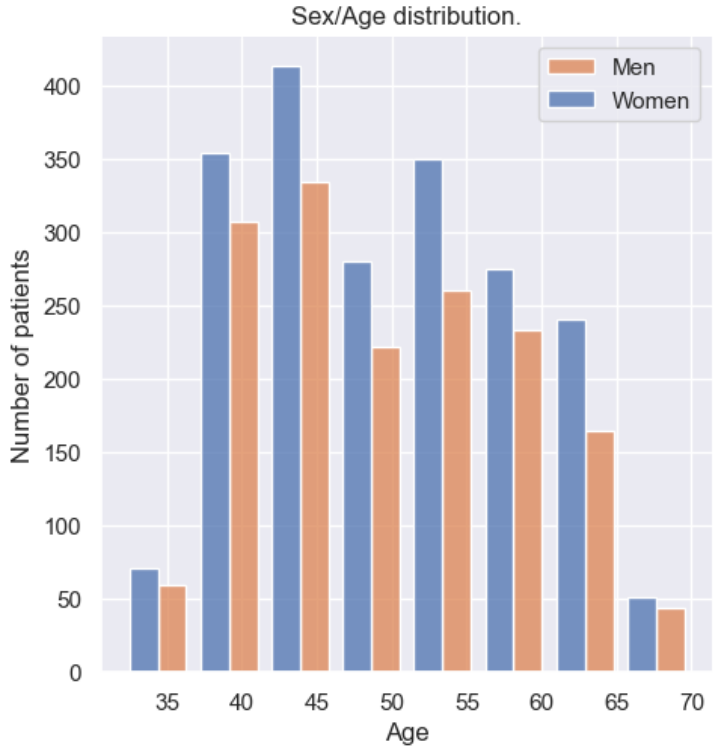
Let's dig deeper in the smoking factor and age/sex distributions to determine possible confounders.



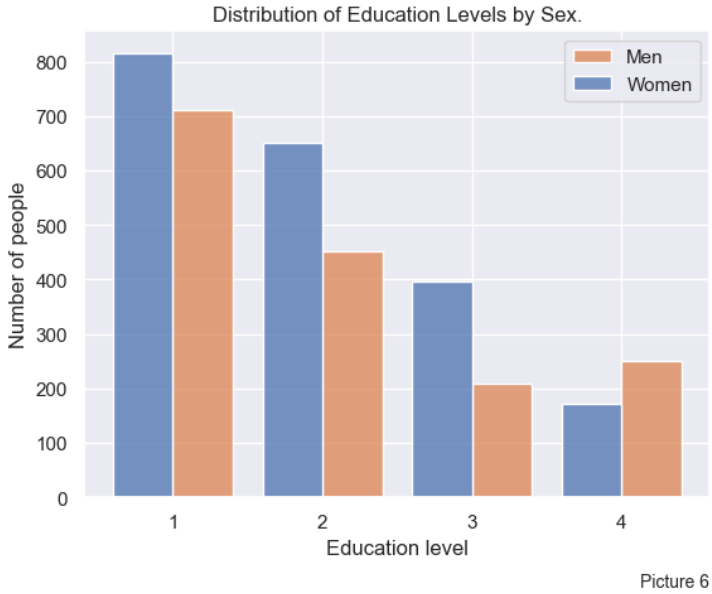
Picture 4

As seen on the Picture 4 most smokers are indeed male. And among all men, there are more smokers than non-smokers, where only half of all women tend to smoke. Could it be somehow related to age or education? Let's see sex/age distribution.

Most patients are women within all age groups as seen on the Picture 5.



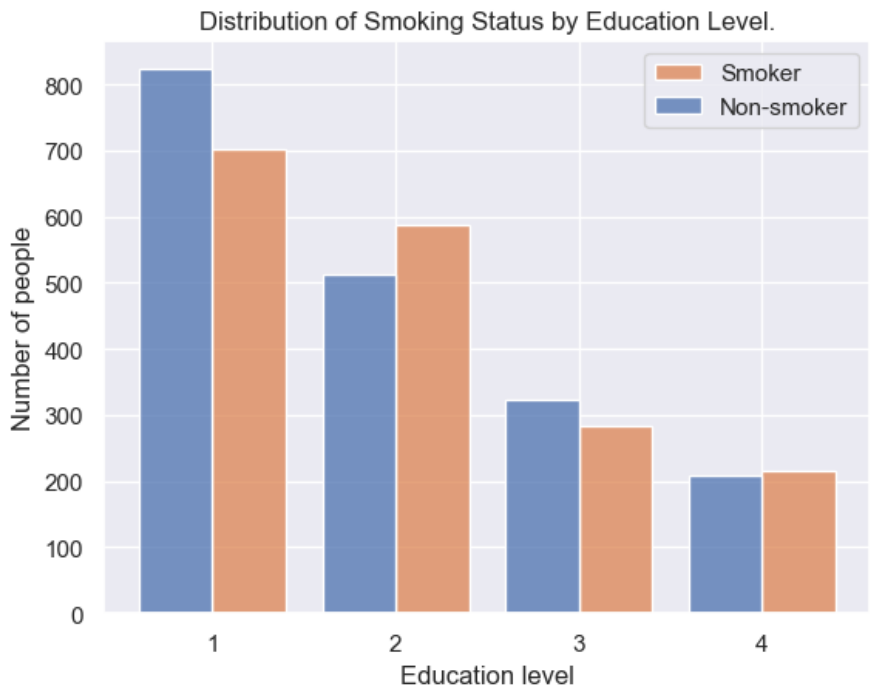
Picture 5



Picture 6

As seen on Picture 6, women are more prevalent within education levels 1-3, which reflects the overall sex distribution of the dataset. However there are nuances. Within the Level 3 group, the number of women is almost a double of that of men. Also, within the Level 4 group there are more men than women.

Can smoking status change with education level?



