Capstone Project - 3

Cardiovascular Risk Prediction

Team Member

Vinay Kumar
Nikhil Lamje

Let’s Predict the Cases

1. Defining Problem Statement
2. Exploratory Data Analysis
3. Feature Engineering
4. Preparation of Dataset for Modeling
5. Applying Model
6. Model Validation and Selection

What is the need

- World Health Organization has estimated 12 million deaths occur worldwide, every year due to Heart diseases. Half the deaths in the United States and other developed countries are due to cardio vascular diseases. The early prognosis of cardiovascular diseases can aid in making decisions on lifestyle changes in high risk patients and in turn reduce the complications.

Variables

- Our DataFrame has 3389 records and 17 columns.
- Each attribute is a potential risk factor. There are both demographic, behavioural and medical risk factors.

Categorical Data →

1. **sex values:** ['F' 'M']
2. **is_smoking values:** ['YES' 'NO']
3. **BPMeds values:** whether or not the patient was on blood pressure medication. [0. 1.]
4. **prevalentStroke values:** wheater or not the patient had a stroke in past [0 1]
5. **prevalentHyp values:** whether or not the patient was hypertensive[0 1]
6. **diabetes values:** [0 1]
7. **TenYearCHD values:** 10 year risk of coronary heart disease [1 0]

Data Summary

Numerical Data

1. Age
2. cigsPerDay : the number of cigarettes that a person smoked on average in a day.
3. totChol : total cholesterol level.
4. sysBP : systolic blood pressure.
5. diaBP : diastolic blood pressure.
6. BMI : Body Mass Index
7. heartRate : heart rate
8. glucose : glucose level.

Features

Independent Features →

1. age
2. cigsPerDay
3. BPMeds
4. prevalentHyp
5. totchol
6. HeartRate
7. Glucose
8. Sex
9. BP : diaBP + sysBP

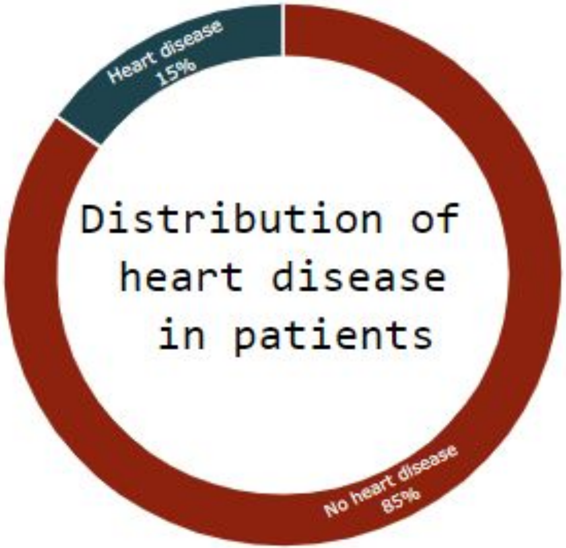
Dependent Feature →

1. TenYearCHD

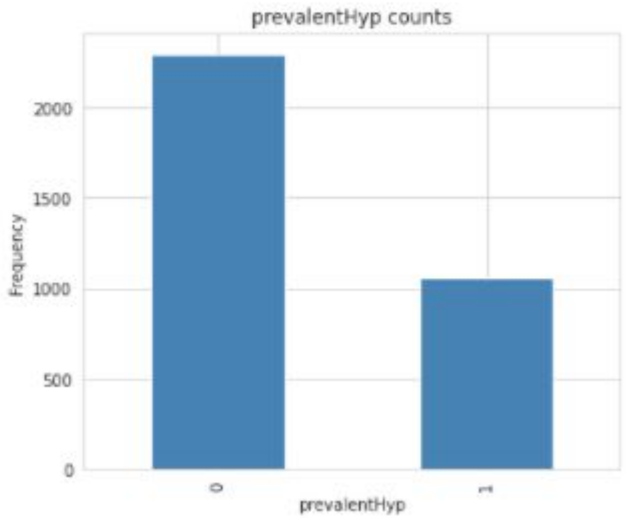
Dependent Variable

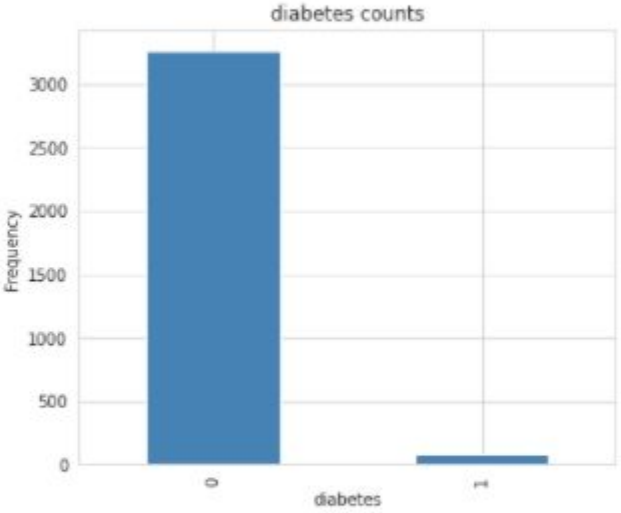
TenYearCHD

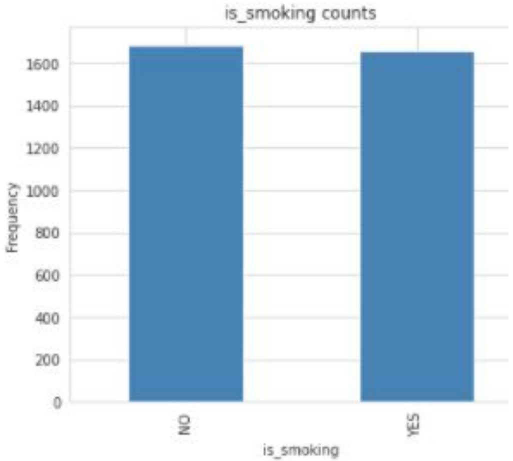
- We can Class Imbalance in our dependent variable with 2848 records of class 0 and only 504 records of class 1.
- To tackle this class imbalance problem, we can perform SMOTE (Synthetic Minority Oversampling Technique) to overcome the overfitting problem posed by random oversampling.



