# stone4مبادرة رواد مصر الرقمية

DEPI Graduation Project

HealthCare Predictive analysis

**Team Members:**

* Amina Zeyad Atta Ibrahim
* Amr Ashour Mohamed Ali
* Elham Mokhtar Ahmed Mohammed
* Enas Samir Abbas Mohamed
* Esraa Mohamed Sayed Mohamed

**Training Company:** CLS

**Group Number:** CLS\_\_CAI2\_AIS4\_G4

**Instructor:** Mahmoud El-Sayed

# **HealthCare Predictive analysis**

Contents

[**1. Project Overview** 5](#_heading=h.amleyvsthfjk)

[2. Milestone1 5](#_heading=h.joleahk7gptk)

[2.1 Data Collection: 5](#_heading=h.u0a8ml366yij)

[2.2 Data Exploration: 5](#_heading=h.r61pnbffc4ti)

[2.2.1 Check Data Statistics; First 5 records, Last 5 records, NULLs , Duplicates in Dataset and check if there is any categorical column needs encoding: 6](#_heading=h.5h6g7lv2frcw)

[2.2.2 Data Distribution and Handling: 8](#_heading=h.vsgnek635l53)

[2.2.2.1 Features Distribution and Box plots after data handling 16](#_heading=h.yhjylaqkzbh4)

[**3. Milestone2 17**](#_heading=h.bsfyjix4jj94)

[3.1 Exploratory Data Analysis (EDA) 17](#_heading=h.2w63awjvehr4)

[3.1.1 Body Mass Indicator (BMI) 17](#_heading=h.1mrgvhi9zbc3)

[3.1.2 Hypertension Feature 18](#_heading=h.114a3pg9rtx)

[3.1.3 Pulse Pressure Feature (PP) & Mean Arterial Pressure (MAP) 18](#_heading=h.z0hqshnef602)

[3.1.4 Age group and PP\_Category Features: 19](#_heading=h.88nx634ercxo)

[3.2 Data Visualization 20](#_heading=h.lsd44nr8155g)

[3.3 Check Dependency among features 21](#_heading=h.511j69y9d1kj)

[3.3.1 Check dependency between categorical columns 21](#_heading=h.xedkdgkgglt3)

[3.3.2 Check Dependency between Target column and numerical features 22](#_heading=h.co8krba4r4ye)

[3.3.3 Check Dependency among Features 22](#_heading=h.yfrz83rodu8m)

[3.3 Check Target column balance 23](#_heading=h.bsi12zrgvwga)

[3.4 Data Scaling 23](#_heading=h.y4vhvzzds78g)

[**4. Milestone3 25**](#_heading=h.6mqsawya05rb)

[4.1 Model Selection 25](#_heading=h.t0b5ldl8fynf)

[4.1.1 Trial 1 25](#_heading=h.qwl8qc3lkkf3)

[4.1.2 Trial 2 27](#_heading=h.mspo90ayxoqv)

[4.1.3 Trial 3 27](#_heading=h.l1hphrrwc418)

[4.1.4 Trial 4 27](#_heading=h.nij0rgequwra)

[4.1.5 Trial 5 28](#_heading=h.4pa2yrif0sub)

[📈 Model Performance: 28](#_heading=h.i9kxtg9lmftv)

[📋 Classification Report: 28](#_heading=h.vcsuibu31roe)

[4.1.6 Trial 6 29](#_heading=h.wtiv9vvorll3)

[📈 Model Performance: 29](#_heading=h.75zakazbhesn)

[📋 Classification Report: 29](#_heading=h.5j14qaovp3o3)

[4.1.7 Trial 7 30](#_heading=h.as5r1it0pvgz)

[📈 Model Performance: 30](#_heading=h.fui230fbxevx)

[📋 Classification Report: 30](#_heading=h.mq1m5jar0mwy)

[**5. MileStone 4 31**](#_heading=h.1m03rvctillv)

[5.1 mlflow for tracking models 31](#_heading=h.99w1p0ggk8a)

[5.2 Demplyment 31](#_heading=h.q4ahpguer68p)

# List of Figures

Figure 1 Original Dataset sample 5

Figure 2 First 5 records in the dataset 6

Figure 3 Last 5 records in the dataset 7

Figure 4 Checks on Data(columns type,NULLS) 7

Figure 5 Checks on Duplicates and Nulls 8

Figure 6 Data Description Part1 8

Figure 7 Data Description part2 9

Figure 8 Sample of Data after age conversion 9

Figure 9 Data Description after age conversion. Changes are in age 9

Figure 10 Features Distribution part1 10

Figure 11 Features Distribution part2 10

Figure 12 Features Distribution part3 10

Figure 13 Number of records in which gender=3 11

Figure 14 Features Box plots part1 11

Figure 15 Features Box Plots part2 12

Figure 16 Possible Lowest Systolic (ap\_hi) 13

Figure 17 Possible Highest Systolic(ap\_hi) 13

Figure 18 Thresholds for ap\_lo 14

Figure 19 Mean of ap\_hi of subset (ap\_hi < 40mmHg & ap\_hi > 150) 14

Figure 20 Distribution of ap\_lo filtered on (<40 & >150 & cardio=1) 15

Figure 21 Q-Q Plot of ap\_lo after removing outliers 15

Figure 22 Sample of Dataset after cleaning 16

Figure 23 Data Description after cleaning 16

Figure 24 Features Distribution Part1 17

Figure 25 Features Distribution Part2 17

Figure 26 BMI formula 18

Figure 27 BMI Categories 18

Figure 28 Sample of dataset after adding bmi & bmi\_category columns 19

Figure 29 Sample of Dataset after adding hypertension column 19

Figure 30 Pulse Pressure PP formula 19

Figure 31 Mean Arterial Pressure (MAP) formula 19

Figure 32 Dataset sample after adding MAP and PP features 20

Figure 33 Dataset sample after adding PP\_category and age\_group 21

Figure 34 BMI Category & Age group vs cardio 21

Figure 35 Hypertension\_category & age\_group vs Cardio 22

Figure 36 Chi2\_Contengency between target column and features. 22

Figure 37 Pi-serial correlation 23

Figure 38 Pearson Correlation among features 23

Figure 39 Cardio column 24

Figure 40 Data Before scaling 24

Figure 41 PowerTransform of features 25

Figure 42 Sample of data after scaling 25

Figure 43 SnapShot of PowerBI 26

# **Project Overview**

The Healthcare Predictive Analytics specially **Cardiovascular Disease (CVD)** project focuses on developing a classification model to classify the patient if he/she suffers from cardiovascular disease or not by providing data-driven insights. The model will be designed to help healthcare professionals with tasks such as patient risk detection, making informed decisions based on predictive analytics. The project will utilize a classification model.

# **Milestone1**

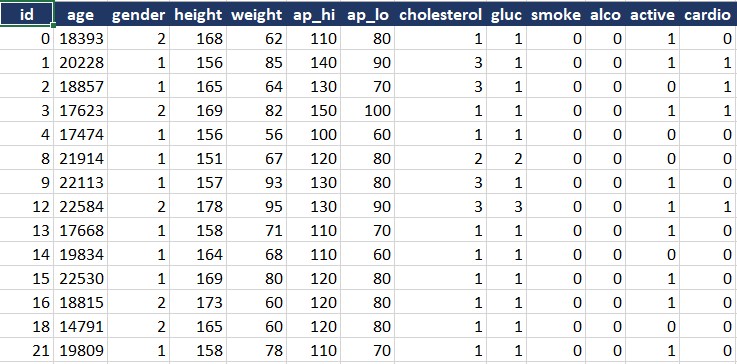
## **Data Collection:**

Our Dataset was collected from Kaggle Link:<https://www.kaggle.com/datasets/scientificstephen/medical-examination-dataset-analysis>

Link of Dataset on our drive: <https://drive.google.com/file/d/1uD5d16AkU_fdwTs3Fq6xqlGCN09A7o_7/view?usp=drive_link>

## **Data Exploration:**

Healthcare cardiovascular disease dataset consists of 70000 records of people and 13 factors taken into consideration in the dataset. The below figure shows a sample of the dataset



*Figure 1 Original Dataset sample*

Description of each column (Key Features):

* id: this is just a number used to identify the patient.
* age: contains the age of each patient in days.
* gender: The column identify the sex of each patient (1: Female,2: Male).
* height: contains the height of each patient in meters (m).
* weight: contains the weight of each patient in kilograms (kg).
* ap\_hi: Systolic of the patient in mmHg.
* ap\_lo: Diastolic of the patient in mmHg.
* cholesterol: categorize the cholesterol level of each patient (1:Low, 2:Medium, 3:High).
* gluc: categorize the glucose level of each patient (1:Low, 2:Medium, 3:High).
* Smoke: Categorize the patient if he/she is a smoker or not.(1: Smoker, 0: Not a smoker)
* alco: Categorize the patient if he/she drinks alcohol or not. (1: drinker, 0: not a drinker)
* active: Categorize if the patient practices any sport or not (1: practice any activity, 0: not practice any activity).
* cardio: this is the target column in which classifies the patient if he/ she suffers from cardiovascular disease or not.

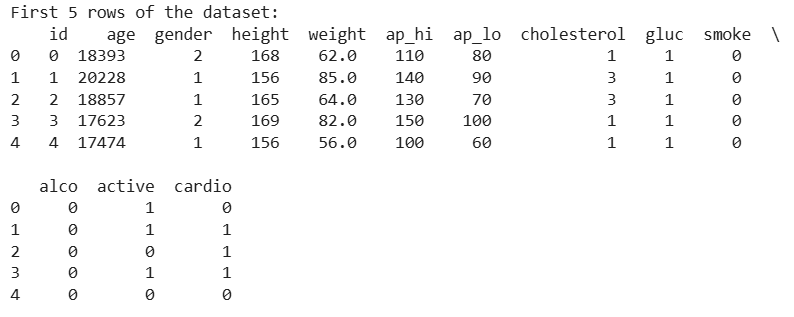
**Data Summary:**

1- Size: (13\*70000) Thousands of individual records.

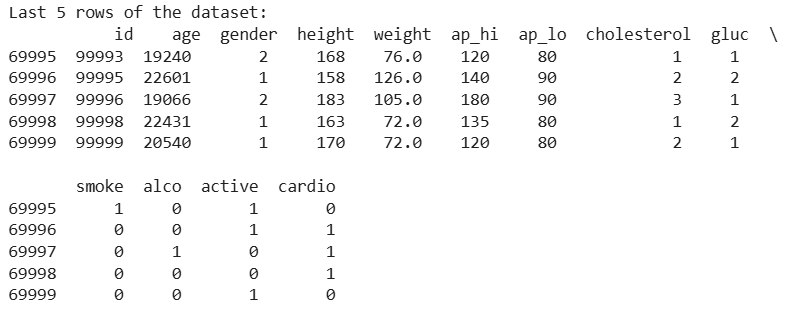
2- Type: Mixed numeric and categorical data.

3- Challenges: Includes outliers and categorical data requiring cleaning and preprocessing.