Picture 7

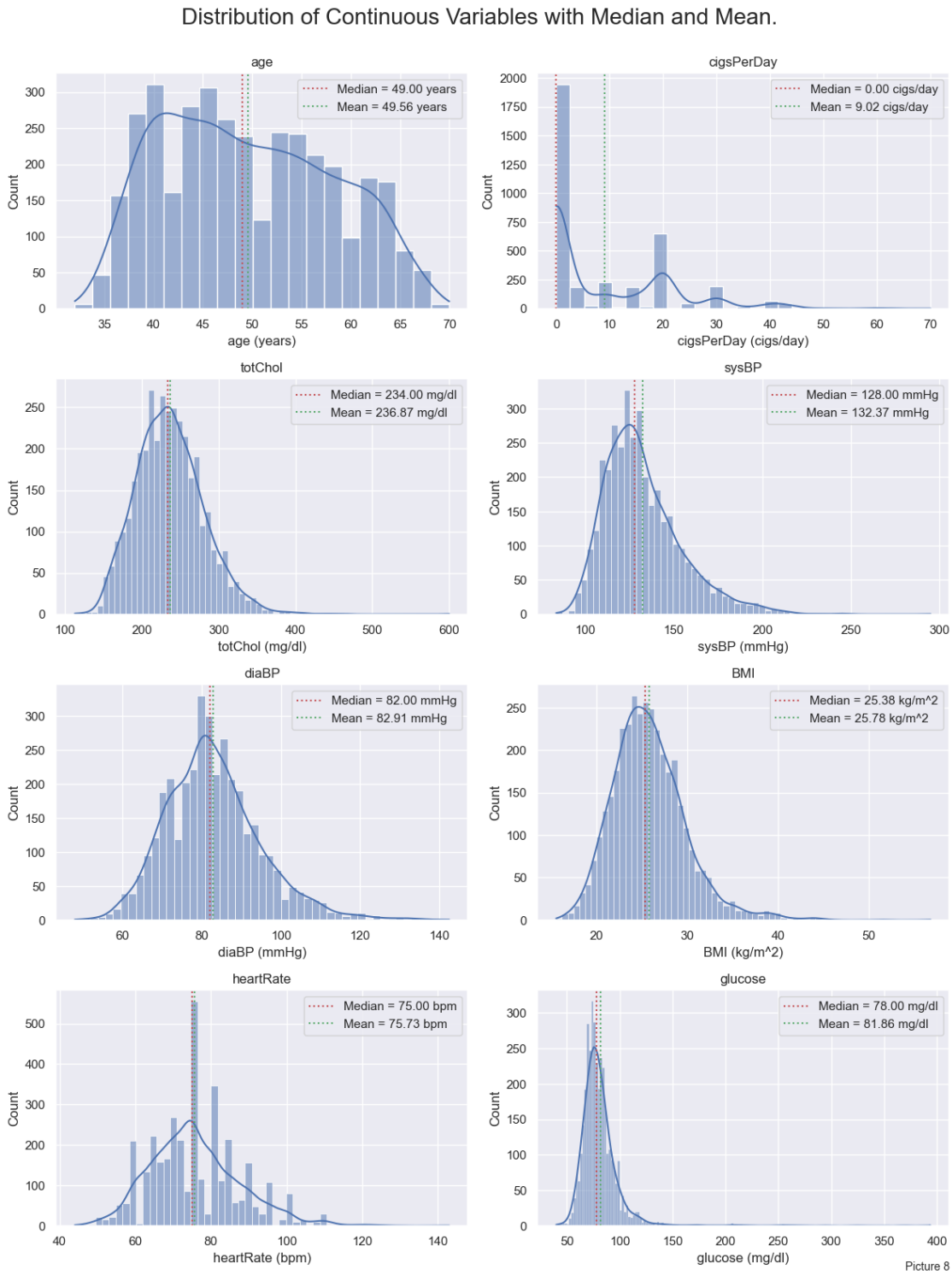
On the Picture 7 we see a very interesting tendency. The largest portion of non-smokers is within an education level 1 group. Level 2 shows the opposite pattern, with smokers being more prevalent. Level 3 shows similar ratio as level 1. And there are slightly more smokers than non-smokers in the level 4 group.

Let's investigate the relative numbers of smokers for these education groups and find out if they somehow depend on sex.

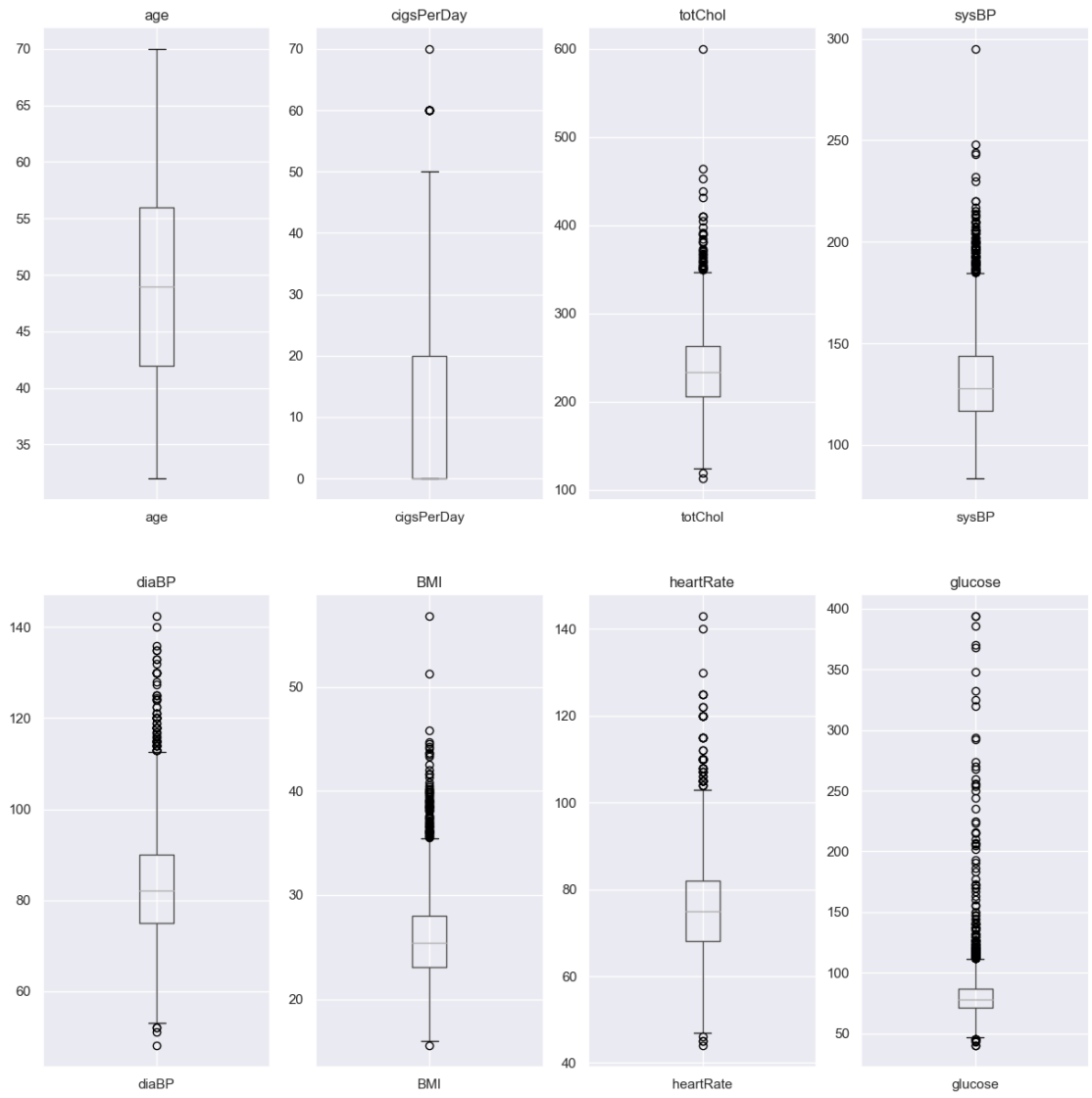
	Smoking Men	Non-Smoking Men	Smoking Women	Non-Smoking Women
Education 1	62%	38%	32%	68%
Education 2	64%	36%	46%	54%
Education 3	56%	44%	42%	58%
Education 4	54%	46%	47%	53%

Men tend to smoke more than women in all education groups. In level 1 only 1/3 of men don't smoke, compared to 2/3 of non-smoking women. In the level 2 group the men's ratio is the same, but there is a drastic increase in smoking among women, with almost 50% smoking. In groups 3 and 4 the ratio of smokers and non-smokers is roughly 1/1, but there is slightly more smoking men and slightly more non-smoking women.