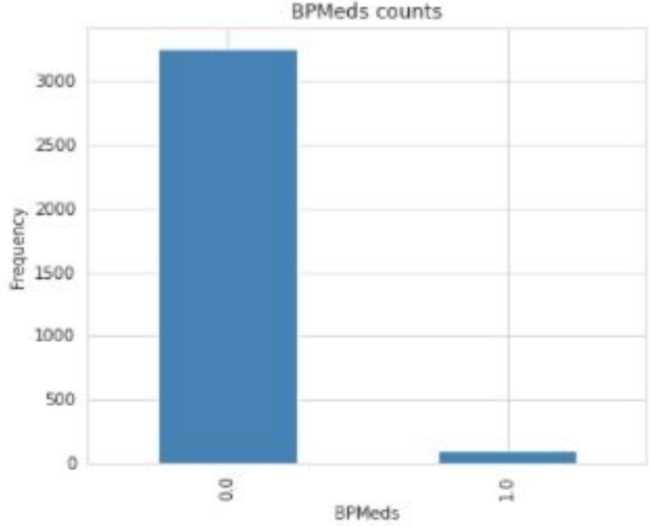
EDA

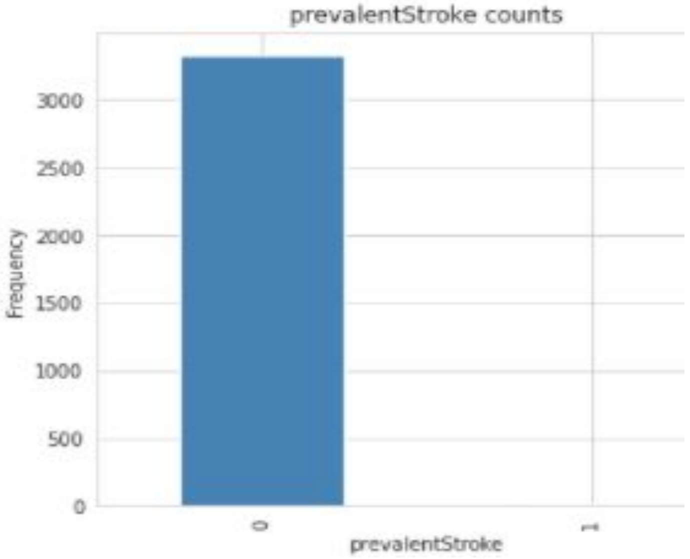

prevalentHyp counts
Frequency
0
500
1000
1500
2000
0
1
prevalentHyp


diabetes counts
Frequency
0
500
1000
1500
2000
2500
3000
0
1
diabetes

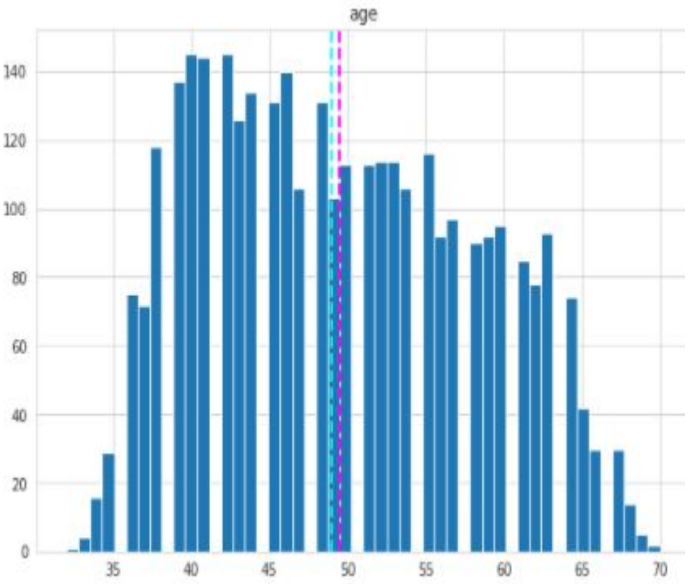

is_smoking counts
Frequency
0
200
400
600
800
1000
1200
1400
1600
NO
YES
is_smoking

