Cardiovascular Risk Prediction

Shrinidhi Choragi
Data Science Trainee, AlmaBetter
Bangalore.

Abstract

Multivariate cardiovascular risk prediction models quantify the risk of cardiovascular disease over a defined period or during a lifetime in apparently healthy persons, patients with established cardiovascular disease, or patients with a medical history that support the risk factors.

However, translating this simple observation into a quantitative probability of disease requires a number of factors. First, reliable data must be available to indicate disease incidence rates. Second, a number of risk markers must be available to assess future disease incidence. Statistical methods must be available to quantify the prospective relation between risk markers and the occurrence of disease. Finally, and importantly, the utility of risk prediction algorithms must be assessed in the context of the clinical environment, including the severity of the disease being predicted.

Machine learning offers a good opportunity to build and improve predictions by exploiting complex interactions between risk factors. It significantly improves the accuracy of cardiovascular risk prediction, increasing the number of patients identified who could benefit from preventive treatment.

Keywords: *risk*, *prediction*, *prevention.*

Problem Statement

The objective is to understand the rationale for using cardiovascular risk prediction methods to make effective and appropriate risk factor treatment decisions in clinical practice. The classification goal is:

- **To predict whether the patient has a 10-year risk of future coronary heart disease (CHD).**
- To understand and exploit the impact of different risk factors.
- To study the vulnerability in patients due to different risk factors/features.
- To be able to interpret and analyze the prediction outcomes of the model built to classify cardiovascular patients.

Introduction

Cardiovascular disease (CVD) remains the most important cause of mortality worldwide. For the prevention of CVD, cardiovascular risk management is advocated in international guidelines.

Healthy lifestyle behaviour should always be promoted at the individual *and* population levels. With this growing plethora of choices in cardiovascular prevention, it can be difficult for healthcare professionals and patients to make the most appropriate treatment decisions for each individual.

Identifying those patients who will benefit most from risk factor treatment is pivotal in the global CVD prevention effort. Risk assessment using risk prediction tools can thus play a highly important part in global CVD prevention efforts in choosing the right treatment, for the right patient. Thus the motivation for the project.

Why should CVD risk prediction be used in clinical practice?

Prevention is better than cure, also in the context of any heart-related disease. The higher the absolute risk, the higher the absolute benefit of risk factor treatment. In line with this thinking, international guidelines state that the level of risk in an individual patient should guide the decision whether or not to treat risk factors, or how intensively to treat them.10-year risk is largely age-driven, and is thus almost invariably low in young persons and high in older persons, severely limiting its use for clinical decision-making in these age groups.

So, what are risk prediction models?

Risk prediction models are the mathematical functions that predict the occurrence of an event of interest based on certain predictors, such as patient demographics, medical history, medication use, behaviour, physical examination, disease characteristics, and heart laboratory values.

To be useful in clinical practice, risk prediction models should be easily accessible including clinically relevant and readily available predictors, and reliably estimate risk in a way that improves treatment decision-making and patient outcomes.

Dataset

The dataset is from an ongoing cardiovascular study on residents of the town of Framingham, Massachusetts. The dataset provides the patients' information. **It includes over 4,000 records and 15 attributes.**

Each attribute is a potential risk factor. There are demographic, behavioral, and medical risk factors.

Data Description

• *Sex*: male or female("M" or "F")

•*Age*: patient's age;(Continuous)

• *is_smoking*: whether or not the patient is a current smoker (YES/ NO)

• *Cigs Per Day*: the number of cigarettes that the person smoked on average in one day

• *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)

• *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)

• *Glucose*: glucose level (Continuous)

• *ten_year_chd*: 10-year risk of coronary heart disease (binary: 1/0 means "Yes"/"No")- Target variable.

As our dependent variable is a discrete categorical variable, this makes it a classification problem.

What is a Classification problem?

In Machine Learning, most classification problems require predicting a categorical output variable called *target*, based on one or more input variables called *features*. The idea is to fit a statistical model that relates a set of features to its respective target variable to use this model to predict the output for future input observations.