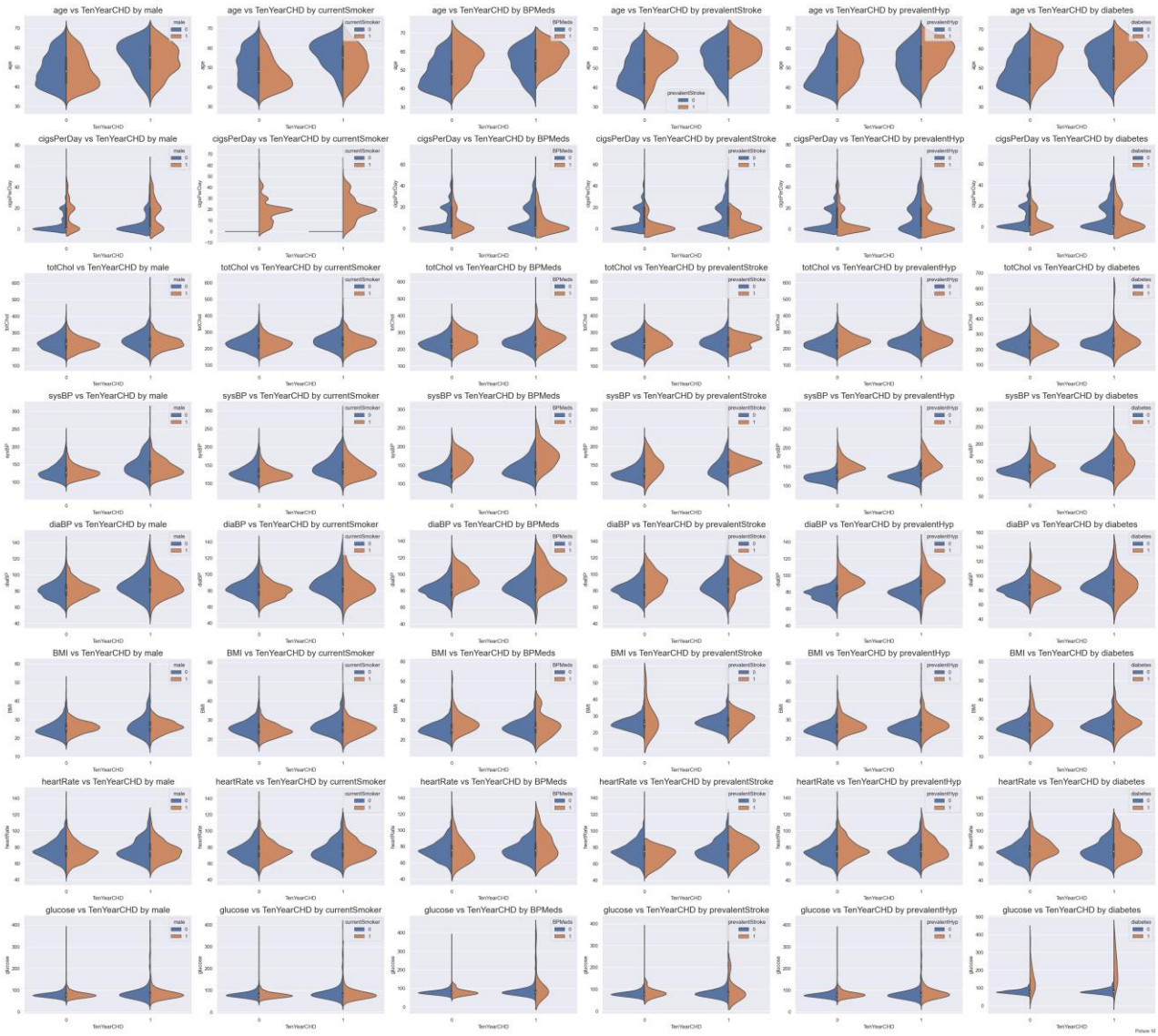
Continuous variables



Picture 8


Box Plots of Continuous Variables
age
cigsPerDay
totChol
sysBP
diaBP
BMI
heartRate
glucose

Picture 9



There are several conclusions that can be made from the Picture 8, 9 and 10:

- Age is one of the strongest factors influencing the 10-year risk of CHD.
- As well as being male.
- Smoking also contributes to 10-year CHD as well and the number of cigarettes per day.
- Blood pressure and blood glucose may also be contributing factors. Although, there may be a hypo-diagnosis of diabetes and hypertension in this population.

So, we might need to categorize these values into the stages of hyperglycemia and elevated blood pressure.

Side notes: Many features have interesting but expected behaviors, shifting distributions if they are present:

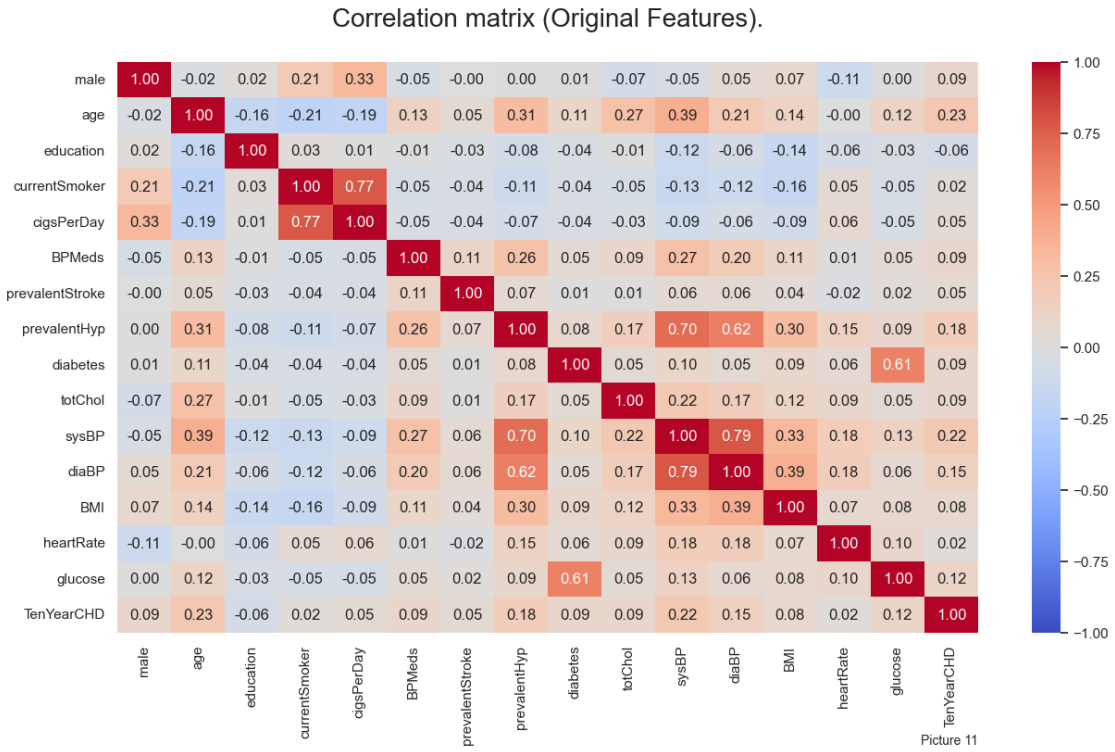
- Blood pressures tend to be higher in patients that are taking BP medications. Also, patients with higher BP tend to be already taking BP meds.
- Diabetes flattens some distributions (except BMI, which is interesting, because higher BMI values usually linked to diabetes)

Data Preprocessing

During the data preprocessing step several aspects of data should be addressed. One of them is handling missing values. There are two main approaches to it: dropping missing values and imputing missing values. Both techniques were used, resulting in two datasets: with all missing values dropped and with some of them dropped and some imputed. The goal is to leave as much valuable data as possible, especially for the relatively small datasets as this one (only 4238 rows originally). After dropping all missing values what we are left with are 3656 rows, and 582 rows dropped (13.7%). Imputation of missing data consisted of different approaches for continuous and categorical features (mean and mode respectively). We will see if this imputation gives the model any performance boost.

Feature Engineering

Now, let's see if there is any collinearity among the variables in the data.