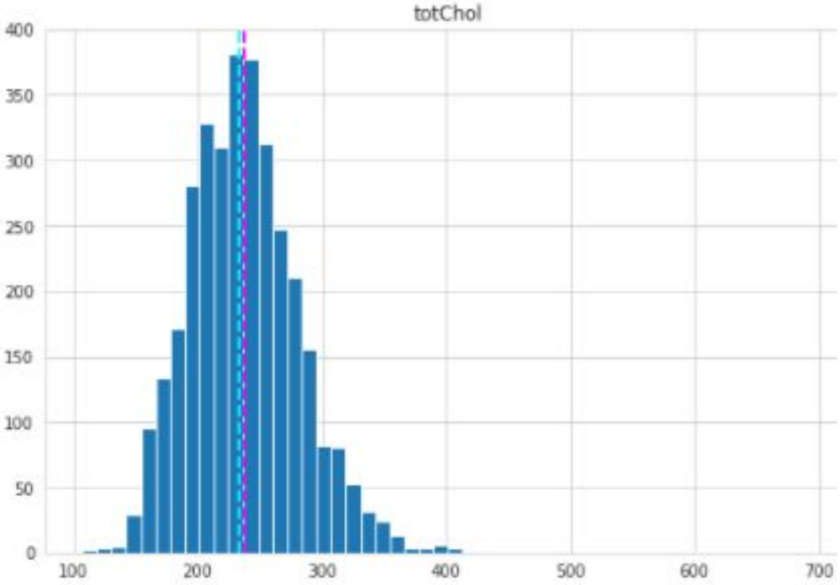
EDA


BPMeds counts
Frequency
0
500
1000
1500
2000
2500
3000
0.0
1.0
BPMeds

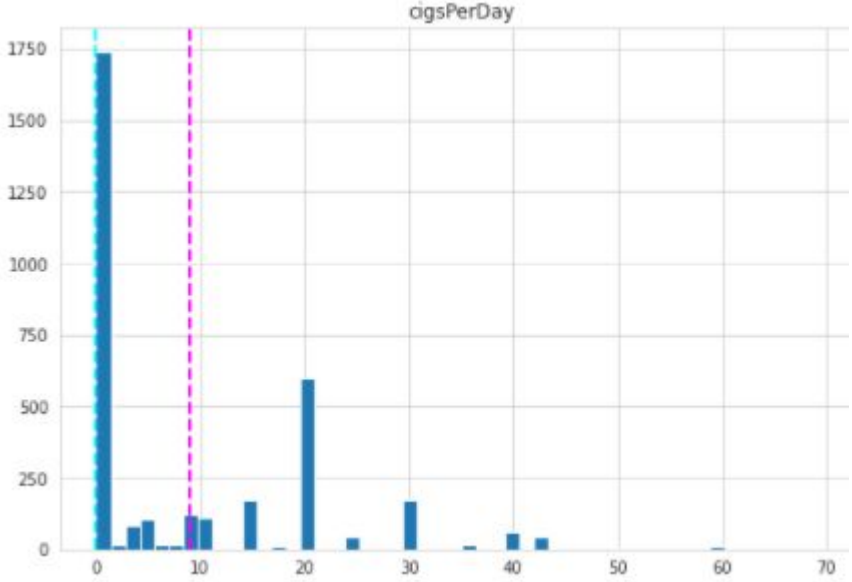

prevalentStroke counts
Frequency
0
500
1000
1500
2000
2500
3000
0
1
prevalentStroke