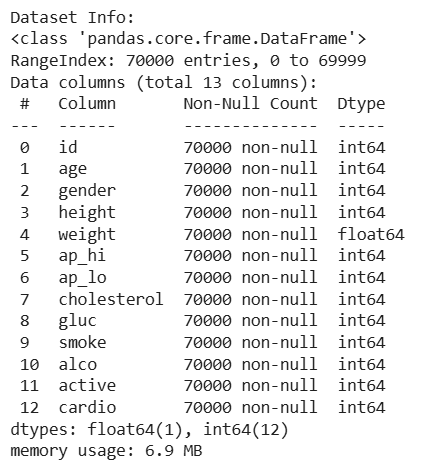
### **Check Data Statistics; First 5 records, Last 5 records, NULLs , Duplicates in Dataset and check if there is any categorical column needs encoding:**



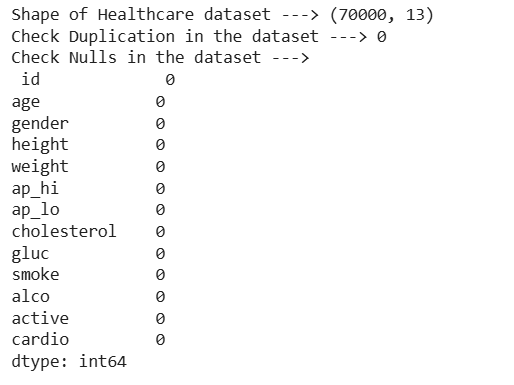
*Figure 2 First 5 records in the dataset*

**

*Figure 3 Last 5 records in the dataset*

**

*Figure 4 Checks on Data(columns type,NULLS)*

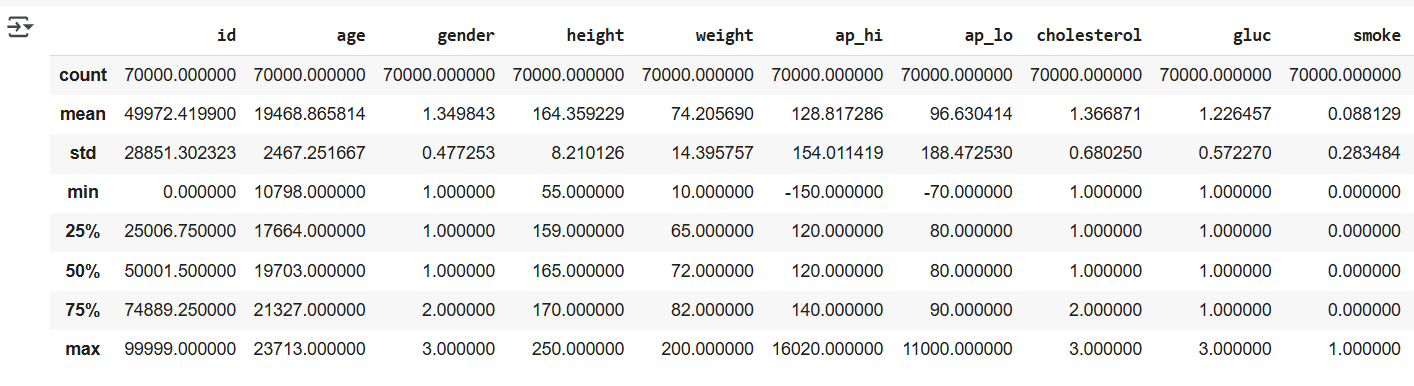
**

*Figure 5 Checks on Duplicates and Nulls*

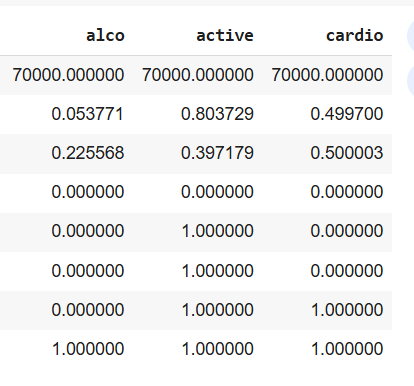
From the above figure it is found that:

* No Duplicates.
* No NULLs.
* No need for encoding.

## **Data Distribution and Handling:**



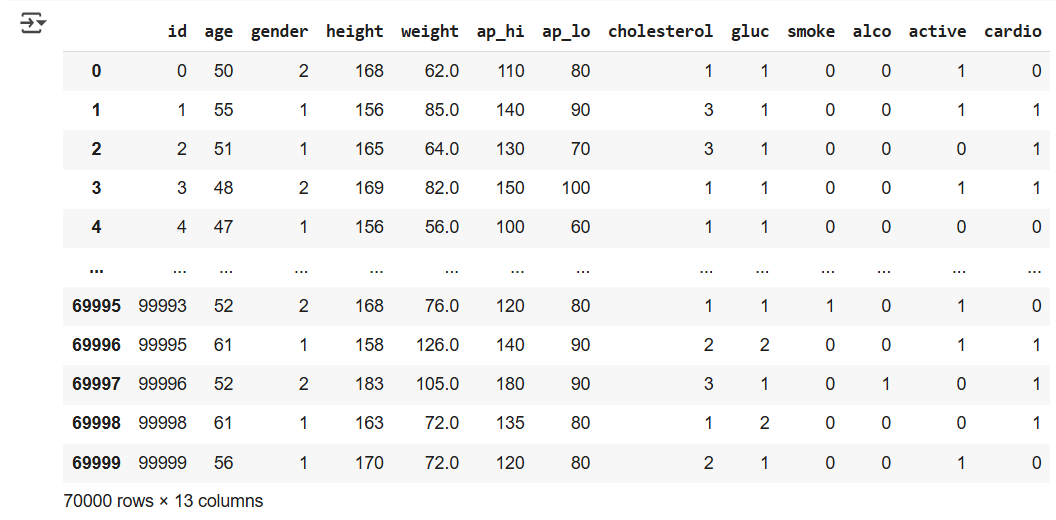
*Figure 6 Data Description Part1*



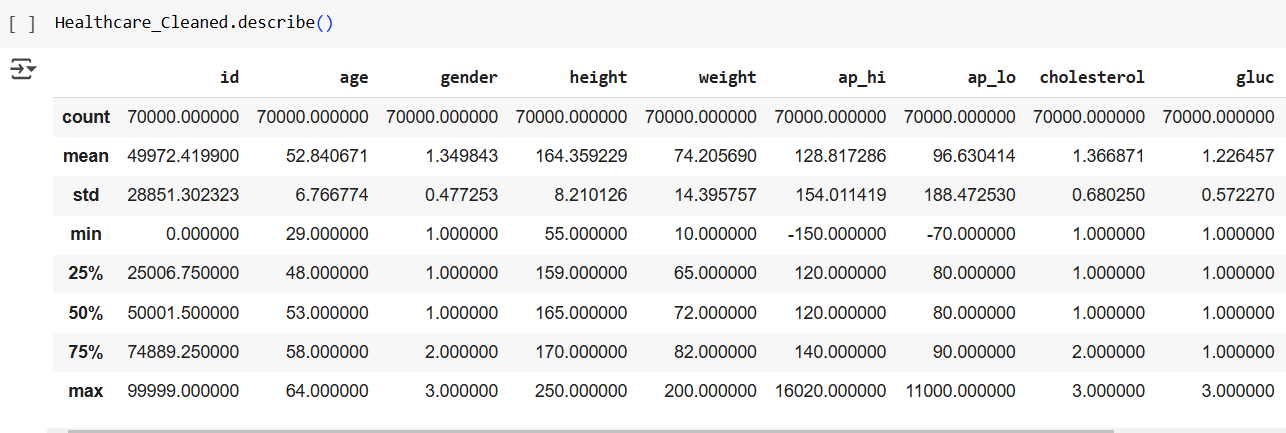
*Figure 7 Data Description part2*

From the above figures, it is found that the age has very large numbers as it is calculated in days.

So we have converted days into years then we have shown a sample of the data after conversion as follow



*Figure 8 Sample of Data after age conversion*



*Figure 9 Data Description after age conversion. Changes are in age*

Then we have drown distribution of each key feature as follow:

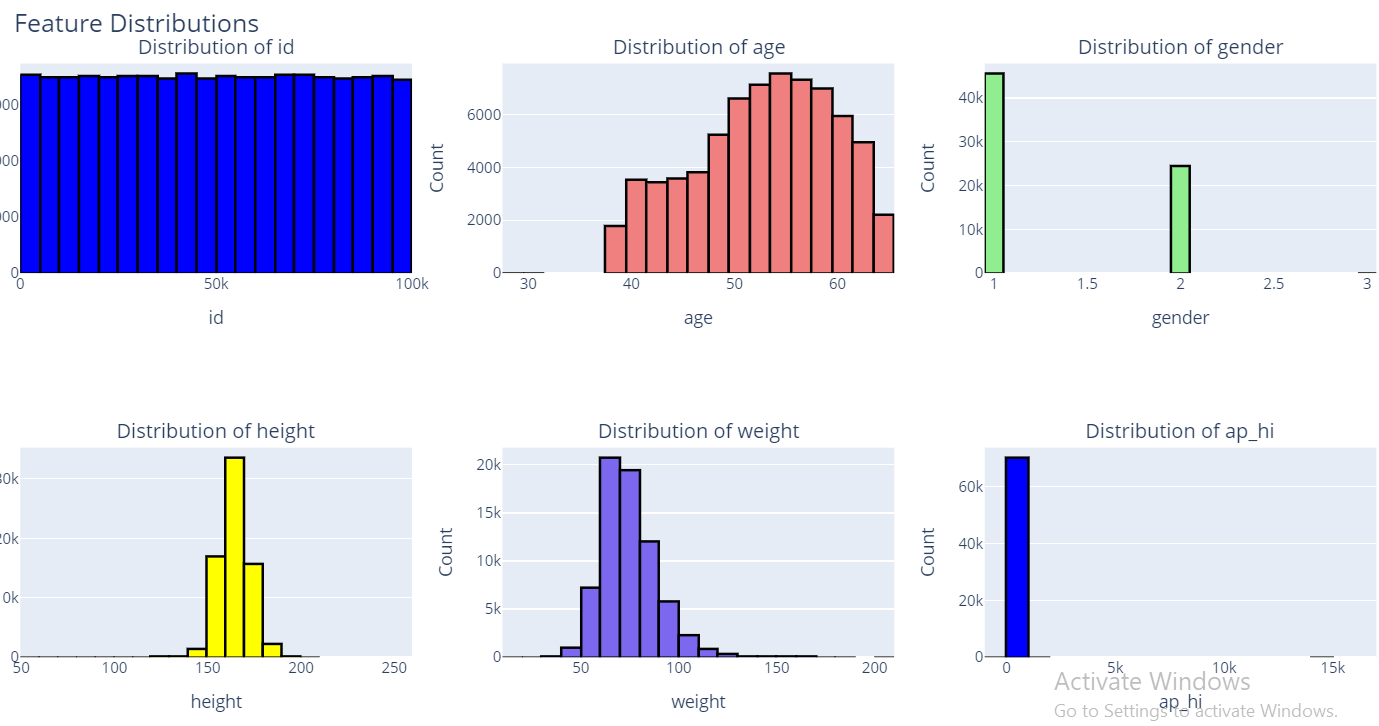


Figure 10 Features Distribution part1

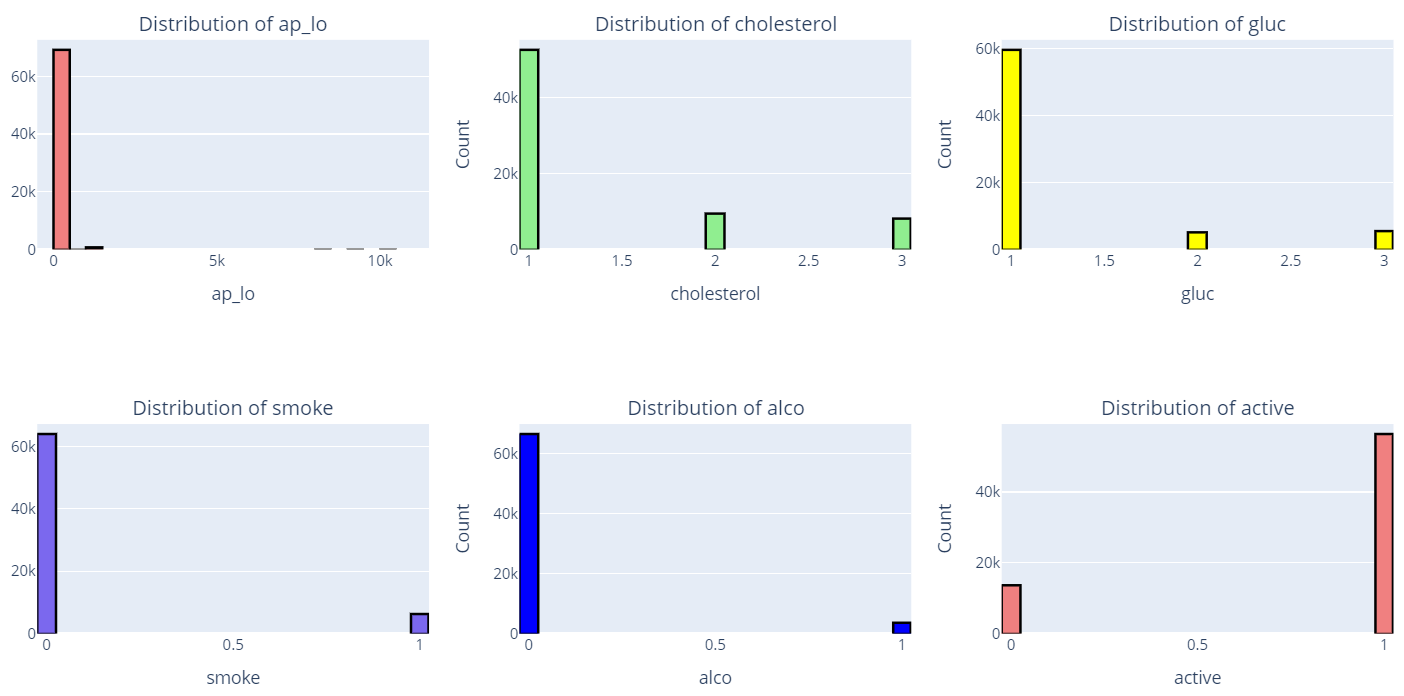


Figure 11 Features Distribution Part2

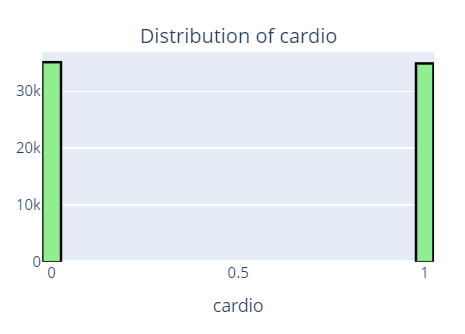


Figure 12 Features Distribution Part3

From the above figure we found that:

* Age distribution is left skewed data so it needs normalization to convert the left skewed to Gaussian distribution.
* Gender contains 3 categories which is not logic as 0 for Males, 1 for Females and 2 for what?! so it depends on the number of samples of this category. the number of samples is 11 records as shown below, it is recommended to eliminate it.