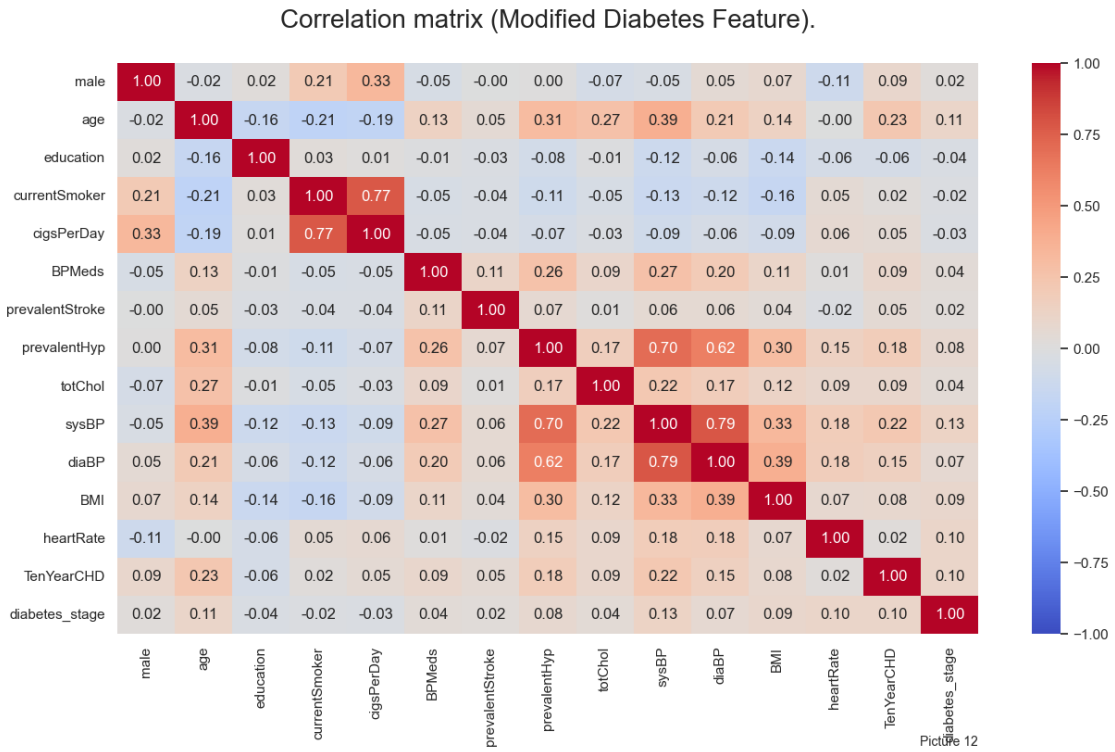


Picture 11

We see that there are multiple collinear variables: - diabetes and glucose - currentSmoker and cigsPerDay - prevalentHyp and sysBP, diaBP - sysBP and diaBP.

First we will engineer new feature, combining the diabetes data and glucose concentration by staging the diabetes into 3 stages (0-2). Let's try to divide diabetes by stages per CDC classification: * Normal (0): < 100 mg/dl * Pre-diabetes (1): 100 to 125 mg/dl * Diabetes(2): > 125 mg/dl

The resulting correlation matrix is shown on the Picture 12.

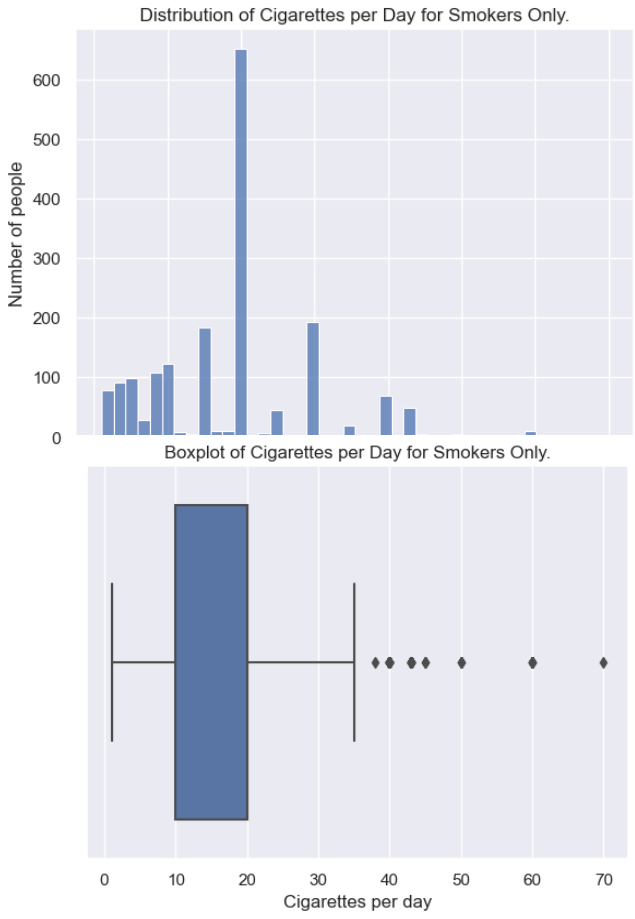


Picture 12

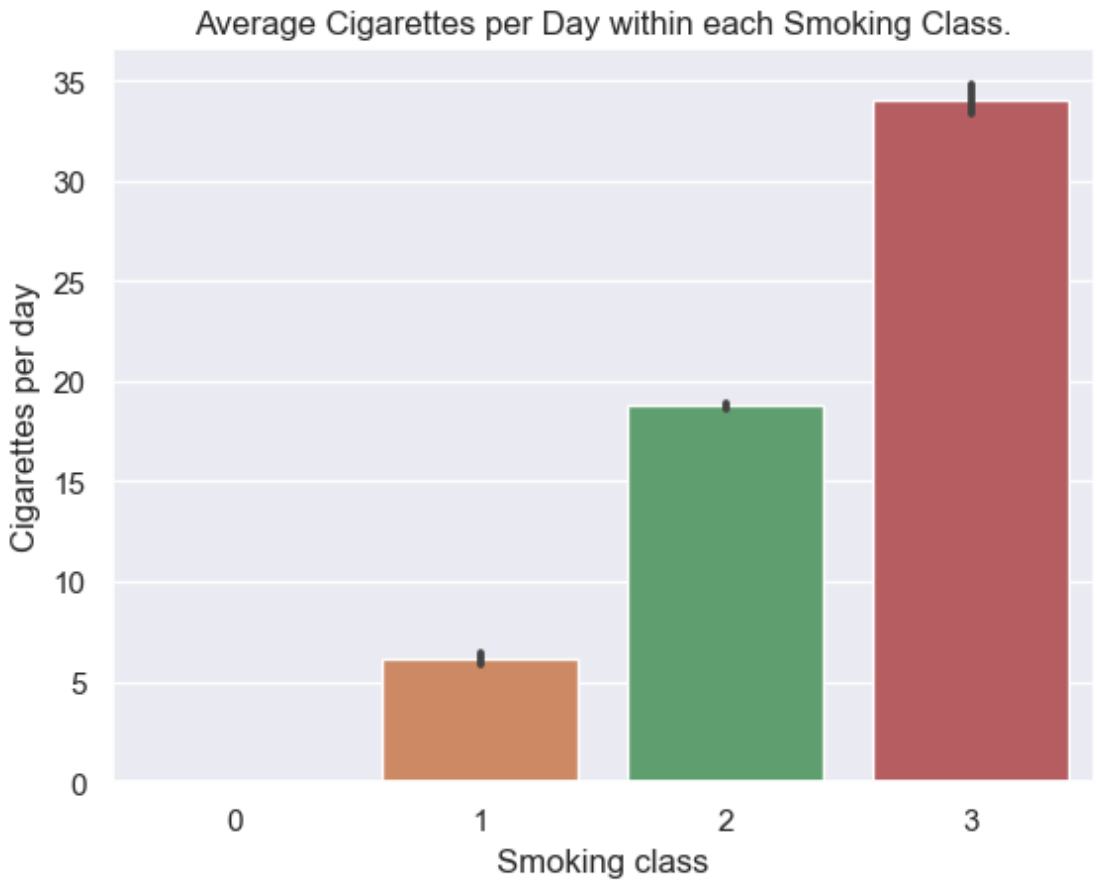
Next, we need to address the smoking and number of cigarettes features.