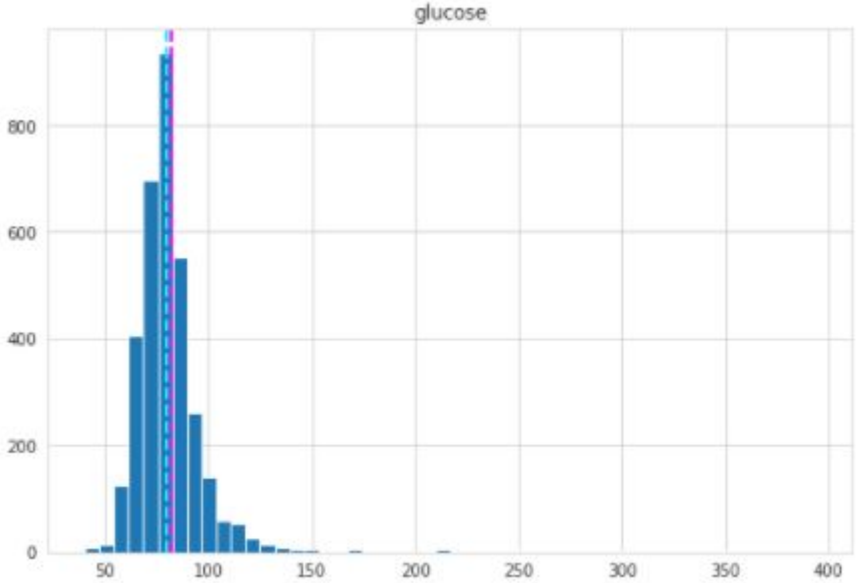
EDA


age
0
20
40
60
80
100
120
140
35
40
45
50
55
60
65
70

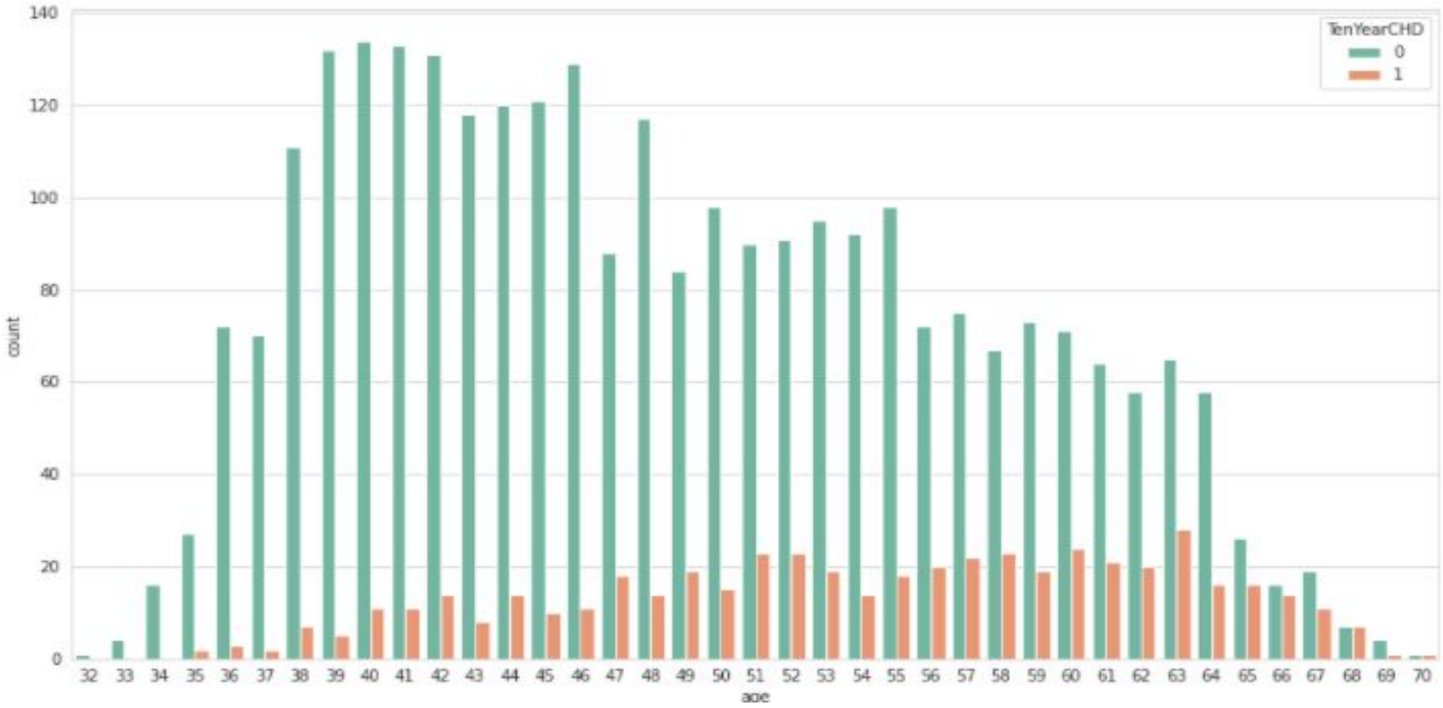

totChol
0
50
100
150
200
250
300
350
400
100
200
300
400
500
600
700

EDA


cigsPerDay
1750
1500
1250
1000
750
500
250
0
0
10
20
30
40
50
60
70

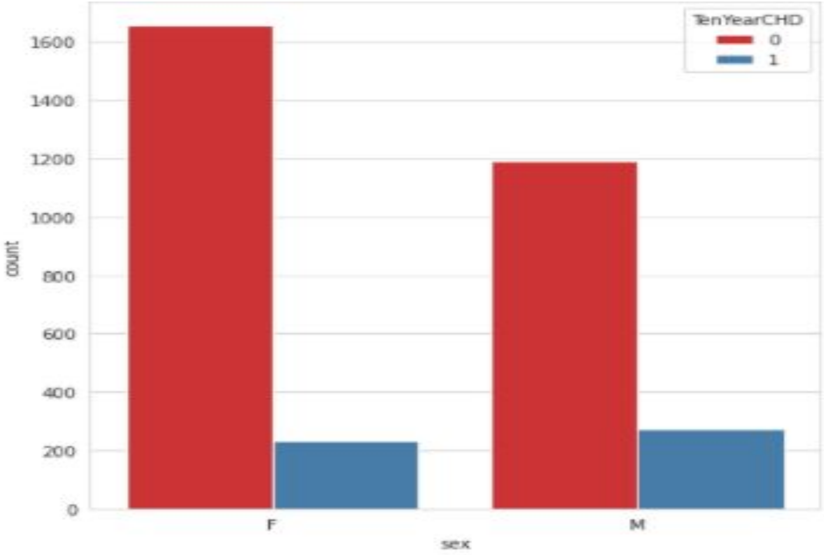

glucose
800
600
400
200
0
50
100
150
200
250
300
350
400

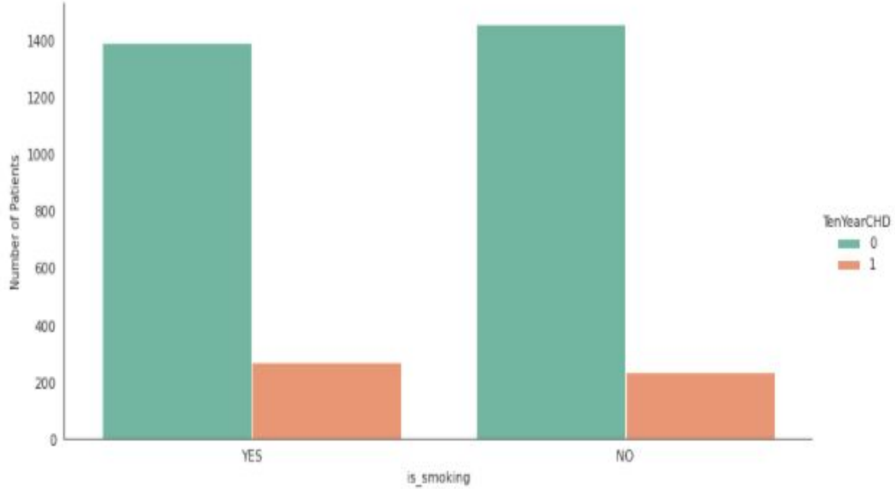
EDA



- People between the ages of 35 and 50 are less likely to develop chronic health problems.