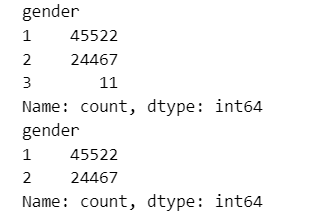
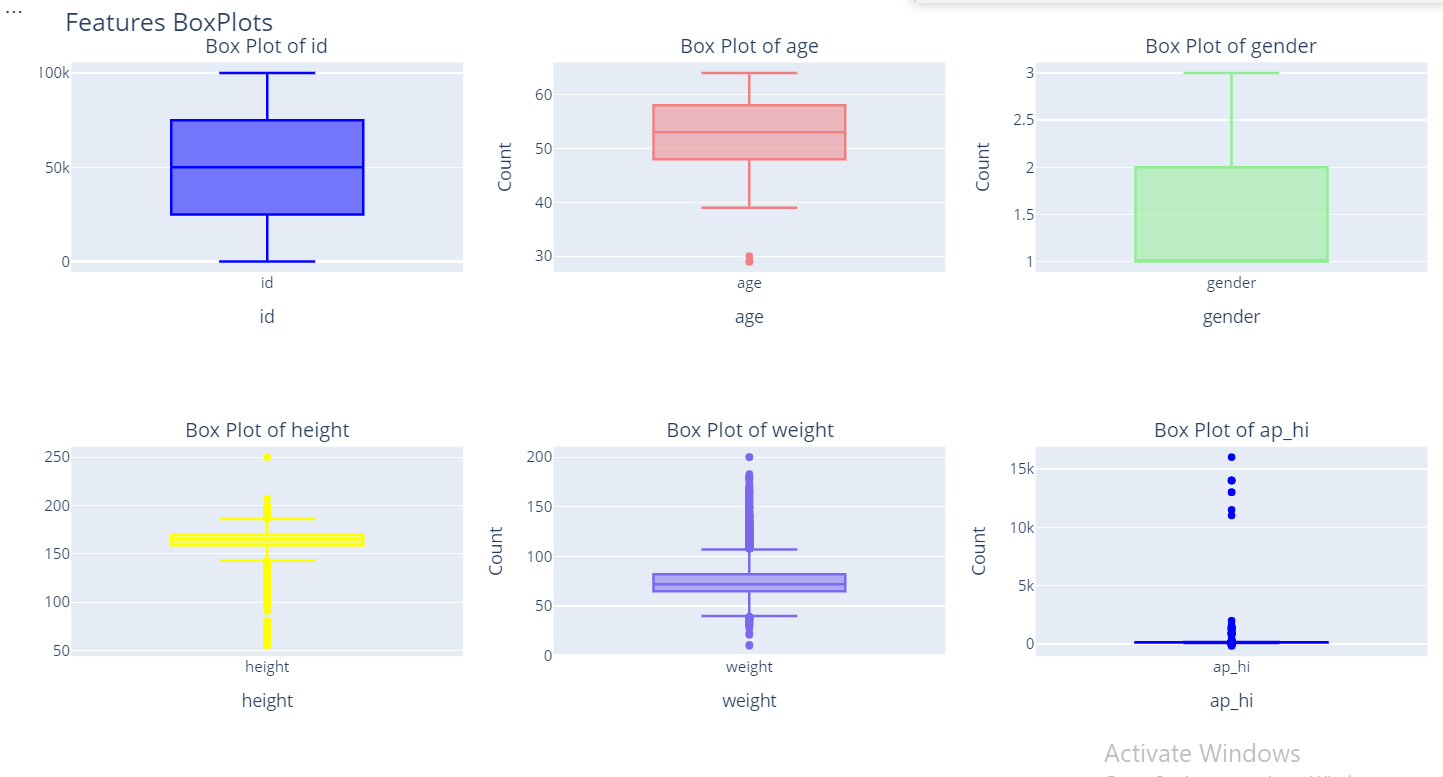
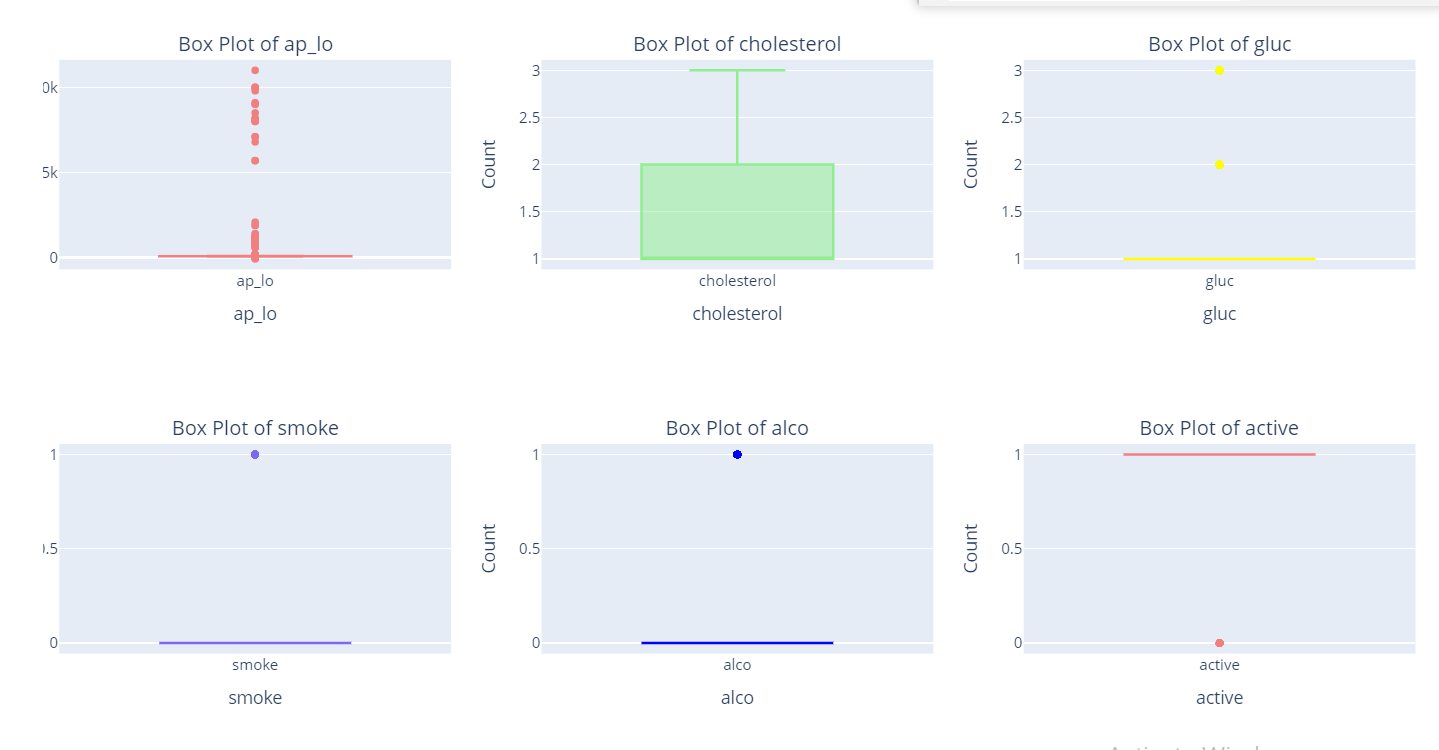


Figure 13 Number of records in which gender=3

* Height and weight are almost Gaussian distribution.
* ap\_hi and ap\_lo seem to have outliers as the maximum values from data description are 16020 and 11000 respectively which are not logic values and number of samples at these values are not large so we can handle them by elimination as follow.
* In features the most dominant samples are the normal samples of people but cardiovascular column is balanced which means that there are outliers in the dataset.



*Figure 14 Features Box plots part1*

**

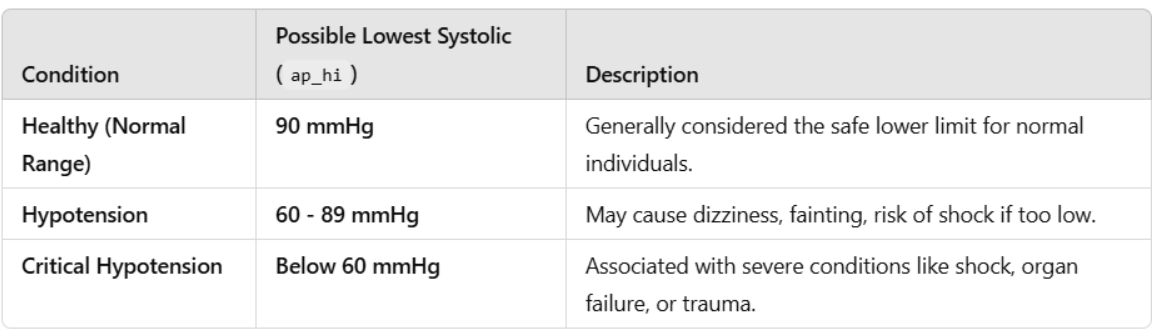
*Figure 15 Features Box Plots part2*

From the above figures we found that:

* The minimum and maximum values of age are 29 and 64 respectively which are normal values so we decided to keep it as it is.
* In height column minimum and maximum values are 55 and 250 cm respectively. But some of these values are not logic and not correlated with age. Thresholds we have chosen are 100 cm for the lower threshold and 200 cm for the upper threshold.

Based on these thresholds outlier samples count was 31 sample so we decided to eliminate them.

* In weight column minimum and maximum values are 10 and 200 kg respectively. Value 10 does not match with the minimum value of age 29 so our thresholds for weight were 45 kg as a lower threshold and 190 kg as an upper threshold. We found that 304 samples were out of this range (outlier) so we decided to eliminate them.
* For ap\_hi and ap\_lo thresholds chosen are dependent on medical domain as follow:



*Figure 16 Possible Lowest Systolic (ap\_hi)*

From the above figure we found that the possible lowest systolic value is 60 mmHg below that the human will not be alive. But if there is a noise in the device used to measure the pressure this may affect the measurement to we have chosen the lower threshold to be 50 mmHg.

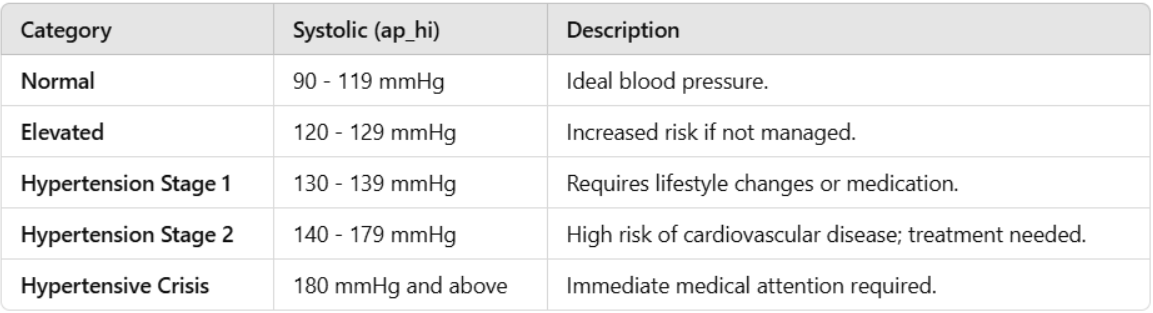


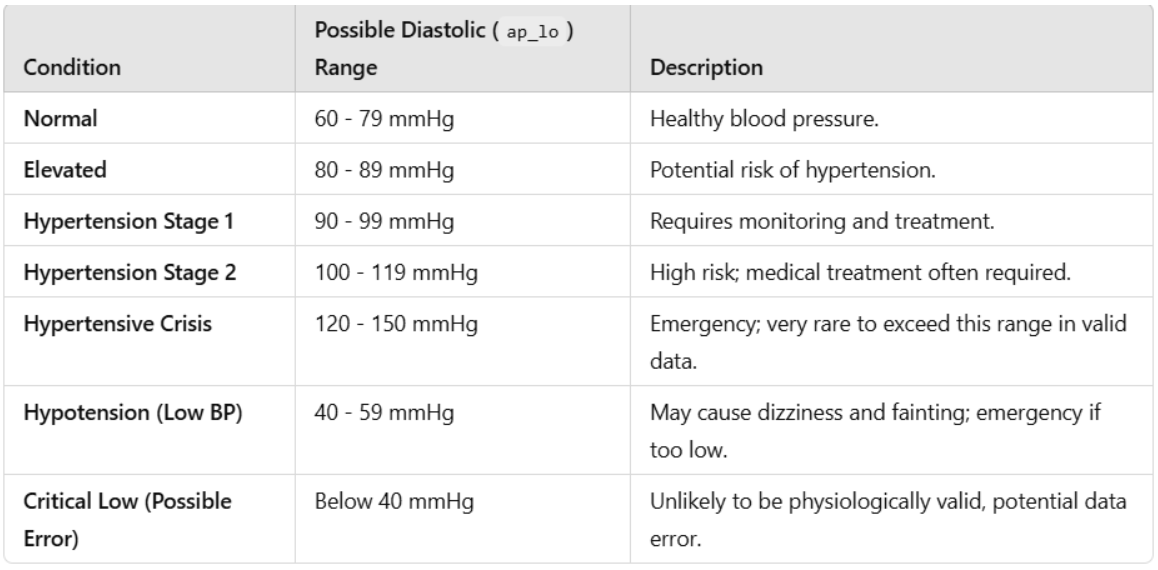
Figure 17 Possible Highest Systolic(ap\_hi)

From the above figure we found that the maximum value for ap\_hi is 180 mmHg but during search we found that the maximum value ap\_hi can be taken and the human is a live is around 250 mmHg.