Pictures 13 and 14 show smoking data for all patients who do smoke, i.e. dropping all zeros. As seen on the histogram and the box plot, there are many outliers, which we need to adjust for. The good way to do it is to classify cigarettes per day into the smoking classes as follows: - 0 cigs/day - 1-10 cigs/day - 11-20 cigs/day - more than 20 cigs/day This somewhat reflects our data. Also, this is going to be easier to interpret, as usual cigarette pack has 20 cigarettes. Hence, the classes would mean:

- Non-smoker
- Half a pack per day
- One pack per day
- More than one pack per day



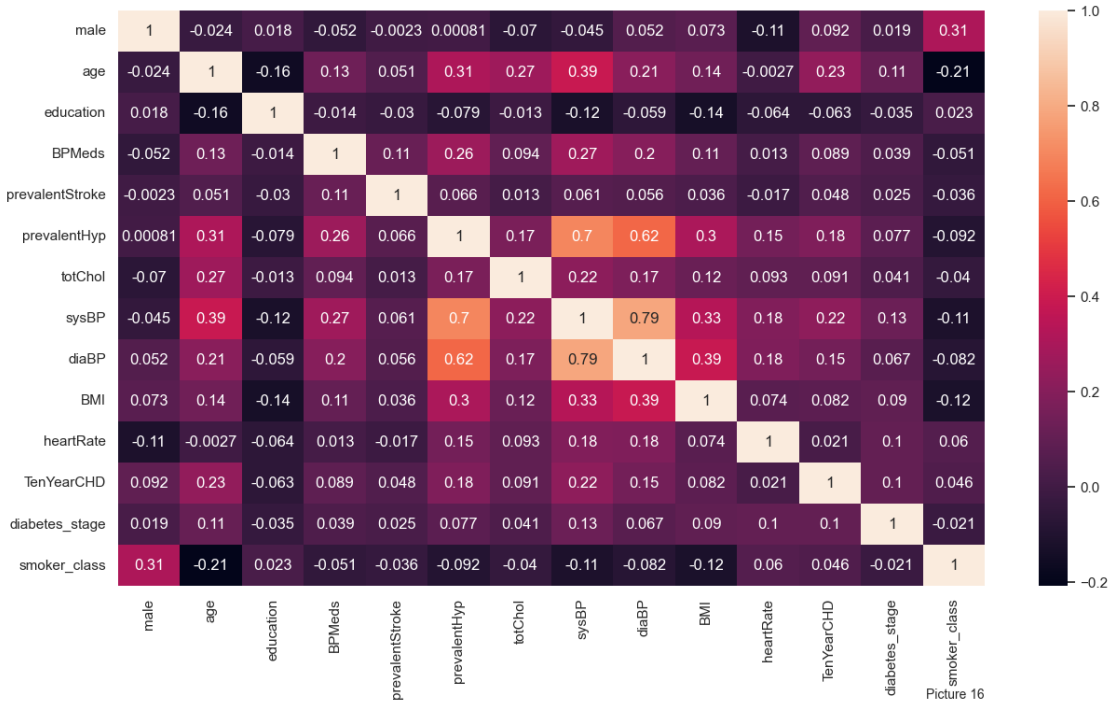
Picture 14



Picture 15

The numbers of cigarettes per day within each resulting class are shown on the Picture 15.

Correlation matrix (Modified Smoking Features).



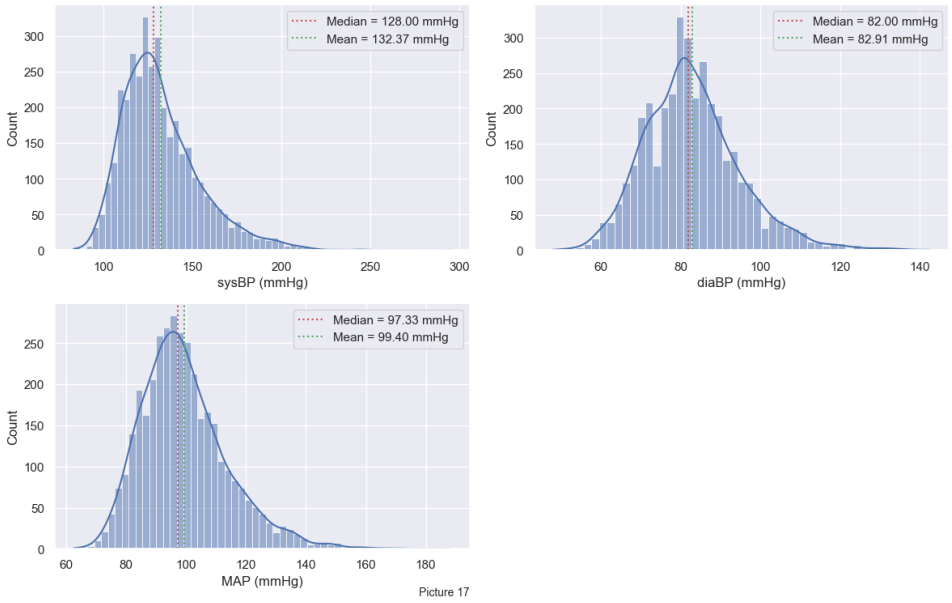
Picture 16

Picture 16 shows the correlation matrix with newly added smoker class.

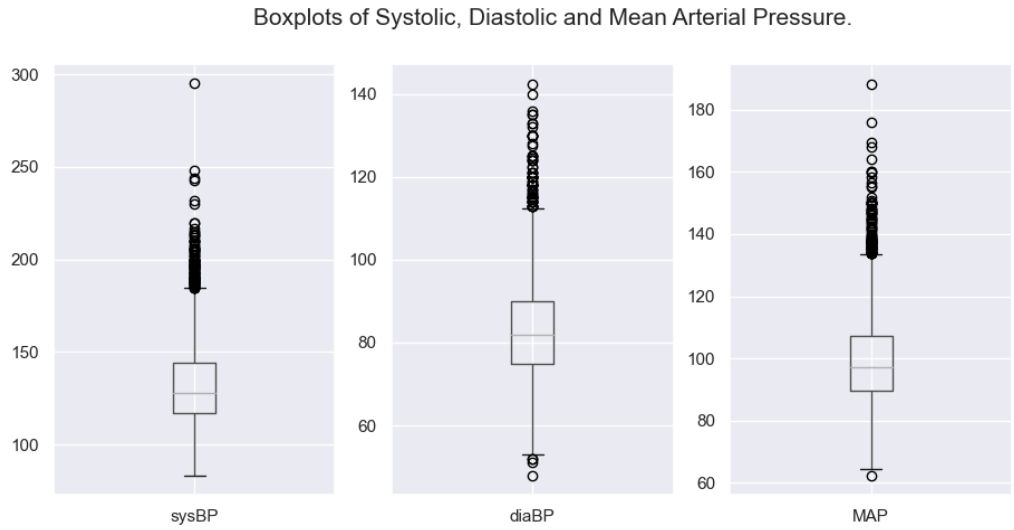
Next, we are going to address the hypertension and blood pressures. Systolic and Diastolic pressures can be very accurately represented with the Mean Arterial Pressure (MAP) value. The formula used to calculate MAP is:

$$MAP = DP + \frac{SP - DP}{3}$$

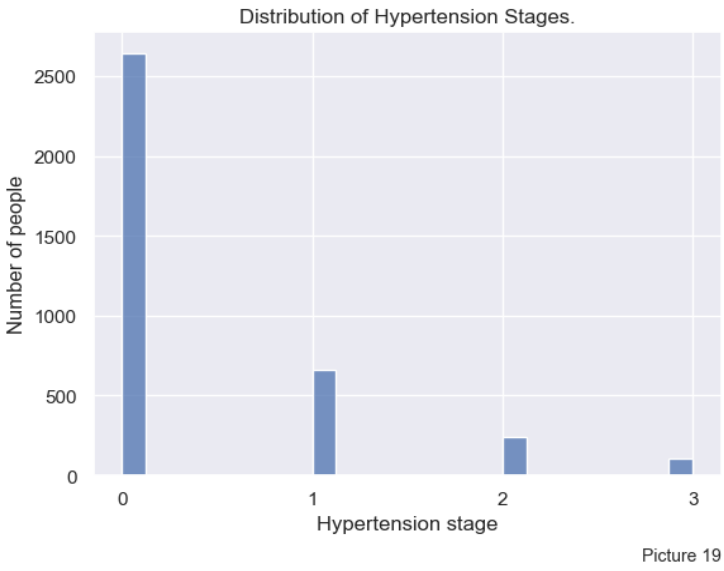
Systolic, Diastolic and Mean Arterial Pressure Distributions.