EDA

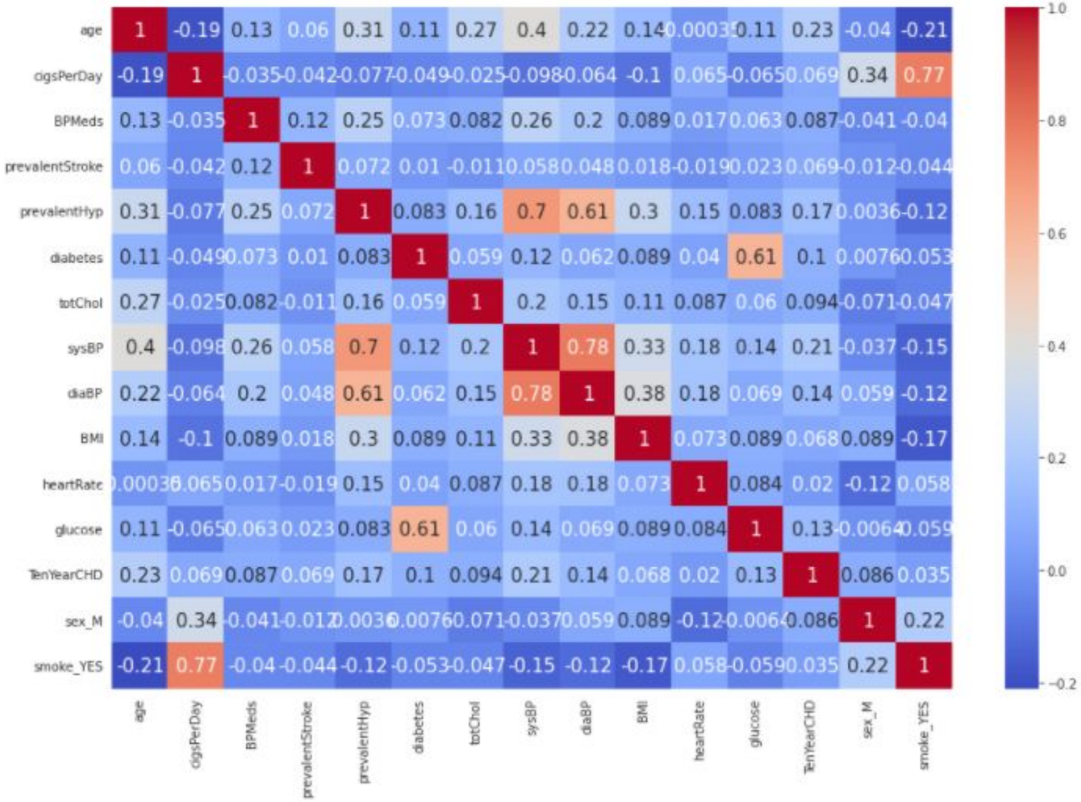




* We don't detect a significant difference in the risk of heart disease among smokers and non smokers.

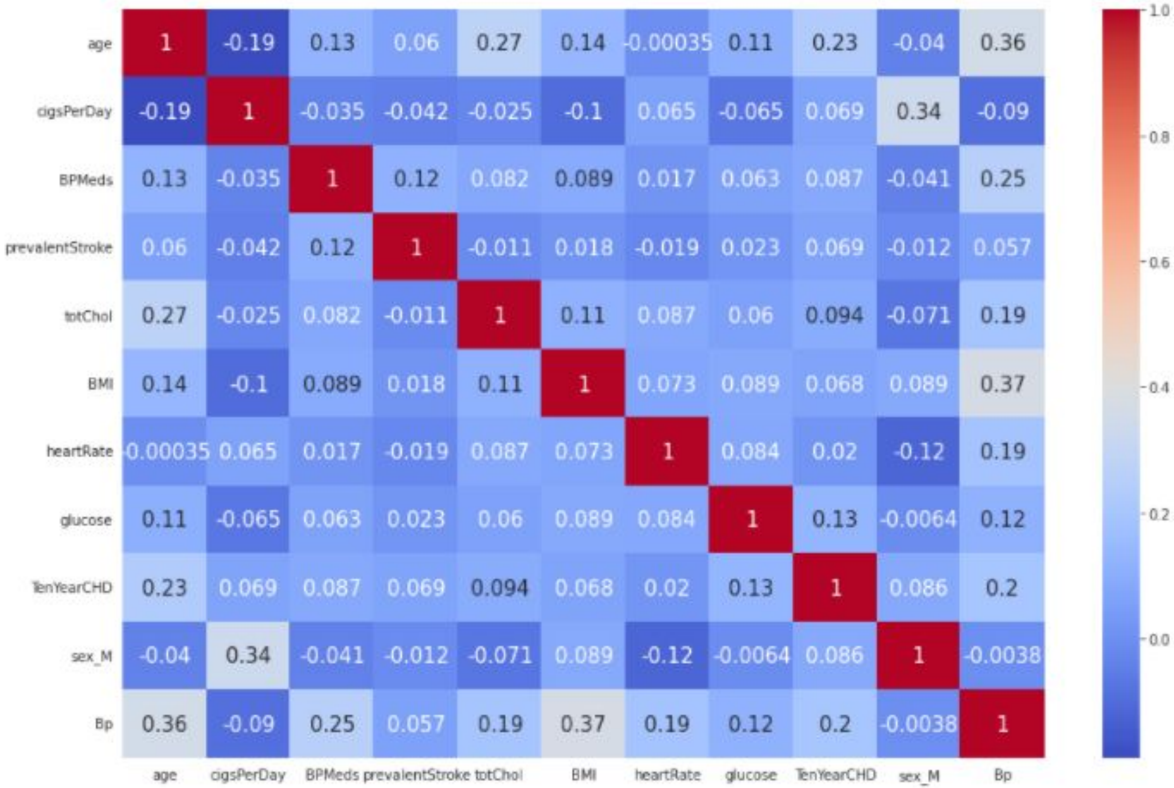
Feature Selection

- Because there is a strong link between sysBP and diaBP, we can add these two columns to make a new Bp column.
- smoke_yes and cigsPerDay are also strongly co-related. So, we can drop smoke_yes.
- Glucose and diabetes are also strongly strongly co-related. So, we can drop diabetes.



	age	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	BMI	heartRate	glucose	TenYearCHD	sex_M	smoke_YES
age	1	-0.19	0.13	0.06	0.31	0.11	0.27	0.4	0.22	0.14	0.00036	0.11	0.23	-0.04	-0.21
cigsPerDay	-0.19	1	-0.035	-0.042	-0.077	-0.049	-0.025	-0.098	-0.064	-0.1	0.065	-0.065	0.069	0.34	0.77
BPMeds	0.13	-0.035	1	0.12	0.25	0.073	0.082	0.26	0.2	0.089	0.017	0.063	0.087	-0.041	-0.04
prevalentStroke	0.06	-0.042	0.12	1	0.072	0.01	-0.011	0.058	0.048	0.018	-0.019	0.023	0.069	-0.012	-0.044
prevalentHyp	0.31	-0.077	0.25	0.072	1	0.083	0.16	0.7	0.61	0.3	0.15	0.083	0.17	0.0036	-0.12
diabetes	0.11	-0.049	0.073	0.01	0.083	1	0.059	0.12	0.062	0.089	0.04	0.61	0.1	0.0076	0.053
totChol	0.27	-0.025	0.082	-0.011	0.16	0.059	1	0.2	0.15	0.11	0.087	0.06	0.094	-0.071	-0.047
sysBP	0.4	-0.098	0.26	0.058	0.7	0.12	0.2	1	0.78	0.33	0.18	0.14	0.21	-0.037	-0.15
diaBP	0.22	-0.064	0.2	0.048	0.61	0.062	0.15	0.78	1	0.38	0.18	0.069	0.14	0.059	-0.12
BMI	0.14	-0.1	0.089	0.018	0.3	0.089	0.11	0.33	0.38	1	0.073	0.089	0.068	0.089	-0.17
heartRate	0.00036	0.065	0.017	-0.019	0.15	0.04	0.087	0.18	0.18	0.073	1	0.084	0.02	-0.12	0.058
glucose	0.11	-0.065	0.063	0.023	0.083	0.61	0.06	0.14	0.069	0.089	0.084	1	0.13	-0.0064	0.059
TenYearCHD	0.23	0.069	0.087	0.069	0.17	0.1	0.094	0.21	0.14	0.068	0.02	0.13	1	0.086	0.035
sex_M	-0.04	0.34	-0.041	-0.012	0.0036	0.0076	-0.071	-0.037	0.059	0.089	-0.12	-0.0064	0.086	1	0.22
smoke_YES	-0.21	0.77	-0.04	-0.044	-0.12	-0.053	-0.047	-0.15	-0.12	-0.17	0.058	-0.059	0.035	0.22	1