This problem is a Binary Classification Problem; a classification predictive modeling problem where all examples belong to one of two classes.

Methodology

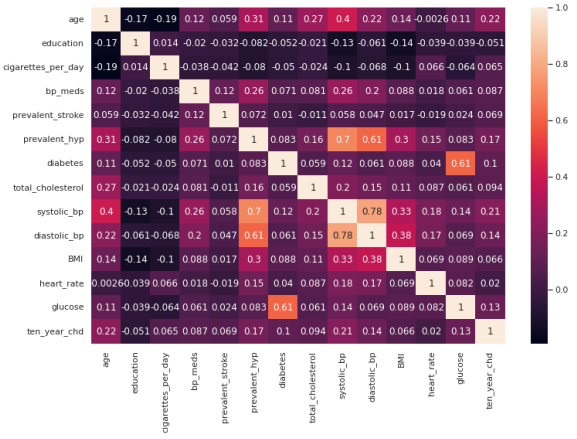
1. Missing Value Analysis

Dropping the missing data isn't a better choice since the size of the dataset is small. Therefore the categorical and numerical missing data are imputed with the mode and median values respectively while avoiding any redundancy in the data.

2. Outlier Analysis

The risk factor outliers may contain important and clinically meaningful information. Therefore deleting/transforming the outliers leads to a loss/modification of potential risk factors. Therefore outliers are left untreated in this case.

3. Correlation Analysis

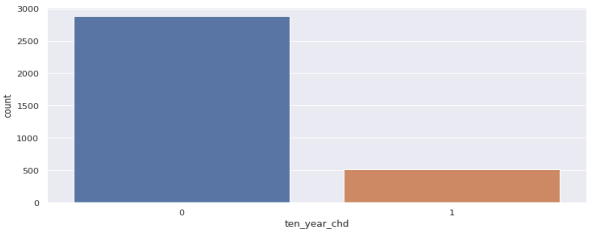


There is no significant correlation between the target variable and features. And independent variables are correlated among themselves.

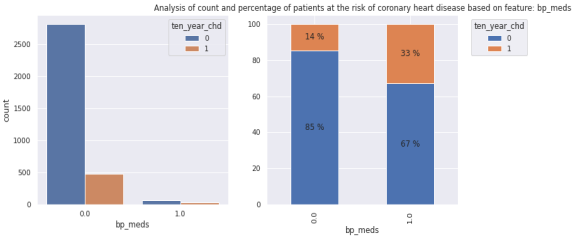
1. High correlation of 'systolic bp' and 'diastolic bp' with prevalent hypertension.
2. Features 'glucose' and 'diabetes' are correlated with a correlation of (0.61).
3. Systolic and diastolic blood pressure are highly correlated(0.78); a new feature could be derived using the two features.

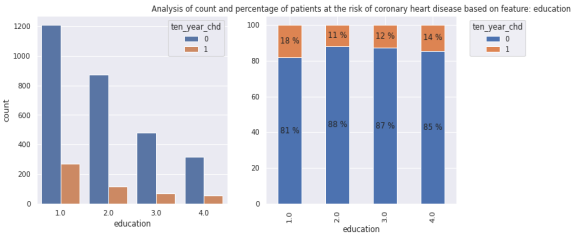
4. Exploratory Data Analysis

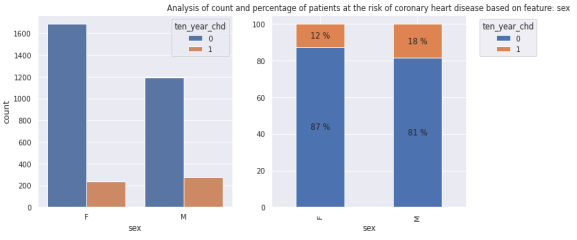
The dependent variable is binary(categoric), implying a classification problem. There is an imbalance in the dataset, that should be taken into consideration during further analysis.