So thresholds chosen for ap\_hi 🡪 from 50 mmHg to 250 mmHg.

Based on the above thresholds it is found that 224 samples are considered as outliers so we decided to eliminate these samples.

* For ap\_lo from search we found that:



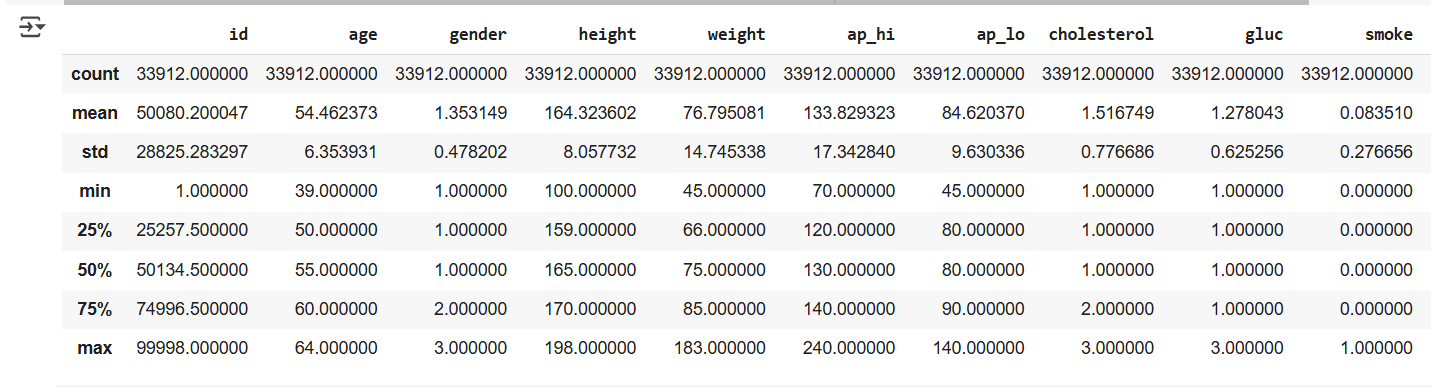
*Figure 18 Thresholds for ap\_lo*

From The above figure we found that the minimum value for ap\_lo is 40 mmHg and the upper threshold 150 mmHg.

Based on the previous thresholds we found that 1013 samples as outlier we decided not to eliminate these samples as they represent 1.4%.

The logic used to impute the outliers is as follow:

* By logic, people whose ap\_lo is lower than 40 mmHg and higher than 150 mmHg definitely suffer from cardiovascular disease which means cardio flag=1. This will give us subset of data. Number of samples in this subset is 837 records. we decide to calculate the mean of ap\_lo of these people and impute the outlier values with this mean.



*Figure 19 Mean of ap\_hi of subset (ap\_hi < 40mmHg & ap\_hi > 150)*

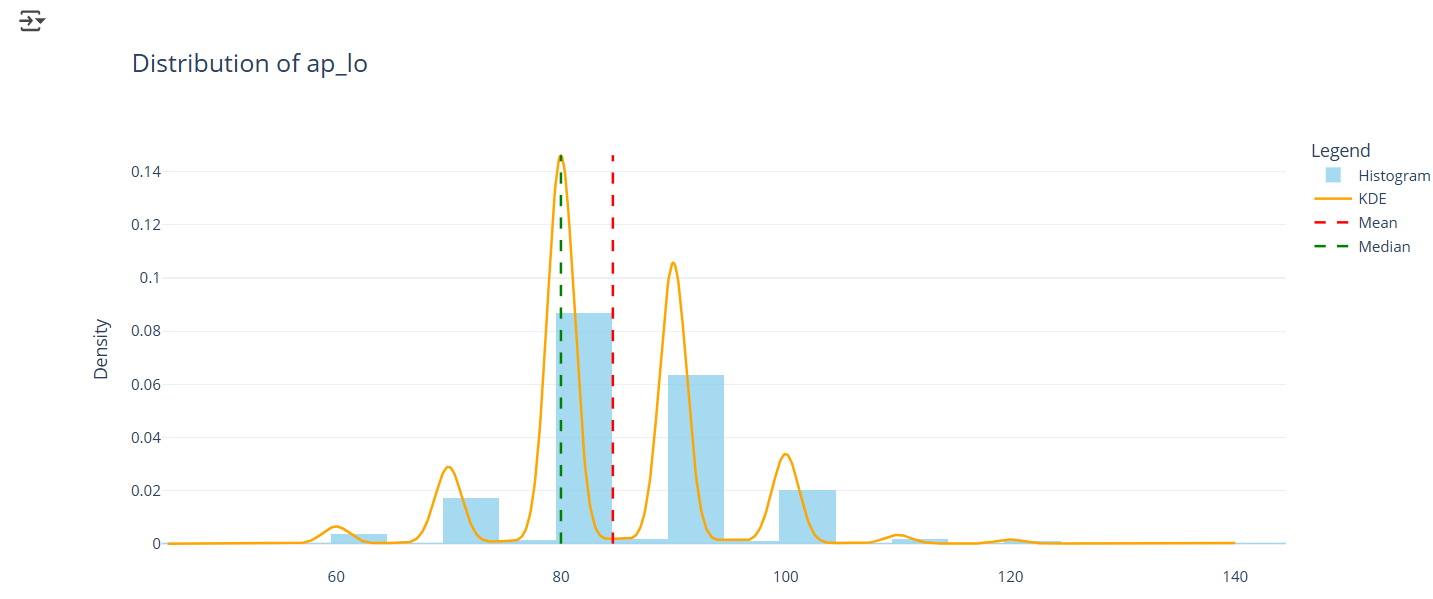
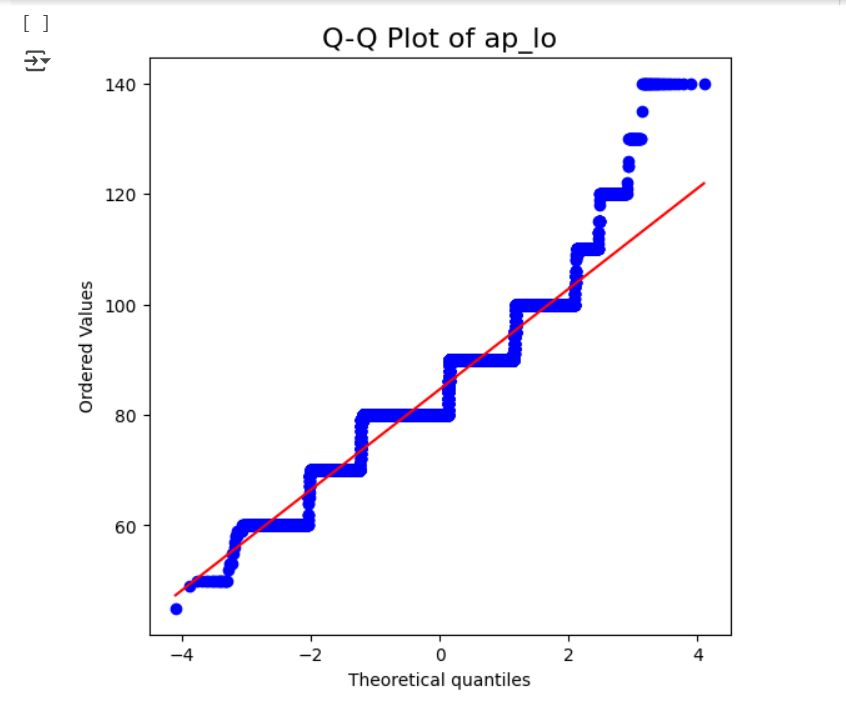


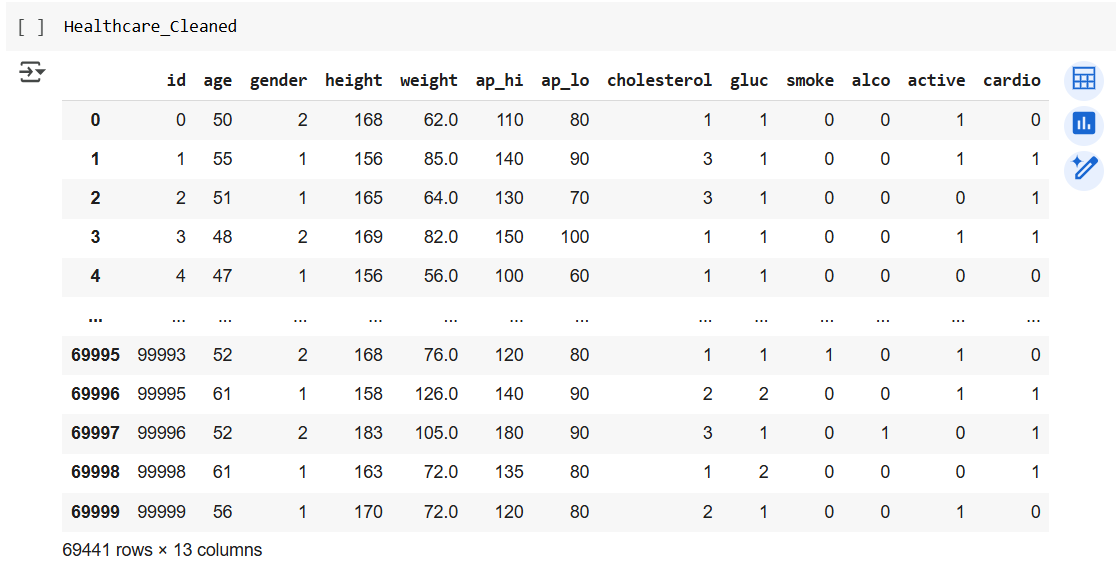
Figure 20 Distribution of ap\_lo filtered on (<40 & >150 & cardio=1)

From the above figure we found that mean=84 and median=80 so we substituted by mean.

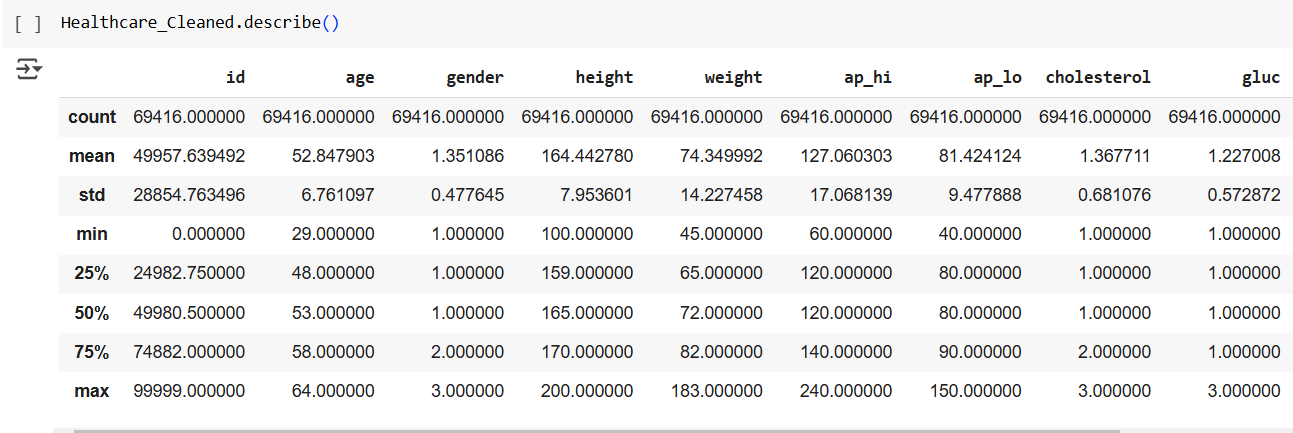


*Figure 21 Q-Q Plot of ap\_lo after removing outliers*

This is a Q-Q plot of ap\_lo after substitution which means that the data distribution almost became normal distribution.



