**Cardiovascular Risk Prediction**

**Vikash Kumar**

**Data science trainees,**

**AlmaBetter, Bangalore**

**Abstract:**

The dataset is from an ongoing cardiovascular study on residents of the town of Framingham, Massachusetts. The classification goal is to predict whether the patient has a 10-year risk of future coronary heart disease (CHD). The dataset provides the patients’ information. It includes over 4,000 records and 15 attributes.

Our experiment can help understand what could be the reason for the classification of such labels by feature selection, data analysis and prediction with machine learning algorithms taking into account previous trends to determine the correct classification.

**1.Problem Statement**

The dataset includes over 4,000 records and 15 attributes. Each attribute is a potential risk factor. There are both demographic, behavioral, and medical risk factors.

**Demographic:**

**• Sex:** male or female("M" or "F")

**• Age:** Age of the patient;(Continuous - Although the recorded ages have been truncated to whole numbers, the concept of age is continuous)

**Behavioral:**

**• is\_smoking:** whether or not the patient is a current smoker ("YES" or "NO")

**• Cigs Per Day:** the number of cigarettes that the person smoked on average in one day.(can be considered continuous as one can have any number of cigarettes, even half a cigarette.)

**Medical( history):**

**• BP Meds:** whether or not the patient was on blood pressure medication (Nominal)

• **Prevalent Stroke:** whether or not the patient had previously had a stroke (Nominal)

• **Prevalent Hyp:** whether or not the patient was hypertensive (Nominal)

**• Diabetes:** whether or not the patient had diabetes (Nominal)

**Medical(current):**

• **Tot Chol:** total cholesterol level (Continuous)

• **Sys BP:** systolic blood pressure (Continuous)

• **Dia BP:** diastolic blood pressure (Continuous)

• **BMI:** Body Mass Index (Continuous)

• **Heart Rate:** heart rate (Continuous - In medical research, variables such as heart rate though in fact discrete, are considered continuous because of the large number of possible values.)

**• Glucose:** glucose level (Continuous) Predict variable (desired target)

**TenYearCHD:** 10-year risk of coronary heart disease CHD(binary: “1”, means “Yes”, “0” means “No”) - DV

**2. Data Cleaning**

Prior to EDA, cleaning the data is essential since it will get rid of any ambiguous information that can have an impact on the results.

Education, cigsPerDay, BPMeds , totChol , BMI , heartRate and glucose columns have missing or null values. I have filled these columns for missing values by using KNN imputer.

For columns sex and is\_smoking, I have changed the categorical value with a numerical value(like yes to 1 and no to 0, and F to 1 and M to 0).

## **3. Exploratory Data Analysis**

Most people smoke between 0 and 10 cigarettes a day.

The risk of coronary heart disease rises with age up to age 63 and then declines beyond that.

Men have a higher propensity for developing coronary heart disease (CHD) than women. Women, in contrast, are at a higher risk of stroke, which often occurs at an older age.

Compared to nonsmokers, smokers have a higher chance of developing coronary heart disease.

## **4. Feature Engineering**

**Impact of heart rate on the target variable**:

People with high heart rates have a high risk of having CHD in the next 10 years.

**Impact of cholesterol level on the target variable**:

People with high cholesterol levels have a high risk of having CHD in the next 10 years.

**Impact of age on the target variable**:

Older people have a high risk of having CHD in the coming 10 years.

**Impact of Body Mass Index on the target variable**:

People with high body mass index have a high risk of having CHD in the next 10 years.

**Impact of systolic blood pressure on the target variable:**

People with high systolic blood pressure have a high risk of having CHD in the next 10 years.

**Impact of Glucose on the target variable:**

People with high Glucose have a high risk of having CHD in the next 10 years.

**5. Data preparation**

The high correlation between :

# 1. Cigs Per Day and is\_smoking

# 2. SysBP and Prevalent Hyp

3. DiaBP and SysBP

Combined SysBp and DiaBP to denote a new feature pulse rate.

# Dropping Cigs Per Day, is\_smoking, SysBP, Prevalent Hyp, and DiaBP columns from the dataset.

# Added the columns like pulse pressure, age bucket, and BMI bucket.

# The dependent column will be predicted as that is the target variable named “Ten Year CHD.

**6. Steps involved:**

* **Exploratory Data Analysis**

After loading the dataset we performed this method by comparing our target variable that is 10-year risk of future coronary heart disease (CHD) with other independent variables. This process helped us figuring out various aspects and relationships among the target and the independent variables. It gave us a better idea of which feature behaves in which manner compared to the target variable.

* **Null values Treatment**

Our dataset contains a large number of null values which might tend to disturb our accuracy hence used a KNN-imputer to perform missing value imputation, and processed data to remove outliers.

* **Encoding of categorical columns**

We used One Hot Encoding to produce binary integers of 0 and 1 to encode our categorical features because categorical features that are in string format cannot be understood by the machine and needs to be converted to numerical format.