Picture 17



Picture 18



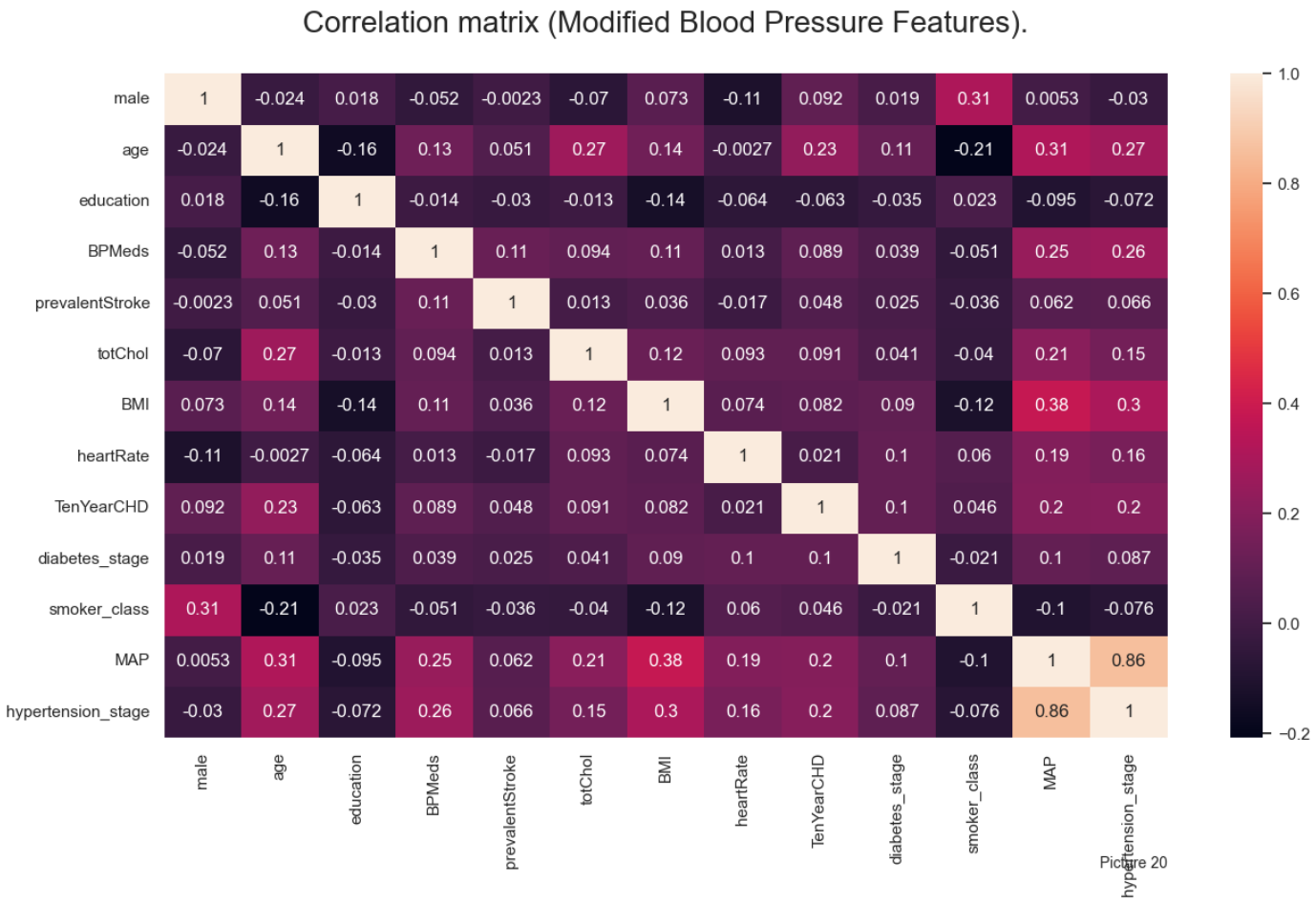
Picture 19

Pictures 17 and 18 show the distributions of SP, DP and new MAP values. MAP decently represents other two columns. Still, something must be done with the outliers.

Let's classify all patients by their hypertension grade using their Mean Arterial Pressure:

1. Normal MAP: < 99 2.
2. High normal MAP: 99.01 – 105.67 3.
3. Grade 1 hypertension: 105.68 – 119.00 4.
4. Grade 2 hypertension:119.01 – 132.33 5.
5. Grade 3 hypertension: ≥132.34 We will unite the Normal MAP and High normal MAP classes, so that Normal MAP is <105.67 mmHg. Number of patients in new classes is shown on Picture 19.

Picture 20 shows the new correlation matrix with dropped SP and DP columns.



Picture 20

Model selection

For the classification problem there several models that could be used. They are Logistic Regression, Decision Trees, Random Forest, Support Vector Machines, K-Nearest Neighbors, and Naive Bayes and different variations of these algorithms.

The choice of model depends on the specific characteristics of the data and the problem at hand. Logistic Regression is a simple and interpretable model that works well for binary classification problems. Decision Trees and Random Forest are often used for their ability to handle non-linear relationships and interactions between features. Support Vector Machines are powerful models that work well for high-dimensional data and can handle complex decision boundaries. K-Nearest Neighbors is a simple algorithm that works well for small datasets with few features. Naive Bayes is a probabilistic classifier that works well for text classification tasks.

To select the best model for our problem, we can compare their performance using metrics such as accuracy, precision, recall, F1 score, and area under the ROC curve. We can also use techniques such as cross-validation, randomized search, Bayesian optimization to tune hyperparameters and optimize model performance.

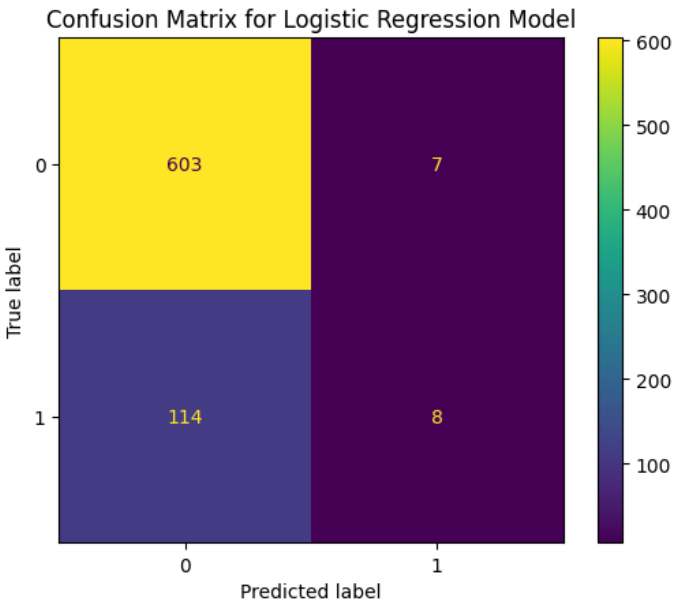
Also, we've tried an automated ML approach by using AutoGluon, the results of which will be shown below.