*Figure 22 Sample of Dataset after cleaning*

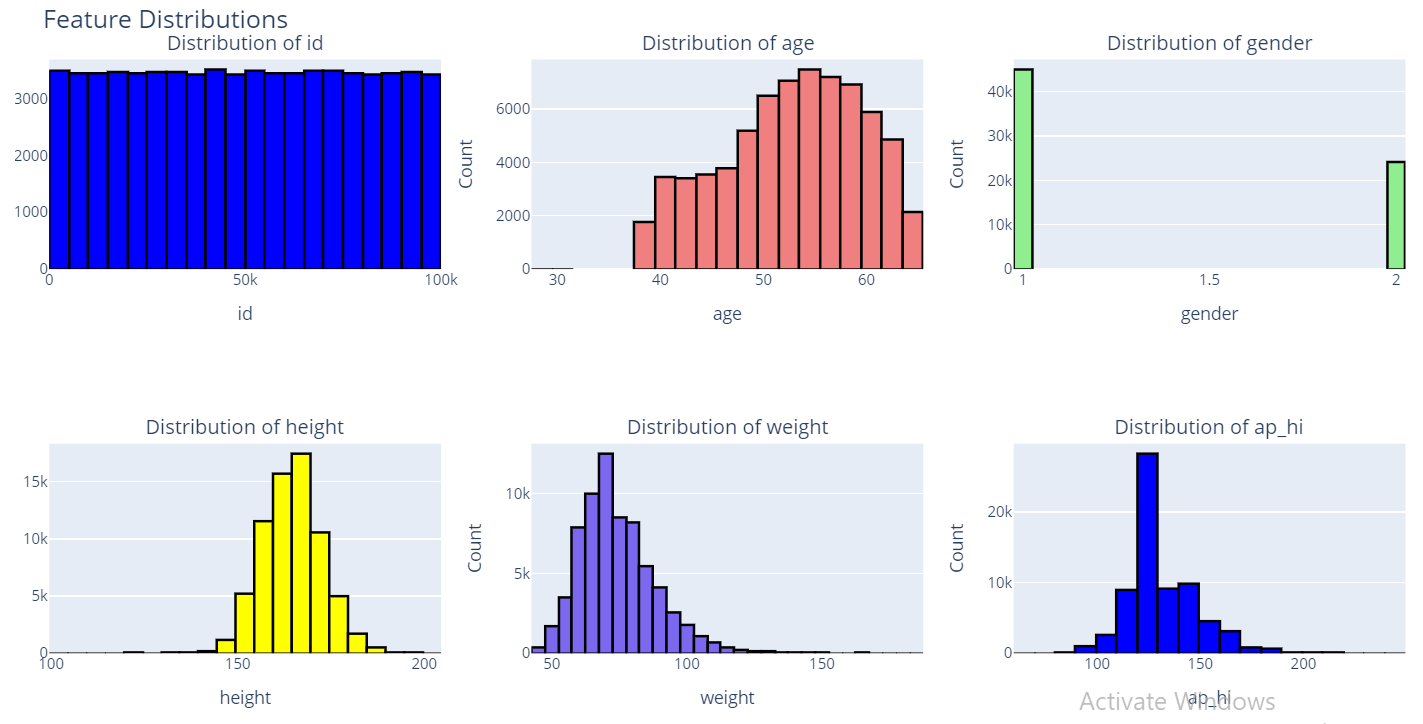


*Figure 23 Data Description after cleaning*

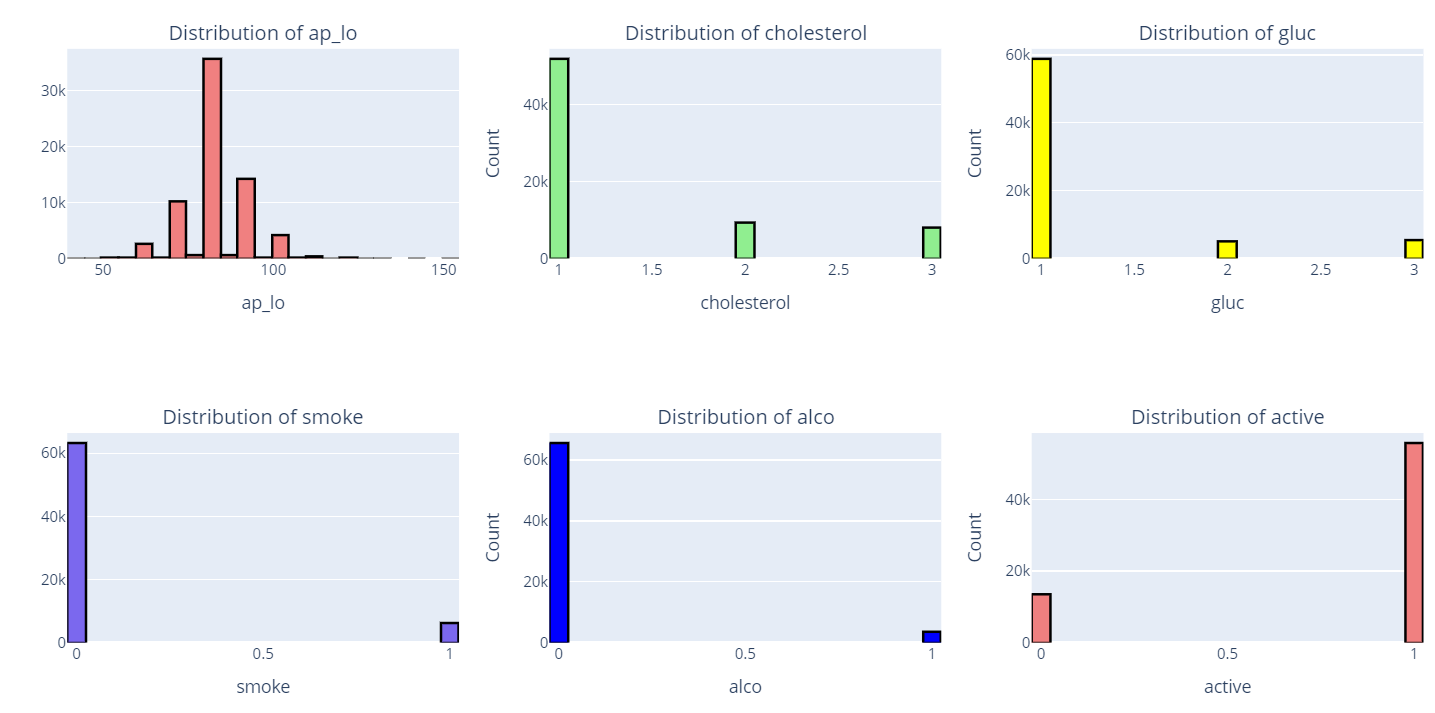
*There is a bug found during data investigation:*

*It is found that there are some records in which ap\_lo > ap\_hi which does not make sense.*

### **2.2.2.1 Features Distribution and Box plots after data handling**

**

*Figure 24 Features Distribution Part1*

**

*Figure 25 Features Distribution Part2*

*From the above figures it is found that:*

* *Gender feature now contains only two categories.*
* *Distribution of ap\_hi & ap\_lo are enhanced*

# **3. Milestone2**

## **3.1 Exploratory Data Analysis (EDA)**

In this part it is required to extract some features from the dataset which called Feature Engineering.

### **3.1.1 Body Mass Indicator (BMI)**

It is the measure which relates between the human’s wright and his height to know if he suffers from obesity or not.

Its formula as shown bemow we have used the first formula.

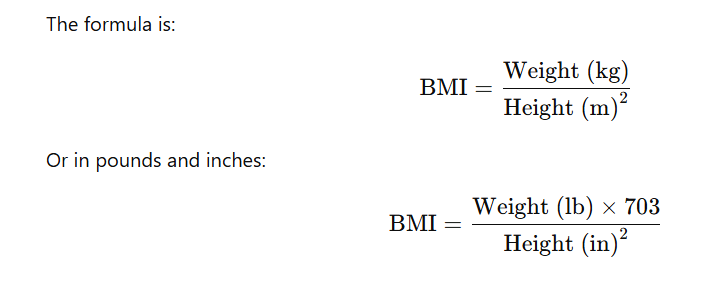


Figure 26 BMI formula

Based on search the BMI category is:



Figure 27 BMI Categories

We have divided our data into 4 categories

1→ Underweight

2→ Normal weight

3→ Overweight

4→ Obesity

So we did not encode this column as the categories are mapped to numbers.

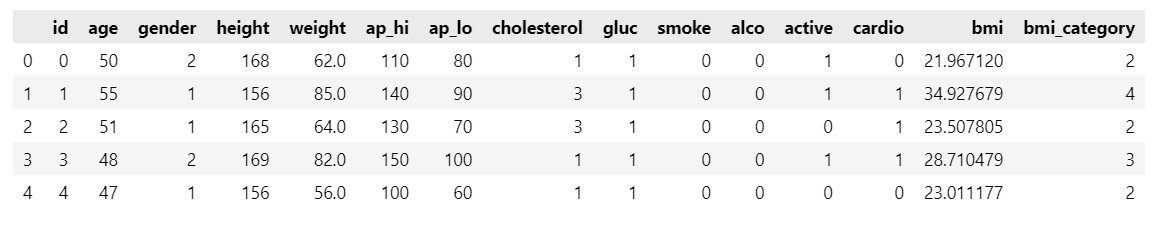


Figure 28 Sample of dataset after adding bmi & bmi\_category columns

### **3.1.2 Hypertension Feature**

We have thought that we have ap\_hi and ap\_lo from that we can know that if the client suffers from hypertension or not based on given thresholds attached above we found that if ap\_hi> 130 mmHg, it is stage1 of hypertension and if ap\_lo >90, it is stage 1 of hypertension. so if(ap\_hi>130 or ap\_lo>90) hypertension flag will be mapped to 1 as shown below.



Figure 29 Sample of Dataset after adding hypertension column

### **3.1.3 Pulse Pressure Feature (PP) & Mean Arterial Pressure (MAP)**

These features give us relation between ap\_hi and ap\_lo and from these features we can detect if the user suffers from cardiovascular or not. the formula of PP and MAP as follow:

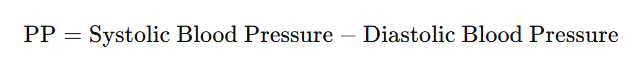


Figure 30 Pulse Pressure PP formula

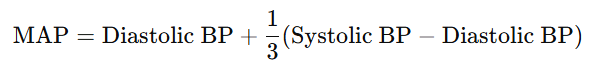


Figure 31 Mean Arterial Pressure (MAP) formula

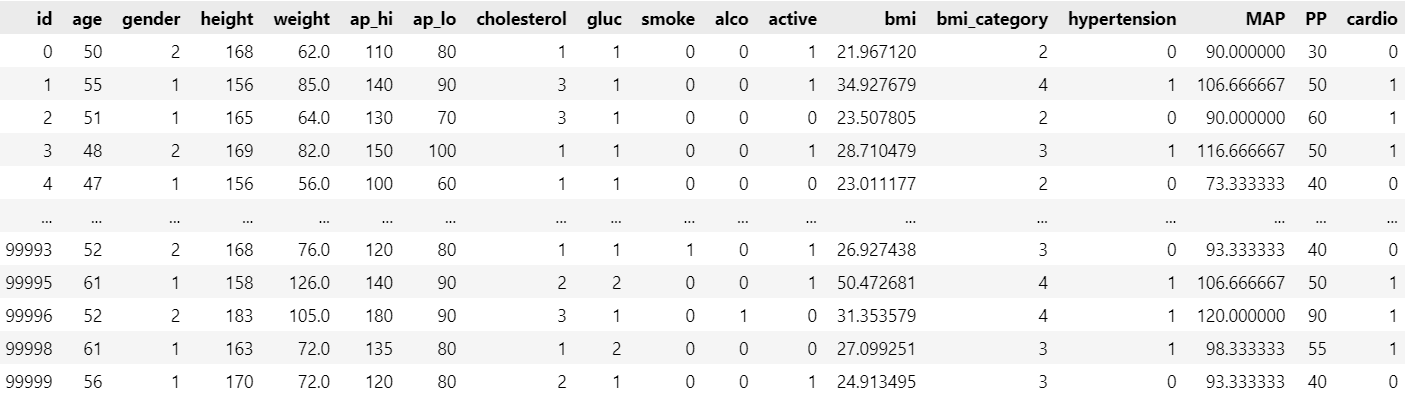


Figure 32 Dataset sample after adding MAP and PP features

### **3.1.4 Age group and PP\_Category Features:**

In this part we wanted to categorize the clients into age groups to know which group suffers from cardiovascular as by logic when age group increases the probability that the user suffers from cardio disease is higher. and want to categorize the PP into features.