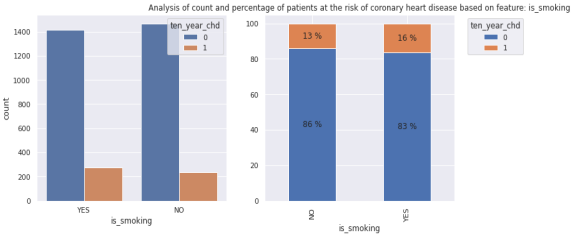


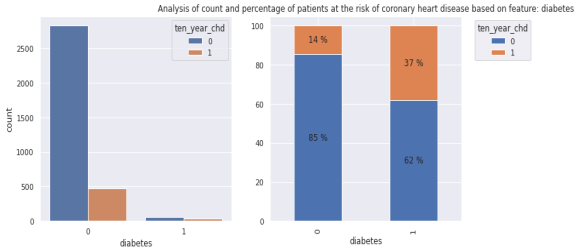
The exploratory data analysis helped in understanding the distribution of predictors, and the relation between the risk factors and the target variable was understood both quantitatively and visually. Vulnerability to disease due to risk factors is depicted using percentage bar plots.

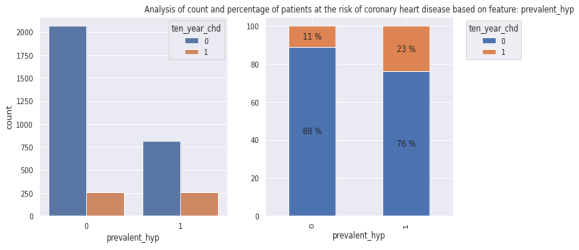


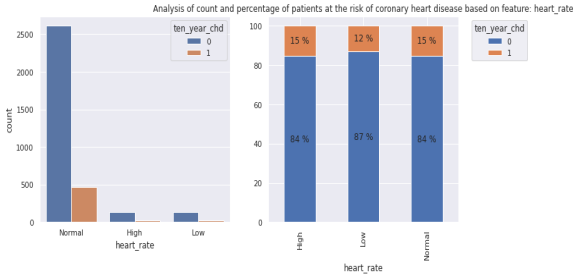


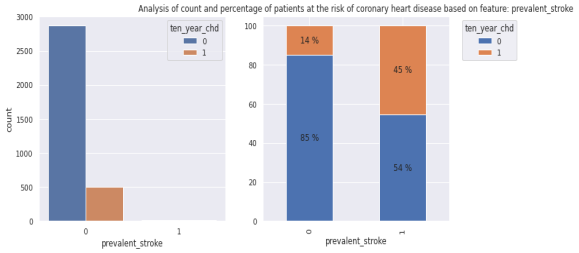




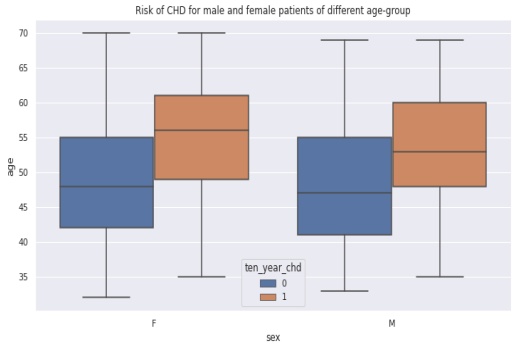








The average age os risk of CHD is higher in female patients.



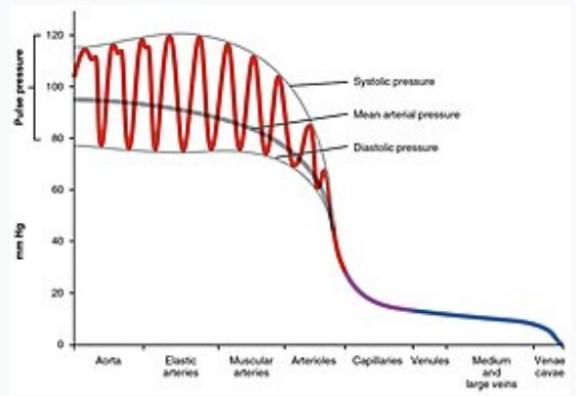
Feature Engineering

1. Encoding categorical columns

One Hot Encoding is used to produce binary integers of 0 and 1 to encode our categorical features. The categorical features namely *sex and is_smoking* are encoded.