* **Feature Selection**

In these steps we used algorithms like ExtraTree classifier to check the results of each feature i.e which feature is more important compared to our model and which is of less importance.

Next we used Chi2 for categorical features and ANOVA for numerical features to select the best feature which we will be using further in our model.

* **Standardization of features**

Our main motive through this step was to scale our data into a uniform format that would allow us to utilize the data in a better way while performing fitting and applying different algorithms to it.

The basic goal was to enforce a level of consistency or uniformity to certain practices or operations within the selected environment.

* **Fitting different models**

For modelling we tried various classification algorithms like:

1. **Logistic Regression**
2. **SVM Classifier**
3. **Random Forest Classifier**
4. **Grid Search CV**

* **SMOTE boosting for features**

To solve the issue of class imbalance, SMOTE boosting was used to over-sample the minority class observations.

**7.1. Algorithms:**

1. **Logistic Regression:**

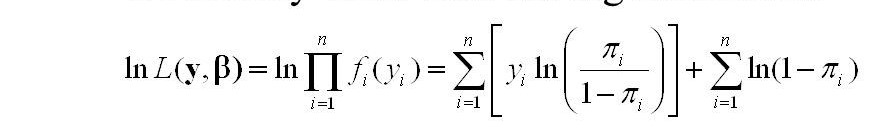
Logistic Regression is actually a classification algorithm that was given the name regression due to the fact that the mathematical formulation is very similar to linear regression.

The function used in Logistic Regression is sigmoid function or the logistic function given by:

f(x)= 1/1+e ^(-x)



The optimization algorithm used is: Maximum Log Likelihood. We mostly take log likelihood in Logistic:

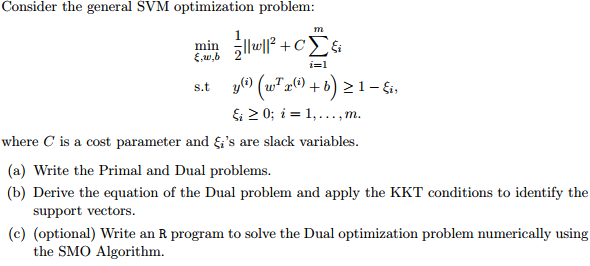


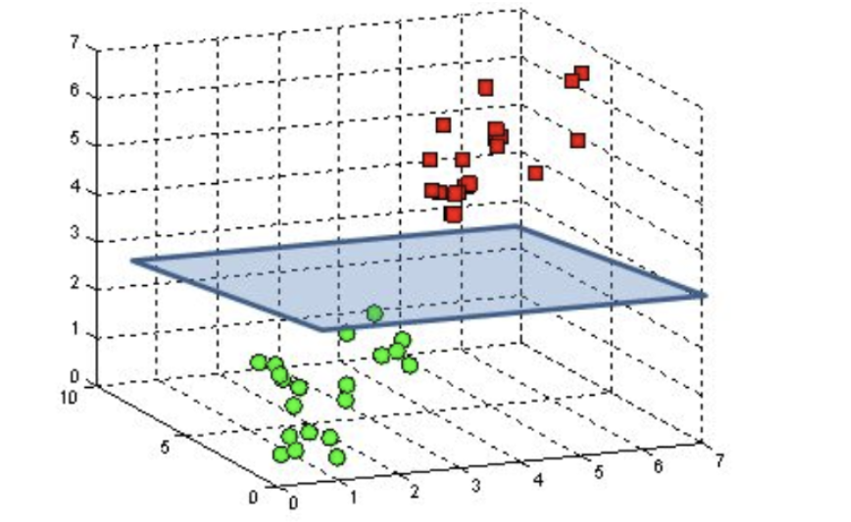
.

1. **Support Vector Machine Classifier:**

SVM is used mostly when the data cannot be linearly separated by logistic regression and the data has noise. This can be done by separating the data with a hyperplane at a higher order dimension.

In SVM we use the optimization algorithm as:





We use hinge loss to deal with the noise when the data isn’t linearly separable.

Kernel functions can be used to map data to higher dimensions when there is inherent non linearity.

1. **Random Forest Classifier:**

Random Forest is a bagging type of Decision Tree Algorithm that creates a number of decision trees from a randomly selected subset of the training set, collects the labels from these subsets and then averages the final prediction depending on the most number of times a label has been predicted out of all.



1. **Decision Tree**:

Decision Tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.



1. **Grid Search CV**

GridSearchCV is the process of performing hyperparameter tuning in order to determine the optimal values for a given model. As mentioned above, the performance of a model significantly depends on the value of hyperparameters. Note that there isno way to know in advance the best values for hyperparameters so ideally, we need to try all possible values to know the optimal values. Doing this manually could take a considerable amount of time and resources and thus we use GridSearchCV to automate the tuning of hyperparameters.

**7.2. Model performance:**

Model can be evaluated by various metrics such as:

1. **Confusion Matrix**-

The confusion matrix is a table that summarizes how successful the classification modelis at predicting examples belonging to various classes. One axis of the confusion matrix is the label that the model predicted, and the other axis is the actual label.