**PP\_Category:**

from 0-->30 category 1 --> very low pressure (0-30 mmHg)

from 30-->60 category 2--> Normal/Low-Normal Pulse Pressure (30-60 mmHg)

from 60-->90 category 3--> Moderately Elevated Pulse Pressure (60-90 mmHg)

from 90-->120 category 4--> High Pulse Pressure (90-120 mmHg)

from 120-->140 category 5--> Very High Pulse Pressure (120-140 mmHg)

**Age\_group:**

from 20-29 → category 1

from 30-39 → category 2

from 40-49 → category 3

from 50-59 → categroy 4

from 60-69 → category 5

from 70-79 → category 6

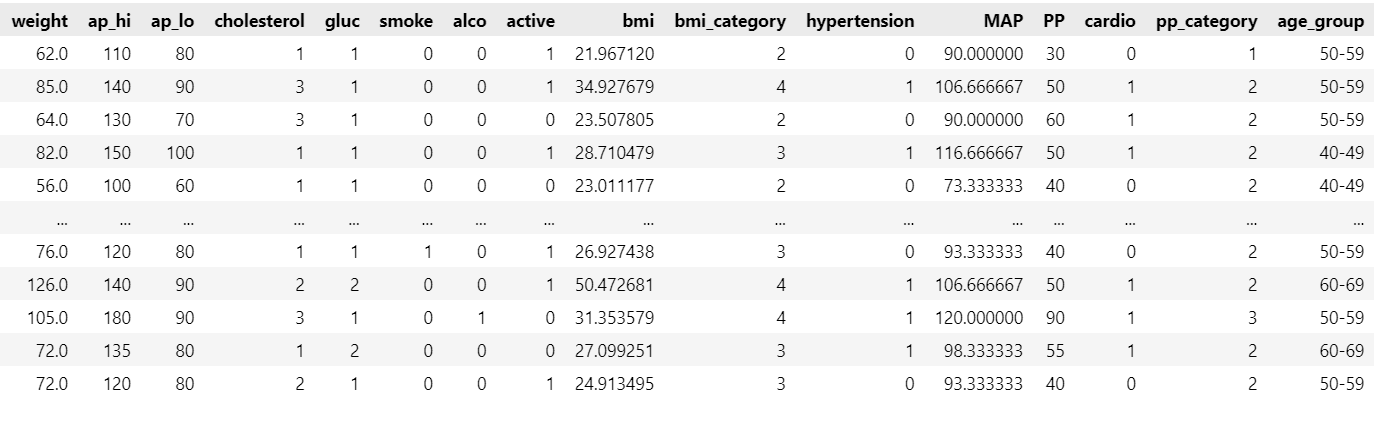


Figure 33 Dataset sample after adding PP\_category and age\_group

## **3.2 Data Visualization**



Figure 34 BMI Category & Age group vs cardio

From the above figure it is found that:

* Cardiovascular is directly proportional with both BMI\_Category & Age\_group categories for example let us take the extreme case

when BMI\_Category =4 and ae\_group is the maximum category (60-69) it has the maximum percentage of people who suffer from cardiovascular disease.

another example which validates the assumption

when BMI\_category=1 (lowest BMI\_Category) and age\_group from 20-29 it has the lowest percentage of clients who suffered from cardiovascular disease.

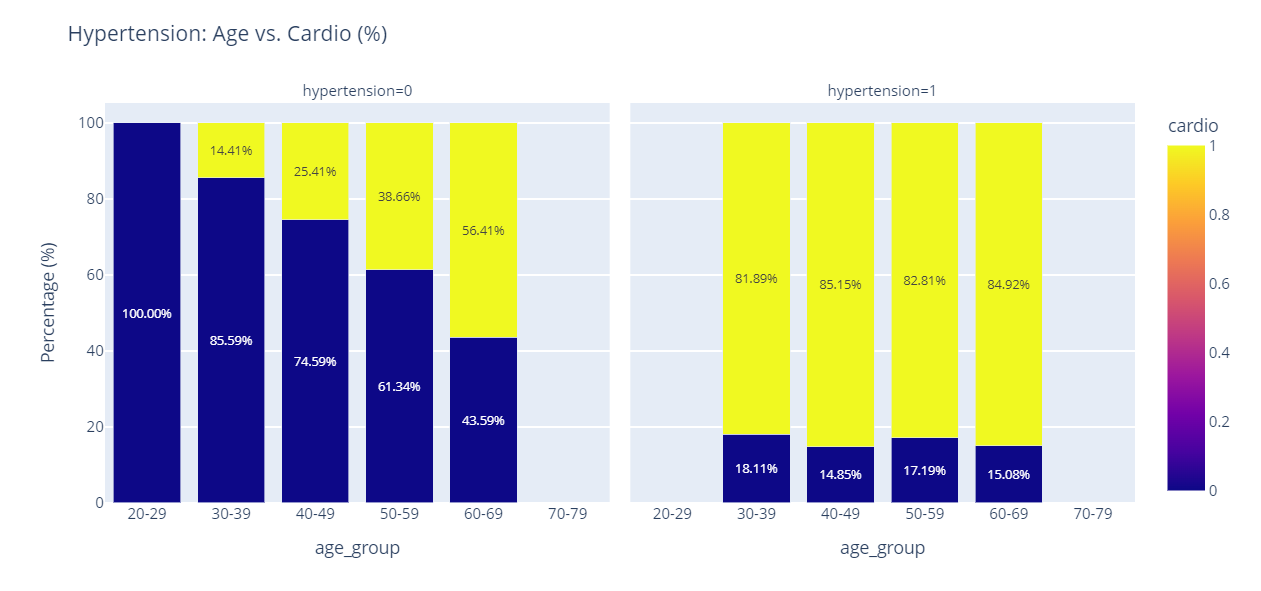


Figure 35 Hypertension\_category & age\_group vs Cardio

From the above photo it is found that

* Age group and hypertension both of them affect the client if he suffers from cardiovascular problem or not.

## **3.3 Check Dependency among features**

### **3.3.1 Check dependency between categorical columns**

As the target column is categorical (binary classification problem) and there are some categorical features ;['gender', 'cholesterol', 'gluc', 'smoke', 'alco','active', 'bmi\_category', 'hypertension','pp\_category','age\_group']

We have used Chi2\_Contengency . It assumes Null hypothesis in which assumes the features are independent of each other and calculate p if p<=0.05 this means that the columns are dependent.

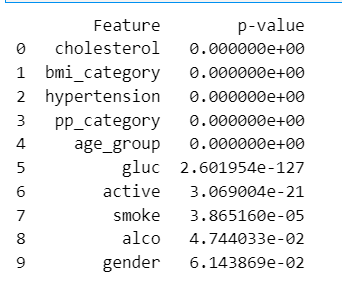


Figure 36 Chi2\_Contengency between target column and features.

From the above figure we found that the target column depends on all features except gender and low relationship with alco\_column.

### **3.3.2 Check Dependency between Target column and numerical features**

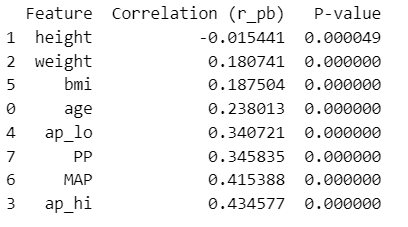


Figure 37 Pi-serial correlation

We find that the most effective features are ap\_hi, MAP ,PP & ap\_lo. Moderate effective features age, BMI, Weight.