2. Feature Imputation

Features with high correlation could hamper the effect of others on the dependent variable. Therefore, when two features have a high correlation, either one of them could be dropped or a new feature can be produced using the two. The latter is implemented in this case.



A new feature *Pulse Pressure* is imputed by deleting the correlated features- *systolic_bp and diastolic_bp.*

"*Pulse pressure is the difference between systolic and diastolic blood pressure.*"

Systolic Blood Pressure - Diastolic Blood Pressure.

It represents the force that the heart generates each time it contracts. Resting blood pressure is normally approximately 120/80 mmHg, which yields a pulse pressure of approximately 40 mmHg.

3. Feature Selection

To check whether the given variables/features are related to the target variable a Null Hypothesis is defined.

Null Hypothesis (H0): Features are independent of the target variable.

Alternate Hypothesis (H1): Features are dependent on the target variable.

If a feature exhibits a 'p_value' smaller than 0.05 for a 95% confidence interval, it shows the dependency on the target variable therefore, we reject the null hypothesis and accept the alternate hypothesis. By this, we

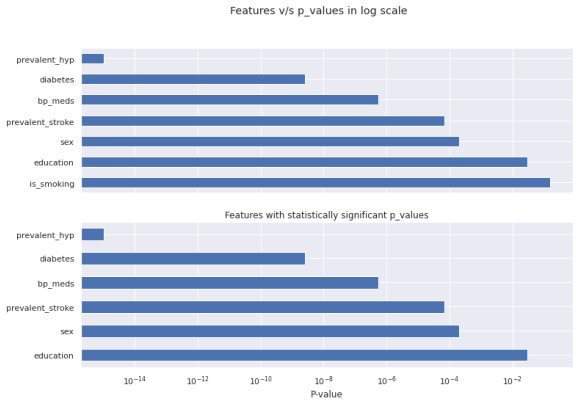
may choose to either drop the column with a high p-value or handle it in some other way.
In feature selection, we aim to select the features which are highly dependent on the response variable. Scores based on statistical tests such as Chi-Square and F-Statistic provide a p-value, that is used to rule out some features.

A **p-value** is a statistical measurement used to validate a hypothesis against observed data and define the feature selection quantitatively. It measures the probability of obtaining the observed results, assuming that the null hypothesis is true. The lower the p-value, the greater the statistical significance of the observed difference.

A p-value less than 0.05 is typically considered to be statistically significant, in which case the null hypothesis should be rejected otherwise the null hypothesis is not rejected.

Two statistical measures used for feature selection are as follows:

Chi-Square Test



A chi-square test is used in statistics to test the independence of two events. Given the data of two variables, we can get observed count O and expected count E. Chi-Square measures how expected count E and observed count O deviate from each other.

The formula for Chi-Square is given by:

$$\chi_c^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

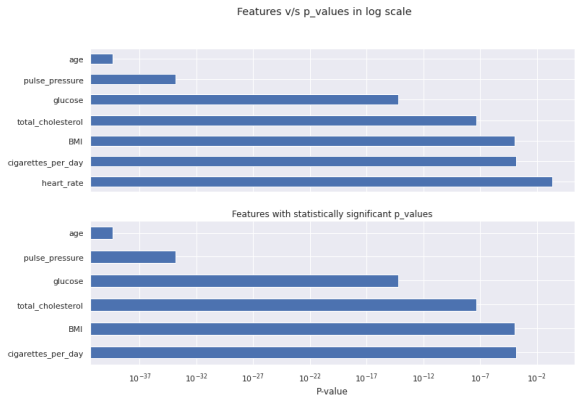
where:

c = degrees of freedom

O = observed value(s)

E = expected value(s)

F -statistic



The ANOVA method is a type of F-statistic referred to here as an ANOVA f-test. Importantly, ANOVA is used when one variable is numeric and one is categorical, such as numerical input variables and a classification target variable (categorical) in a classification task.