Ultimately, the best model will be the one that provides the highest RECALL (as this is a screening test) and generalizes well to new data, maximizes the True Positives, while minimizing False Negatives.

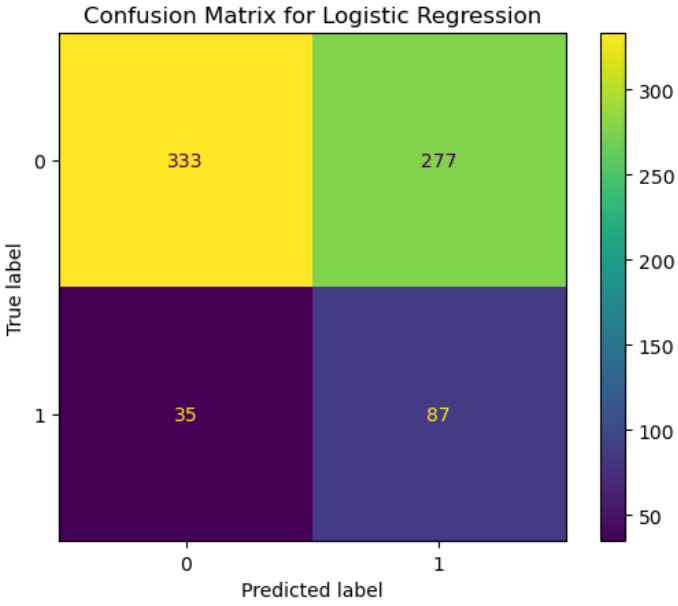
$$Recall = \frac{TruePositive(TP)}{TruePositive(TP) + FalseNegative(FN)}$$

Model Training and Evaluation

As a reference model we chose to train a simple Logistic Regression model on a dataset with dropped missing values. The result was not really what we expected to see from the screening test. Picture 21 shows that the model barely predicts the positive value of a target feature, resulting very low Recall (=0.07).



This happens because the target feature is highly imbalanced (see Picture 1). That's why we need to balance target feature class using.



We can use SMOTE technique, or the built-in functionality provided by Sci-Kit Learn package. For the SMOTE-balanced data the confusion matrix for the Logistic Regression Model (with tuned hyperparameters C=0.004832930238571752, penalty='l1', solver='liblinear') looks much better and the predictions gained from this model can be used for the screening test (see Picture 22).

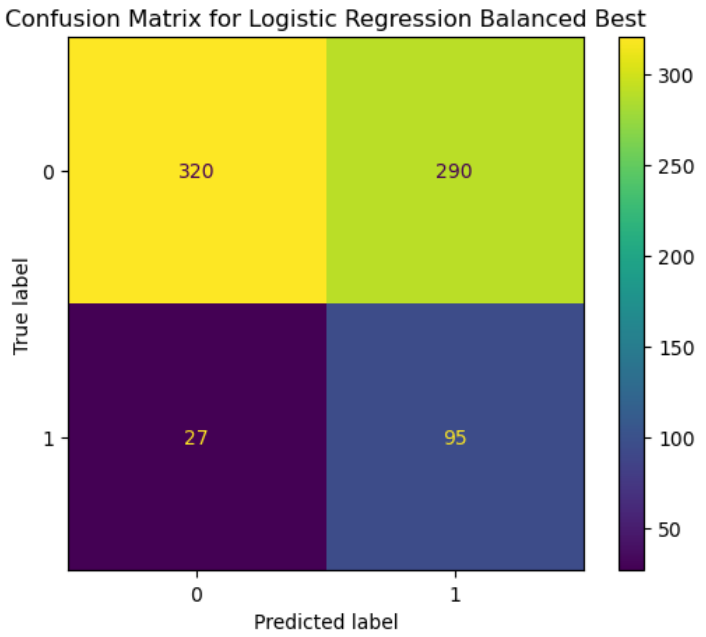
Out of 122 Positive values the model predicted 87. However, this is not the best we can do, as 35 patients are still being False Negatives. Other models' performances are shown in the table below.

Picture 22

Note that the Logistic Regression trained on a SMOTE-balanced data has so far, the greatest recall value (0.714)

	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1 Score	Test F1 Score	Train ROC AUC	Test ROC AUC
Logistic Regression SMOTE Drop	0.705705	0.663934	0.702212	0.272059	0.714343	0.606557	0.708225	0.375635	0.705705	0.640984
Logistic Regression SMOTE	0.705705	0.663934	0.702212	0.272059	0.714343	0.606557	0.708225	0.375635	0.705705	0.640984
Gradient Boosting SMOTE Drop	0.837083	0.732240	0.821702	0.313131	0.860988	0.508197	0.840887	0.387500	0.837083	0.642623
Gradient Boosting SMOTE	0.837083	0.732240	0.821702	0.313131	0.860988	0.508197	0.840887	0.387500	0.837083	0.642623
K-Nearest Neighbors SMOTE	0.862796	0.617486	0.794905	0.213768	0.977903	0.483607	0.876959	0.296482	0.862796	0.563934
Support Vector Machine SMOTE	0.802129	0.661202	0.774854	0.235294	0.851748	0.459016	0.811483	0.311111	0.802129	0.580328
Naive Bayes	0.817031	0.792350	0.355491	0.352941	0.282759	0.295082	0.314981	0.321429	0.596582	0.593443
Random Forest SMOTE	1.000000	0.774590	1.000000	0.313043	1.000000	0.295082	1.000000	0.303797	1.000000	0.582787
K-Nearest Neighbors Best Params	1.000000	0.771858	1.000000	0.281553	1.000000	0.237705	1.000000	0.257778	1.000000	0.558197
XGBoost SMOTE	0.990358	0.801913	0.999182	0.333333	0.981519	0.188525	0.990272	0.240838	0.990358	0.556557
SVM Best Params	0.939466	0.796448	0.992366	0.309859	0.597701	0.180328	0.746055	0.227979	0.798449	0.550000
K-Nearest Neighbors	0.869357	0.830601	0.726496	0.464286	0.195402	0.106557	0.307971	0.173333	0.591273	0.540984
XGBoost	0.990082	0.812842	1.000000	0.317073	0.933333	0.106557	0.965517	0.159509	0.966667	0.530328
Gradient Boosting	0.883379	0.831967	0.989583	0.473684	0.218391	0.073770	0.357815	0.127660	0.608995	0.528689
Logistic Regression	0.855335	0.834699	0.607143	0.533333	0.078161	0.065574	0.138493	0.116788	0.534661	0.527049
Random Forest Best Params	1.000000	0.823770	1.000000	0.315789	1.000000	0.049180	1.000000	0.085106	1.000000	0.513934
Random Forest	1.000000	0.831967	1.000000	0.454545	1.000000	0.040984	1.000000	0.075188	1.000000	0.515574
Support Vector Machine	0.854651	0.833333	0.857143	0.500000	0.027586	0.008197	0.053452	0.016129	0.513391	0.503279

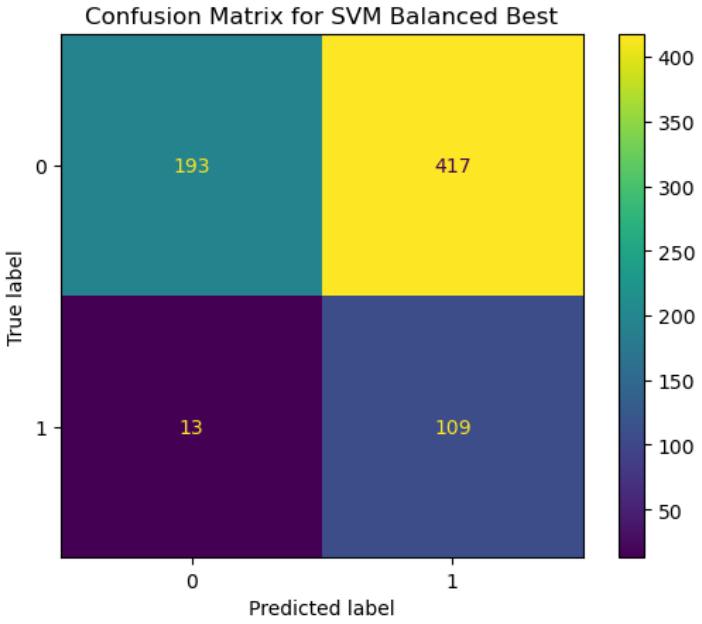
Next, we will try to use built-in class balancing functionality. For the new tuned Logistic Regression model the confusion matrix is shown on the Picture 23.