1. **Precision/Recall**-

Precision is the ratio of correct positive predictions to the overall number of positive predictions : TP/TP+FP

Recall is the ratio of correct positive predictions to the overall number of positive examples in the set: TP/FN+TP

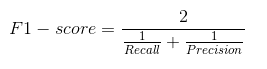
1. **Accuracy**-

Accuracy is given by the number of correctly classified examples divided by the total number

of classified examples. In terms of the confusion matrix, it is given by: TP+TN/TP+TN+FP+FN

1. **F1 score-**

**F1-score is a harmonic mean of Precision and Recall**, and so it gives a combined idea about these two metrics. It is maximum when Precision is equal to Recall.



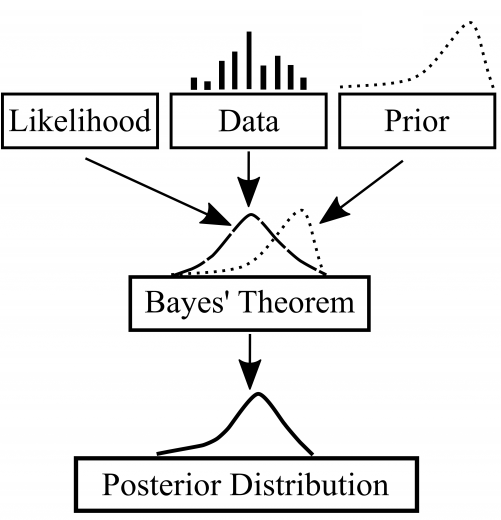
**7.3. Hyper parameter tuning:**

Hyperparameters are sets of information that are used to control the way of learning an algorithm. Their definitions impact parameters of the models, seen as a way of learning, change from the new hyperparameters. This set of values affects performance, stability and interpretation of a model. Each algorithm requires a specific hyperparameters grid that can be adjusted according to the business problem. Hyperparameters alter the way a model learns to trigger this training algorithm after parameters to generate outputs.

We used Grid Search CV, Randomized Search CV and Bayesian Optimization for hyperparameter tuning. This also results in cross validation and in our case we divided the dataset into different folds. The best performance improvement among the three was by Bayesian Optimization.

1. **Grid Search CV-**Grid Search combines a selection of hyperparameters established by the scientist and runs through all of them to evaluate the model’s performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific region of the parameter space and cannot understand which movement or which region of the space is important to optimize the model.
2. **Randomized Search CV-** In Random Search, the hyperparameters are chosen at random within a range of values that it can assume. The advantage of this method is that there is a greater chance of finding regions of the cost minimization space with more suitable hyperparameters, since the choice for each iteration is random. The disadvantage of this method is that the combination of hyperparameters is beyond the scientist’s control

# **Bayesian Optimization-** Bayesian Hyperparameter optimization is a very efficient and interesting way to find good hyperparameters. In this approach, in naive interpretation way is to use a support model to find the best hyperparameters.A hyperparameter optimization process based on a probabilistic model, often Gaussian Process, will be used to find data from data observed in the later distribution of the performance of the given models or set of tested hyperparameters.



As it is a Bayesian process at each iteration, the distribution of the model’s performance in relation to the hyperparameters used is evaluated and a new probability distribution is generated. With this distribution it is possible to make a more appropriate choice of the set of values that we will use so that our algorithm learns in the best possible way.

**8. Conclusion:**

Since our aim was to lower the false-negative value so that patients do not get detected improperly and are demonstrated to be safe, I used the recall score as the evaluation matrix. The patient's health may suffer greatly as a result of this.

Data were resampled because they weren't balanced. High accuracy can be achieved with imbalanced data, however, in these situations, recall, precision, and F1 score must be considered.

Used a KNN-imputer to perform missing value imputation, and processed data to remove outliers. To solve the issue of class imbalance, SMOTE boosting was used to over-sample the minority class observations.

Newer elements like pulse pressure, age bucket, and BMI bucket that helped to explain the separation in the Risk were created using the information from EDA.

Due to the parametric relationship in the data, a logistic regression model was implemented, and it was successful in achieving a Recall of 74.5%. Even though the recall score for SVM was 81.9 %, SVM is not an interpretable model, thus I chose an interpretable model for this situation.

All measures, including Precision, Recall, Accuracy, and F1 score were evaluated for each model.

Based on this analysis,

Logistic regression can identify positive cases with a 74.5% Recall.

Using a decision tree, positive cases may be predicted with a recall of 49%.

With the help of Random Forest, positive cases may be predicted with a 49% Recall.

Using a Support Vector Machine, positive cases can be predicted with an 81.9% Recall.

Using Grid Search CV, positive cases can be predicted with 81.9% Recall.

**References-**

1. MachineLearningMastery
2. GeeksforGeeks
3. Analytics Vidhya