### **3.3.3 Check Dependency among Features**

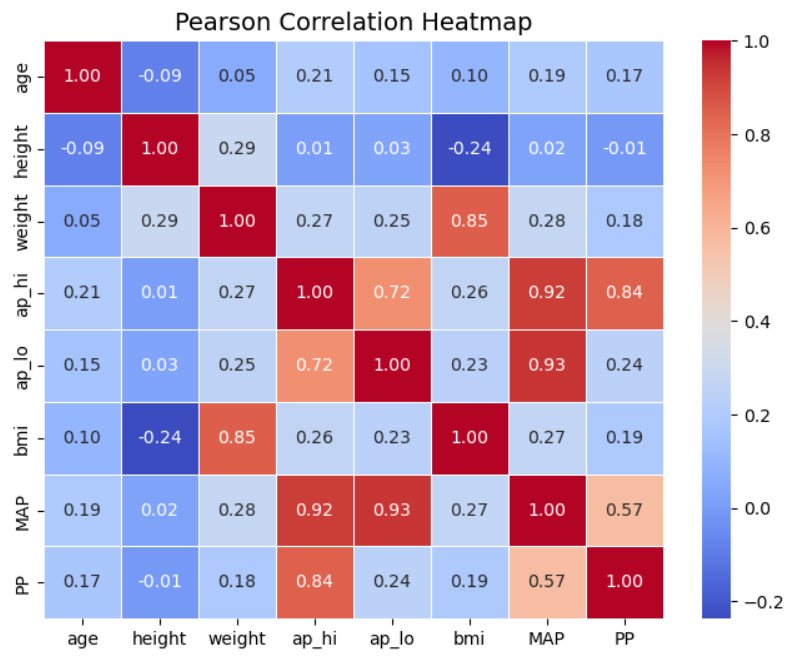


Figure 38 Pearson Correlation among features

## **3.3 Check Target column balance**

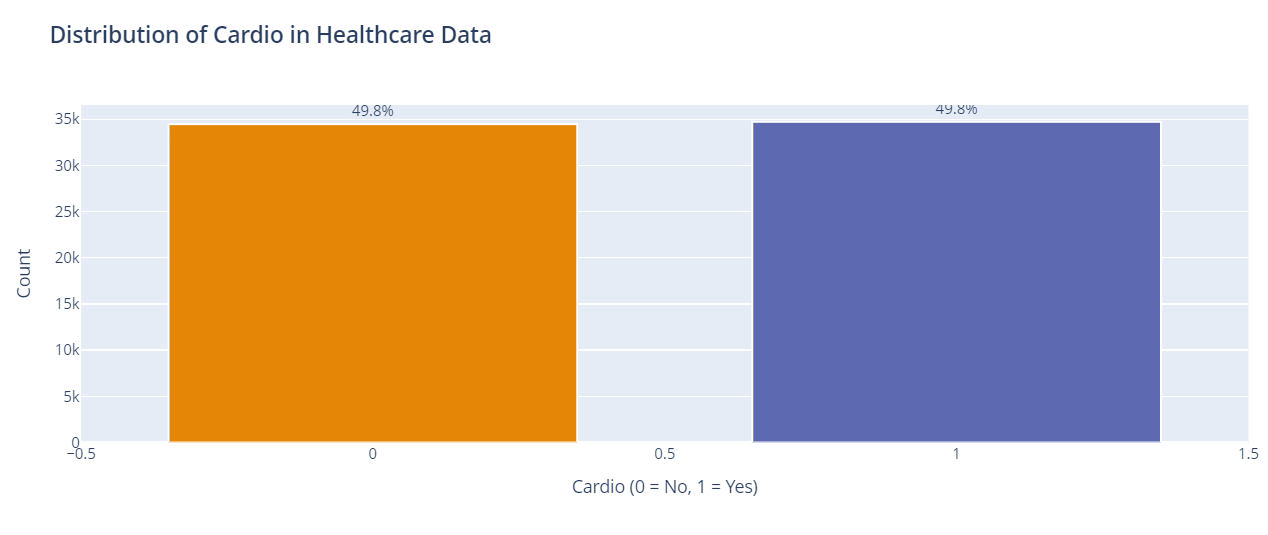


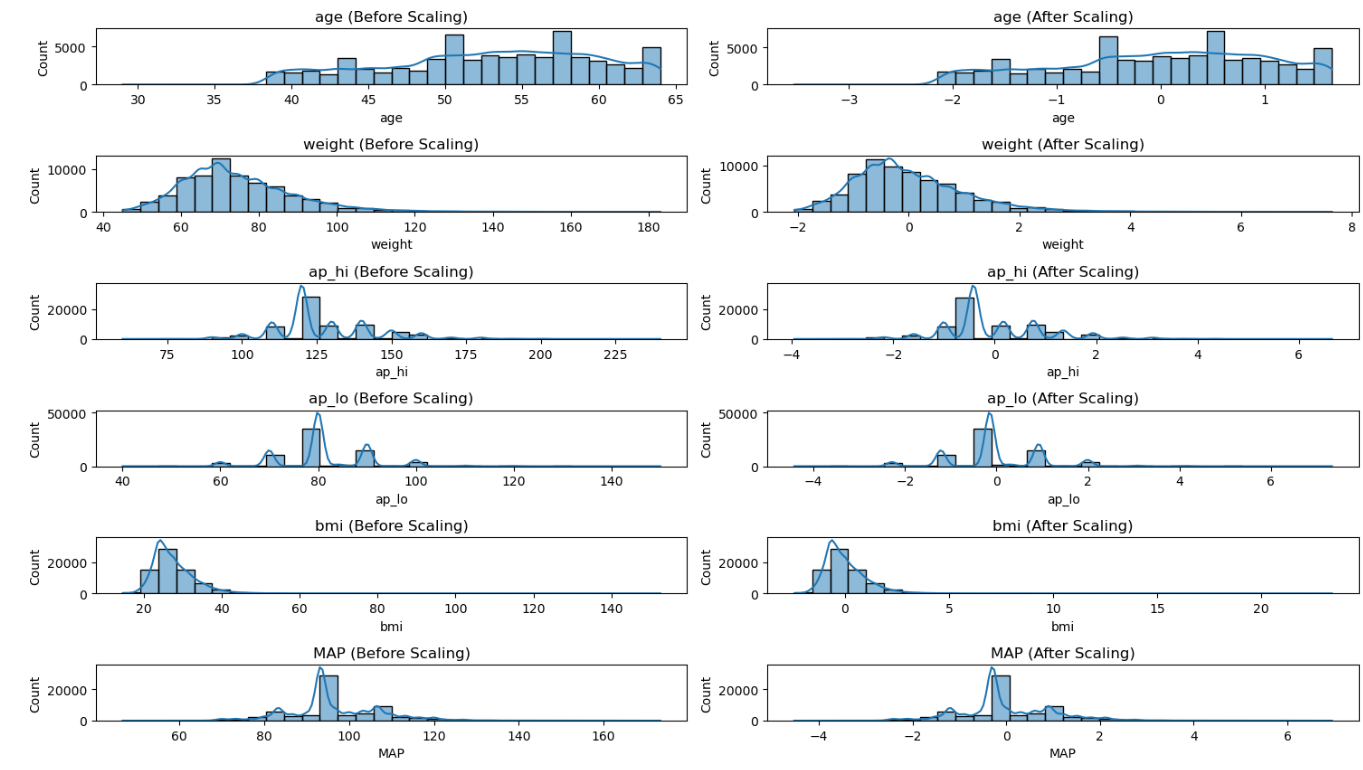
Figure 39 Cardio column

The target column is balanced.

## 

## **3.4 Data Scaling**

Applying Standard Scaler to make sure that all values within the same range



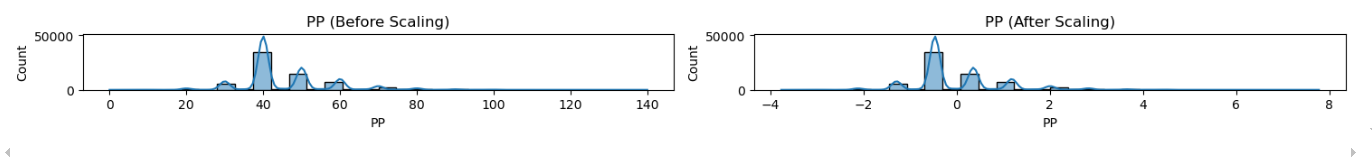
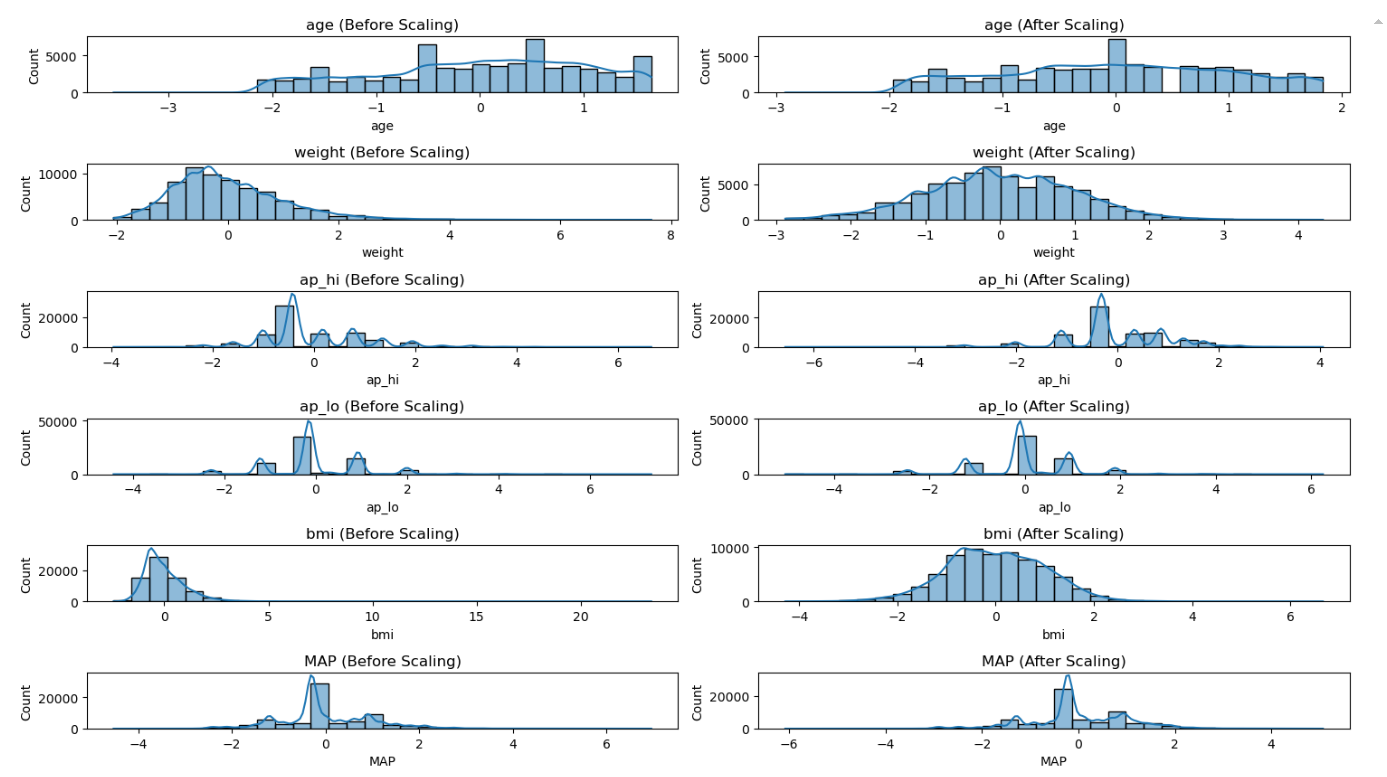


Figure 40 Data Before scaling

Applying PowerTransform to make sure that the features are gaussian distribution as there are some models require the input data has gaussian distribution.



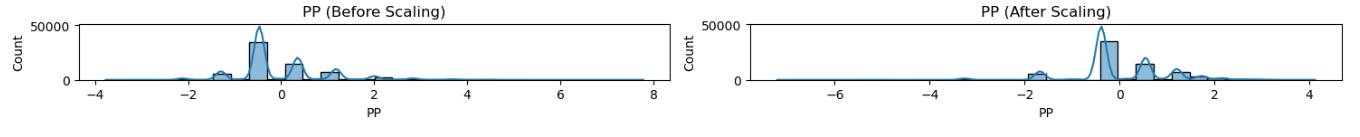


Figure 41 PowerTransform of features

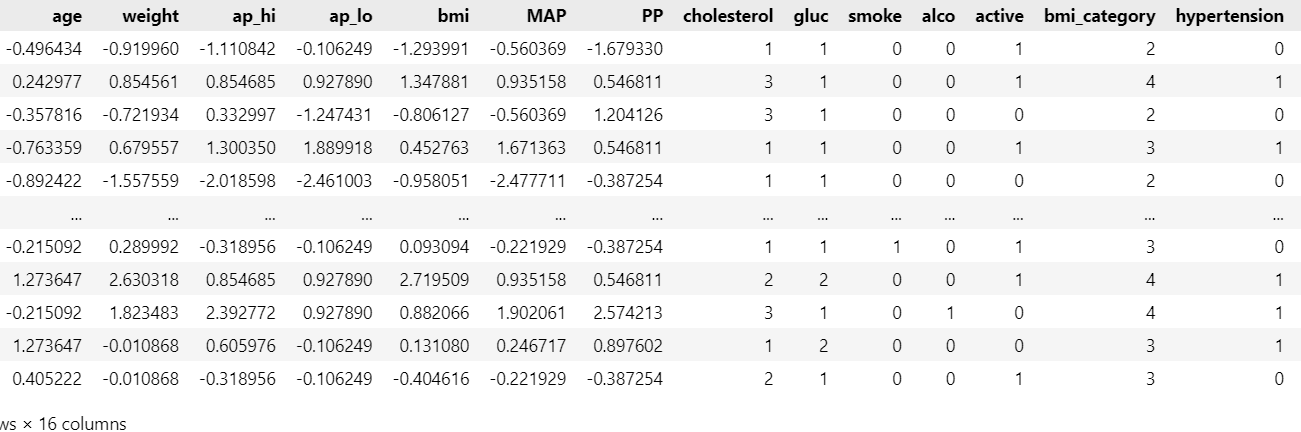


Figure 42 Sample of data after scaling

**Data Visualization using PowerBI:**

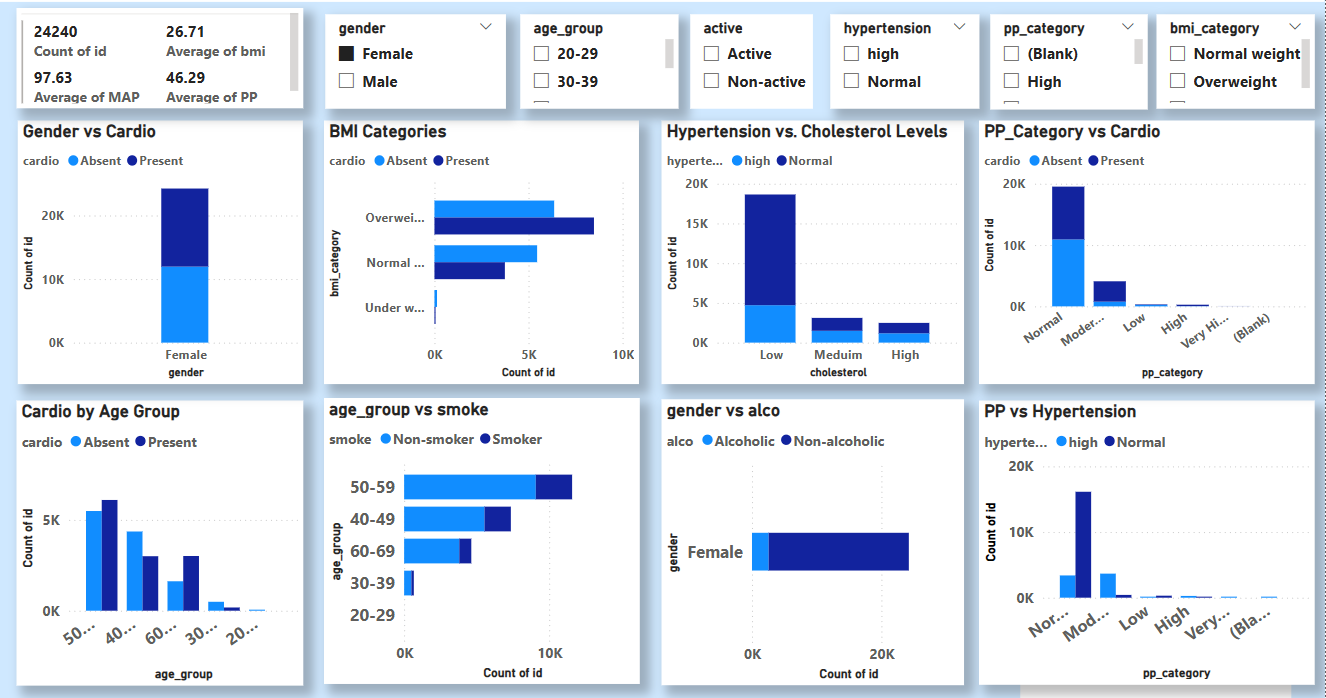
****

Figure 43 SnapShot of PowerBI

# **4. Milestone3**

## **4.1 Model Selection**

Our problem is a supervised machine learning (binary classification) so we have detected all possible models can be used in binary classification problem.