Feature Selection

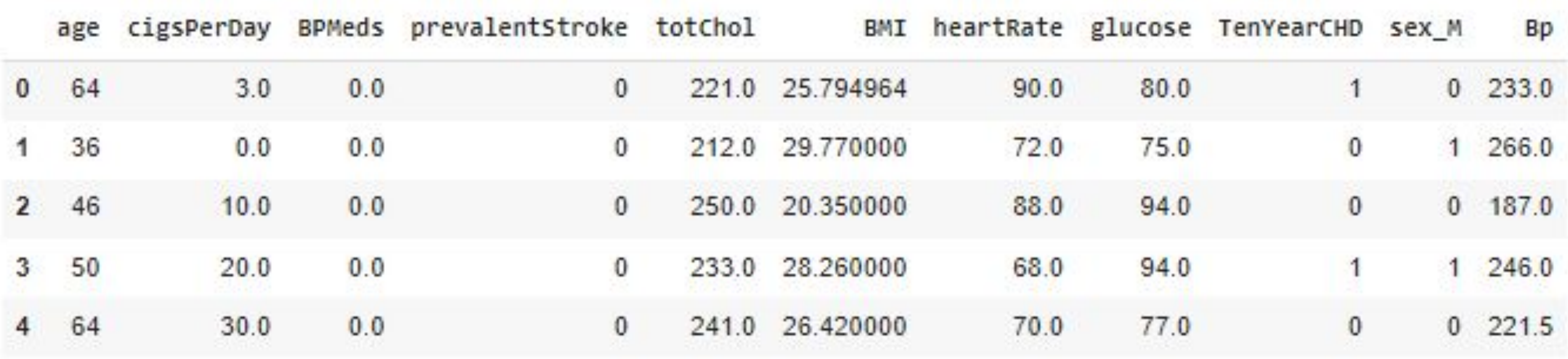


	age	cigsPerDay	BPMeds	prevalentStroke	totChol	BMI	heartRate	glucose	TenYearCHD	sex_M	Bp
age	1	-0.19	0.13	0.06	0.27	0.14	-0.00035	0.11	0.23	-0.04	0.36
cigsPerDay	-0.19	1	-0.035	-0.042	-0.025	-0.1	0.065	-0.065	0.069	0.34	-0.09
BPMeds	0.13	-0.035	1	0.12	0.082	0.089	0.017	0.063	0.087	-0.041	0.25
prevalentStroke	0.06	-0.042	0.12	1	-0.011	0.018	-0.019	0.023	0.069	-0.012	0.057
totChol	0.27	-0.025	0.082	-0.011	1	0.11	0.087	0.06	0.094	-0.071	0.19
BMI	0.14	-0.1	0.089	0.018	0.11	1	0.073	0.089	0.068	0.089	0.37
heartRate	-0.00035	0.065	0.017	-0.019	0.087	0.073	1	0.084	0.02	-0.12	0.19
glucose	0.11	-0.065	0.063	0.023	0.06	0.089	0.084	1	0.13	-0.0064	0.12
TenYearCHD	0.23	0.069	0.087	0.069	0.094	0.068	0.02	0.13	1	0.086	0.2
sex_M	-0.04	0.34	-0.041	-0.012	-0.071	0.089	-0.12	-0.0064	0.086	1	-0.0038
Bp	0.36	-0.09	0.25	0.057	0.19	0.37	0.19	0.12	0.2	-0.0038	1

Preparing Dataset for Modelling

Train Set - (2346,10)

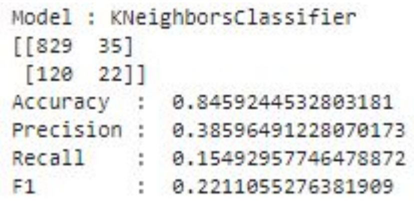
Test Set - (1006,10)



	age	cigsPerDay	BPMeds	prevalentStroke	totChol	BMI	heartRate	glucose	TenYearCHD	sex_M	Bp
0	64	3.0	0.0	0	221.0	25.794964	90.0	80.0	1	0	233.0
1	36	0.0	0.0	0	212.0	29.770000	72.0	75.0	0	1	266.0
2	46	10.0	0.0	0	250.0	20.350000	88.0	94.0	0	0	187.0
3	50	20.0	0.0	0	233.0	28.260000	68.0	94.0	1	1	246.0
4	64	30.0	0.0	0	241.0	26.420000	70.0	77.0	0	0	221.5