Picture 23

This model has a better performance than the one trained using the SMOTE technique. The recall value also improved (0.78). While training the other models we achieved even greater recall values, but there is a catch.

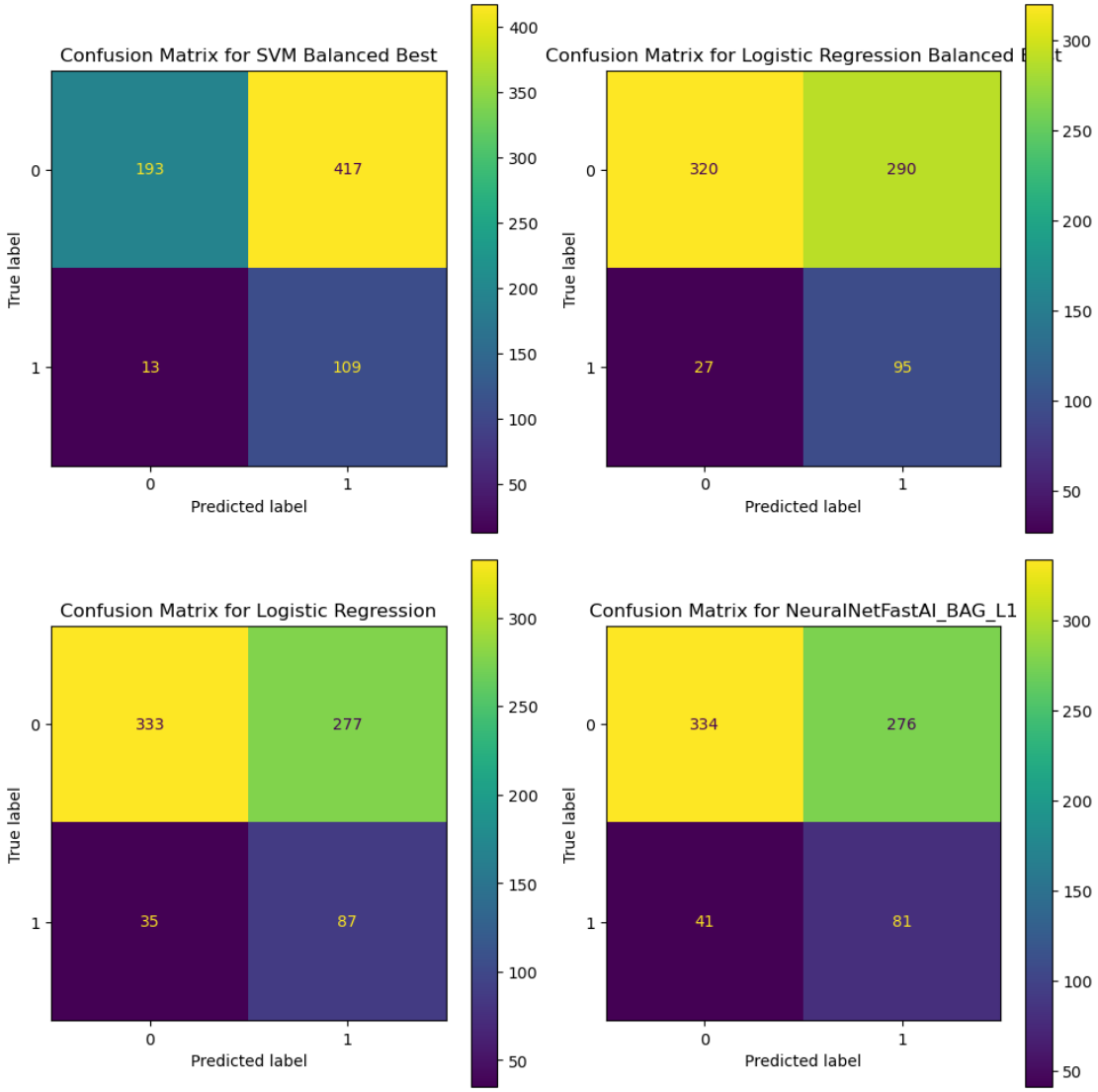
As seen on the Picture 24, the greater recall is achieved by getting more True Positives but also getting too many False Positives (shown on the example of SVM model).



Picture 24

The problem with this model lays directly in the field where it is intended to be used, healthcare screening test. While we want to minimize the False Negatives for any screening test, our goal is also to minimize False Positives in order to lower the rate of patients' false alarms (as good as possible, not lowering the Screening power). High False Positive rate is just going to lower the trust for the test by both patients and providers. We want to avoid that.

Let’s compare the performances of the four best models (evaluated by the Recall Score).



We see that the only case when we get more True Positives and less False Negatives than the tuned Logistic Regression model (Logistic Regression Balanced Best) is from the tuned SVM model, but as discussed above, we cannot accept it.

The more in-depth evaluation of these models compared them by their: Sensitivity (Recall), Positive Predictive Value (Precision), Specificity (True Negative Rate), Balanced Accuracy, F1 Score, Matthews correlation coefficient and Average Precision.

Model	TP	FP	FN	TN	Sensitivity (Recall)	PPV (Precision)	Specificity (TNR)	Balanced Accuracy	F1 Score	MCC	Average Precision
SVM Balanced Best	109	417	13	193	0.893443	0.207224	0.316393	0.604918	0.336420	0.173900	0.185143
Logistic Regression Balanced Best	95	290	27	320	0.778689	0.246753	0.524590	0.651639	0.374753	0.226356	0.192144
Logistic Regression	87	277	35	333	0.713115	0.239011	0.545902	0.629508	0.358025	0.193062	0.170442
NeuralNetFastAI_BAG_L1	81	276	41	334	0.663934	0.226891	0.547541	0.605738	0.338205	0.157672	0.150641
SVM Balanced BayesOpt	80	208	42	402	0.655738	0.277778	0.659016	0.657377	0.390244	0.240120	0.182149

By sorting the table by each of the metrics and giving each model a rank (row it is placed) we can build a rank table for these models, showing which place each model tool when it was ranked by the metric.

	Logistic Regression Balanced Best	SVM Balanced Best	SVM Balanced BayesOpt	Logistic Regression	NeuralNetFastAI_BAG_L1
Sensitivity (Recall)	2	1	5	3	4
PPV (Precision)	2	5	1	3	4
Specificity (TNR)	4	5	1	3	2
Balanced Accuracy	2	5	1	3	4
F1 Score	2	5	1	3	4
MCC	2	4	1	3	5
Average Precision	1	2	3	4	5

We can see that the best model has the balance between being good at Recall, while also keeping other metrics consistent. And this model is Logistic Regression Balanced Best, which ranked 2nd at almost every metric. It is 4th based on the Specificity Rank, but this metric is not crucial. Also, it was ranked 1st in Average Precision, which is great.

Conclusion

Based on our study, we can verify that the tuned Logistic Regression model with balanced class weights (**Logistic Regression Balanced Best**) performs the best across trained models.