* Logistic Regression
* Random Forest
* Gradient Boosting
* XGBoost
* LightGBM
* Naive Bayes
* SVM (Support vector machine)
* Decision Tree

So our target is to filter among these models which will fit our data.

### **4.1.1 Trial 1**

Our first trial we try all these models and trained them on all features without excluding any feature

the comparison between these models.

| **Model** | **Train Accuracy** | **Test Accuracy** | **Train Precision** | **Test Precision** | **Train Recall** | **Test Recall** | **Train F1** | **Test F1** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Logistic Regression | 0.7311 | 0.7296 | 0.7770 | 0.7729 | 0.6510 | 0.6530 | 0.7084 | 0.7079 |
| Random Forest | 0.9734 | 0.7007 | 0.9816 | 0.7013 | 0.9650 | 0.7030 | 0.9732 | 0.7021 |
| Gradient Boosting | 0.7407 | 0.7351 | 0.7661 | 0.7582 | 0.6957 | 0.6930 | 0.7292 | 0.7242 |
| XGBoost | 0.7655 | 0.7369 | 0.7913 | 0.7560 | 0.7236 | 0.7022 | 0.7559 | 0.7282 |
| LightGBM | 0.7498 | 0.7380 | 0.7729 | 0.7587 | 0.7101 | 0.7007 | 0.7402 | 0.7285 |
| Naive Bayes | 0.7252 | 0.7250 | 0.7850 | 0.7824 | 0.6231 | 0.6260 | 0.6948 | 0.6955 |
| SVM | 0.7357 | 0.7340 | 0.7826 | 0.7771 | 0.6553 | 0.6589 | 0.7134 | 0.7131 |
| Decision Tree | 0.9734 | 0.6391 | 0.9929 | 0.6425 | 0.9538 | 0.6328 | 0.9730 | 0.6376 |

Based on the above analysis we are interested on both precision and recall we found that

The top 3 models in Test F1:

* Gradient Boosting
* XGBoost
* LightGBM

The top 3 models in Test Recall:

* Random Forest
* XGBoost
* LightGBM

The top 3 models in Test Precision:

* Light GBM
* Navie Bayes
* SVM

The intersection among all these matrics is → LightGBM

Initially we will take LightGBM into our consideration.

### **4.1.2 Trial 2**

We can use random forest to detect the importance of features on the target column we decided to filter on top7 out of 13 features

| **Model** | **Train Accuracy** | **Test Accuracy** | **Train Precision** | **Test Precision** | **Train Recall** | **Test Recall** | **Train F1** | **Test F1** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Logistic Regression | 0.7298 | 0.7296 | 0.7776 | 0.7738 | 0.6464 | 0.6515 | 0.7060 | 0.7074 |
| Random Forest | 0.9571 | 0.6989 | 0.9695 | 0.7006 | 0.9443 | 0.6982 | 0.9567 | 0.6994 |
| Gradient Boosting | 0.7380 | 0.7359 | 0.7594 | 0.7538 | 0.6995 | 0.7034 | 0.7283 | 0.7277 |
| XGBoost | 0.7605 | 0.7300 | 0.7846 | 0.7500 | 0.7205 | 0.6929 | 0.7512 | 0.7203 |
| LightGBM | 0.7458 | 0.7354 | 0.7670 | 0.7534 | 0.7088 | 0.7028 | 0.7368 | 0.7272 |
| Naive Bayes | 0.7246 | 0.7242 | 0.7965 | 0.7924 | 0.6062 | 0.6103 | 0.6884 | 0.6896 |
| SVM | 0.7348 | 0.7362 | 0.7654 | 0.7636 | 0.6799 | 0.6868 | 0.7201 | 0.7232 |
| Decision Tree | 0.9572 | 0.6389 | 0.9875 | 0.6460 | 0.9264 | 0.6206 | 0.9560 | 0.6330 |

But No gain from this trial.

### **4.1.3 Trial 3**

From the above two trials there are models overfitting like Random Forest and Decision Tree and there are models under fit such as Navie Bayes so it we take voting among these models this may affect the accuracy. But unfortunately it does not happen as shown below.

| **Model** | **Train Accuracy** | **Test Accuracy** | **Train Precision** | **Test Precision** | **Train Recall** | **Test Recall** | **Train F1** | **Test F1** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| voting\_clf | 0.8268 | 0.7337 | 0.8637 | 0.7647 | 0.7777 | 0.6779 | 0.8184 | 0.7187 |

The accuracy did not enhanced according the model complexity so we will neglect this model.

### **4.1.4 Trial 4**

I want to check if there is a noisy feature or redundant feature the result was so hug so I stored it in csv file to choose the best model.

After this trial we found that the best models which have the maximum F1 score are:

* LightGBM which matches the first notice without hypertension or alco column.
* Gradient Boosting when active col is dropped.

| **Dropped Feature** | **Model** | **Train Accuracy** | **Test Accuracy** | **Train Precision** | **Test Precision** | **Train Recall** | **Test Recall** | **Train F1** | **Test F1** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| hypertension | LightGBM | 0.7505 | 0.7384 | 0.7725 | 0.7572 | 0.7128 | 0.7046 | 0.7414 | 0.7299 |

| **Dropped Feature** | **Model** | **Train Accuracy** | **Test Accuracy** | **Train Precision** | **Test Precision** | **Train Recall** | **Test Recall** | **Train F1** | **Test F1** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| active | Gradient Boosting | 0.739 | 0.7372 | 0.7604 | 0.755 | 0.7008 | 0.7053 | 0.7294 | 0.7293 |
| alco | LightGBM | 0.7501 | 0.7385 | 0.7728 | 0.7592 | 0.711 | 0.7012 | 0.7406 | 0.7291 |

### **4.1.5 Trial 5**

Choosing the best parameters for LightGBM the results where as follow: (all feature included)

Best Parameters of the model:

### **📈 Model Performance:**

* **Best CV Score:** 0.7377
* **Precision:** 0.7567
* **Recall:** 0.7027
* **F1 Score:** 0.7287

### **📋 Classification Report:**

| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
| --- | --- | --- | --- | --- |
| **0** | 0.72 | 0.77 | 0.75 | 6892 |
| **1** | 0.76 | 0.70 | 0.73 | 6942 |
|  |  |  |  |  |
| **Accuracy** |  |  | **0.74** | 13834 |
| **Macro Avg** | 0.74 | 0.74 | 0.74 | 13834 |
| **Weighted Avg** | 0.74 | 0.74 | 0.74 | 13834 |

### **4.1.6 Trial 6**

It is required to try model Gradient boosting classifier (with active and without active feature)

They have the same result.

### **📈 Model Performance:**

| **Metric** | **Value** |
| --- | --- |
| Precision | 0.7549 |
| Recall | 0.7040 |
| F1 Score | 0.7285 |

### **📋 Classification Report:**

| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
| --- | --- | --- | --- | --- |
| **0** | 0.72 | 0.77 | 0.74 | 6892 |
| **1** | 0.75 | 0.70 | 0.73 | 6942 |
|  |  |  |  |  |
| **Accuracy** |  |  | **0.74** | 13834 |
| **Macro Avg** | 0.74 | 0.74 | 0.74 | 13834 |
| **Weighted Avg** | 0.74 | 0.74 | 0.74 | 13834 |

### **4.1.7 Trial 7**

Trained LightGBM model without alco column. the results as attached.

### **📈 Model Performance:**

| **Metric** | **Value** |
| --- | --- |
| Best CV Score | 0.7376 |
| Precision | 0.7568 |
| Recall | 0.7028 |
| F1 Score | 0.7288 |

### **📋 Classification Report:**

| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
| --- | --- | --- | --- | --- |
| **0** | 0.72 | 0.77 | 0.75 | 6892 |
| **1** | 0.76 | 0.70 | 0.73 | 6942 |
|  |  |  |  |  |
| **Accuracy** |  |  | **0.74** | 13834 |
| **Macro Avg** | 0.74 | 0.74 | 0.74 | 13834 |
| **Weighted Avg** | 0.74 | 0.74 | 0.74 | 13834 |

We will choose this model as by logic the most of data is for normal people so by logic recall for normal people (cardio=0) is more than recall for patient people (cardio=1).

it is used for optimization **Randomsearch.**

# **5. MileStone 4**

## **5.1 mlflow for tracking models**

mlflow was applied on the code and the code attached in Milestone4.

this is snapshot of tracing which we have done for models.

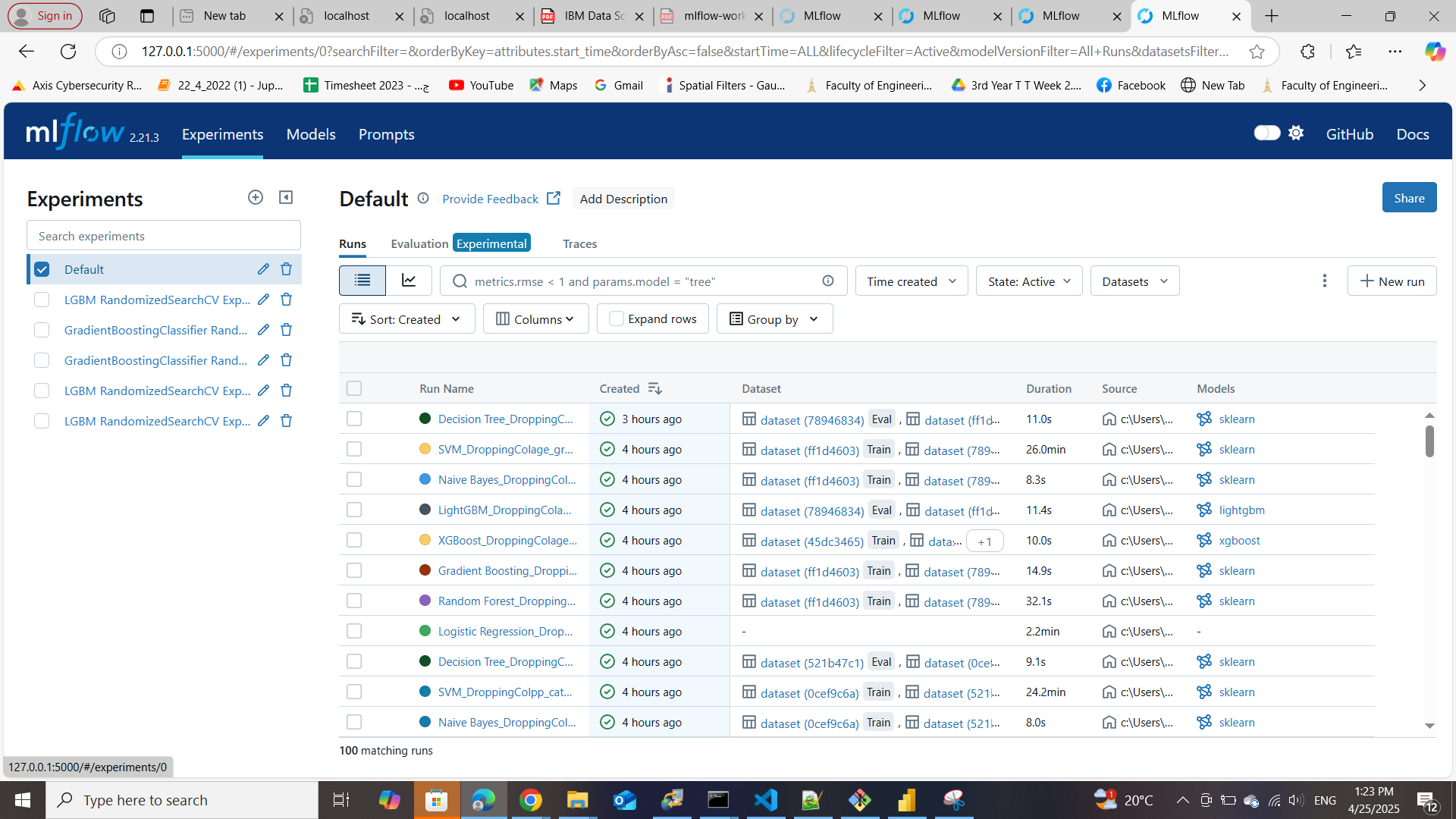


Figure 44 mlflow to track the code.

## **5.2 Demplyment**

When I run streamlit run main\_func.py in command prompt this window will appear and takes inputs from the user then predict as follow.

Noting that the model was deployed as pkl file, standardization as pkl file.