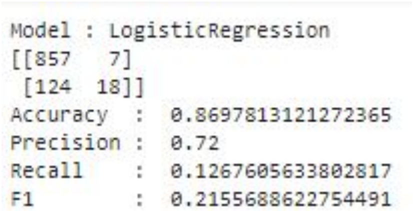
Applying Models

KNeighbors



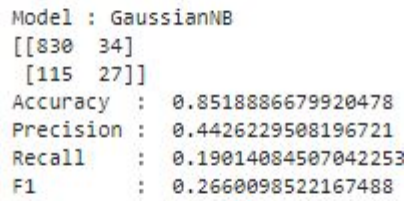
```
Model : KNeighborsClassifier
[[829  35]
 [120  22]]
Accuracy  :  0.8459244532803181
Precision :  0.38596491228070173
Recall    :  0.15492957746478872
F1        :  0.2211055276381909
```

Logistic Regression



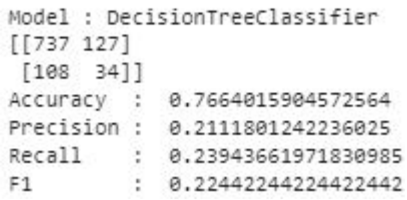
```
Model : LogisticRegression
[[857   7]
 [124  18]]
Accuracy  :  0.8697813121272365
Precision :  0.72
Recall    :  0.1267605633802817
F1        :  0.2155688622754491
```

GuassianNB



```
Model : GaussianNB
[[830  34]
 [115  27]]
Accuracy  :  0.8518886679920478
Precision :  0.4426229508196721
Recall    :  0.19014084507042253
F1        :  0.2660098522167488
```

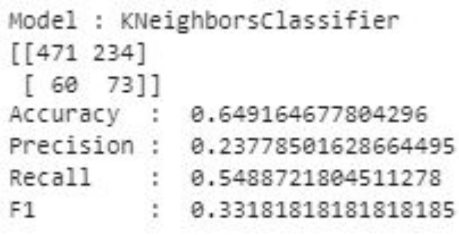
DecisionTree



```
Model : DecisionTreeClassifier
[[737 127]
 [108  34]]
Accuracy  :  0.7664015904572564
Precision :  0.2111801242236025
Recall    :  0.23943661971830985
F1        :  0.22442244224422442
```

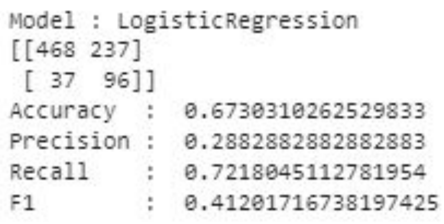
Applying Models (SMOTE)

KNeighbors



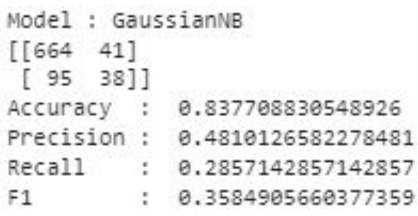
```
Model : KNeighborsClassifier
[[471 234]
 [ 60  73]]
Accuracy  :  0.649164677804296
Precision :  0.23778501628664495
Recall    :  0.5488721804511278
F1        :  0.33181818181818185
```

Logistic Regression



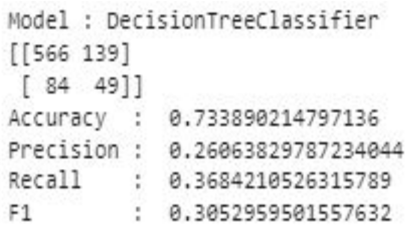
```
Model : LogisticRegression
[[468 237]
 [ 37  96]]
Accuracy  :  0.6730310262529833
Precision :  0.2882882882882883
Recall    :  0.7218045112781954
F1        :  0.41201716738197425
```

GuassianNB



```
Model : GaussianNB
[[664  41]
 [ 95  38]]
Accuracy  :  0.837708830548926
Precision :  0.4810126582278481
Recall    :  0.2857142857142857
F1        :  0.3584905660377359
```

DecisionTree

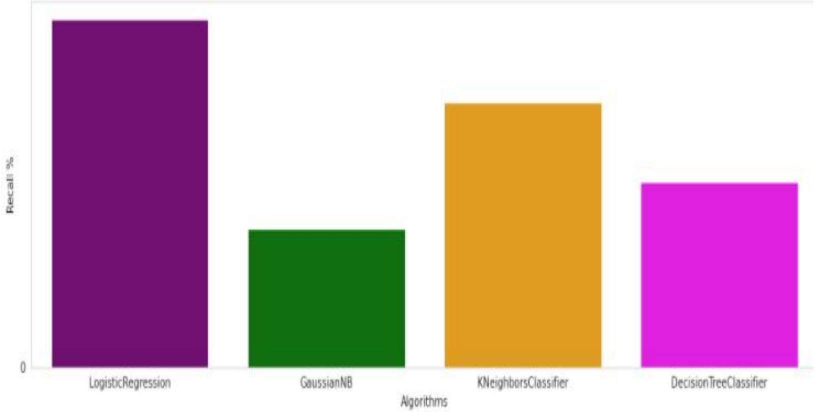


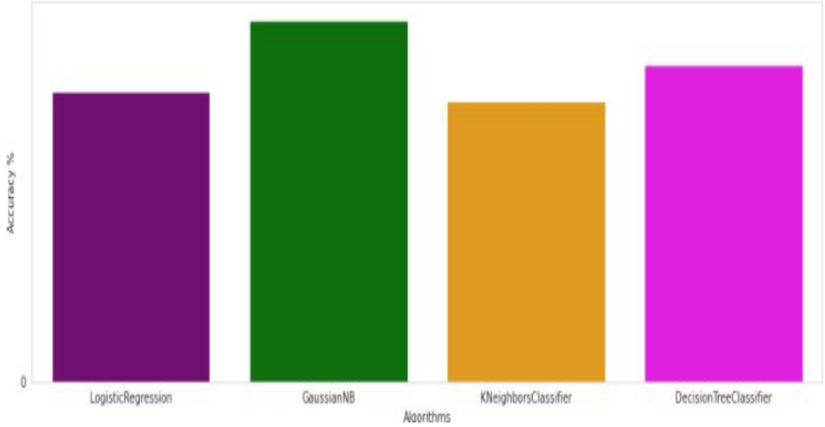
```
Model : DecisionTreeClassifier
[[566 139]
 [ 84  49]]
Accuracy  :  0.733890214797136
Precision :  0.26063829787234044
Recall    :  0.3684210526315789
F1        :  0.3052959501557632
```

Model Performance

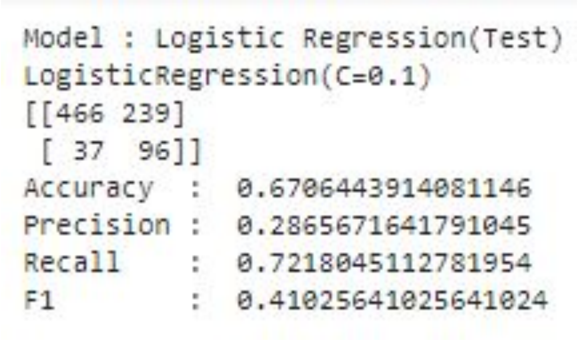
- the recall in GaussianNB is 0.28, which is the lowest of all of the models.

- These two models are selected for hyperparameter tweaking since they have recall values of 0.54 and 0.72 from KNeighborsClassifier and Logistic Regression, respectively..



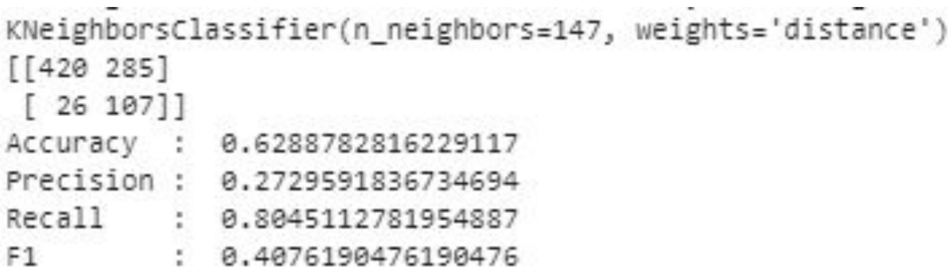


Hyperparameter Optimization



```
Model : Logistic Regression(Test)
LogisticRegression(C=0.1)
[[466 239]
 [ 37  96]]
Accuracy  :  0.6706443914081146
Precision :  0.2865671641791045
Recall    :  0.7218045112781954
F1        :  0.41025641025641024
```

Logistic Regression

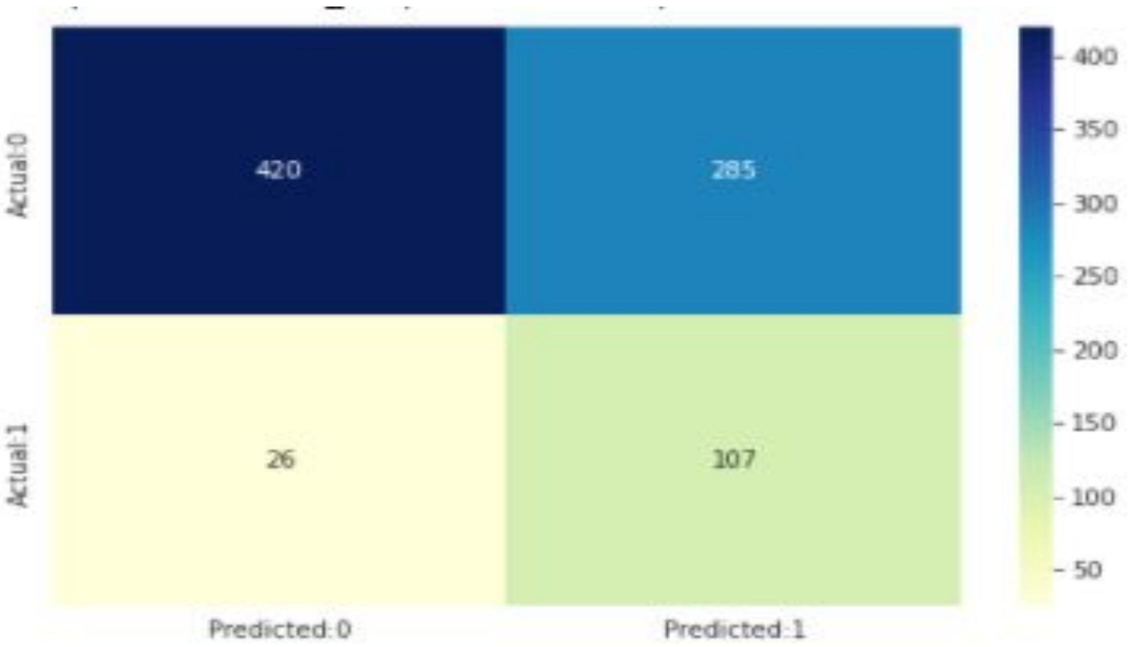


```
KNeighborsClassifier(n_neighbors=147, weights='distance')
[[420 285]
 [ 26 107]]
Accuracy  :  0.6288782816229117
Precision :  0.2729591836734694
Recall    :  0.8045112781954887
F1        :  0.4076190476190476
```

KNeighborsClassifier

- The best results were obtained utilising hyperparameter adjustments, with recall as high as 0.80 with KNeighborsClassifier.

Confusion Matrix



	Predicted:0	Predicted:1
Actual:0	420	285
Actual:1	26	107

Conclusion

- This study proposed the use machine learning techniques to identify the person with Cardiovascular Disease Risk in future.

- Indeed, selecting the appropriate features to get the intended outcome was crucial, with the glucose and diabetes features being the most closely related to the target variable.

- During the model development, we found that our dependent exhibits a class imbalance. We then performed SMOTE on the data to tackle the class imbalance and got the highest recall value with KNeighborsClassifier.

- By tweaking the hyperparameters of the KNeighborsClassifier, we were able to achieve the desired outcomes, with recall value of 0.80 on the test data, up from 0.54 earlier.