F- Value: It is the *ratio of two Chi-distributions* divided by their degrees of Freedom.

$$F = (\chi_1^2 / n1 - 1) / (\chi_2^2 / n2 - 1)$$

Where χ_1, χ_2 are Chi distributions and n1,n2 are its respective degrees of freedom.

Data Preparation and Modeling

1. Skew Transformation

The skewed distributions are converted to a normal distribution using logarithmic and reciprocal transformation.
As a rule of thumb,

- If *skewness*<-1 or *skewness*>1: the distribution is highly skewed.
- If -1<*skewness*<-0.5 or 0.5<*skewness*<1: the distribution is moderately skewed.
- If -0.5<*skewness*<0.5: the distribution is approximately symmetric.

2. Data Splitting

The dataset is split into train and test data in the ratio of 70:30 resp. The model is fit on the training data and tested on the newest data. Sklearn's *train_test_split* is used.

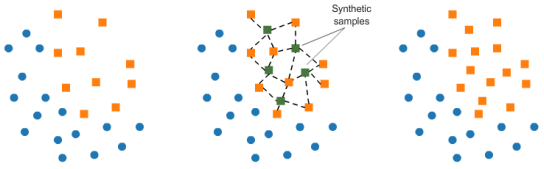
3. Oversampling:

Handling Data Imbalance

Due to the large difference in the number of observations of both the types of the dependent variable, it becomes an Imbalanced Classification, where predictive models are developed on classification datasets that have a severe class imbalance.

One approach to addressing imbalanced datasets is to oversample the minority class.

The simplest approach involves duplicating examples in the minority class, although these examples don't add any new information to the model. This is a type of data augmentation for the minority class and is referred to as the Synthetic Minority Oversampling Technique, or SMOTE for short.



SMOTE: Synthetic Minority Oversampling Technique

- Choose a minority class as the input vector and find its k nearest neighbors.
- Choose one of these neighbors and place a synthetic point anywhere on the line joining the point under consideration and its chosen neighbor
- Repeat the steps until the data is balanced.

Note: *Typically undersampling/oversampling techniques will be done on train split only, this is the correct approach. In order to avoid using synthetic data for testing purposes*.
***Oversampling both train and test sets will only overestimate the results*.**

Feature Scaling

The objective of feature scaling is to enclose all features in a common boundary without losing information.

The three things that define the features are units, magnitude, and range. Most of the time, the dataset varies in these aspects. But since, since most machine learning algorithms use the distance between two data points in their computations, this is a problem. To suppress this effect, scaling is done.

Tree-based models are not distance-based models and can handle varying ranges of features. Hence, Scaling is not required while modeling trees.

In the given classification problem, Standard Scaler is used, which is defined as

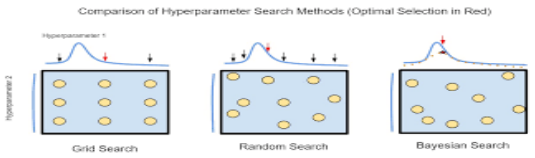
Standard Scaler	$\frac{x_i - \text{mean}(\boldsymbol{x})}{\text{stdev}(\boldsymbol{x})}$

Hyperparameter 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 the performance, stability, and interpretation of a model. Each algorithm requires a specific hyperparameter 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 uses cross-validation.



Grid Search CV

This approach is effectively a brute force strategy, simply creating and testing a model for each hyper-parameter configuration — the approach benefits from the exhaustive search behavior. 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. In addition, there is a certainty that each combination has been compared.

However, it should be clear that using a comprehensive search for hyper-parameters is expensive. Since these searches handle combinatorial complexity, the number of models that have to be tested increases dramatically. Because of this computational burden, grid search is most effective when the optimal selection of some hyper-parameters is already known.

Randomized Search CV

In contrast to an exhaustive search of each hyper-parameter configuration, a random search is only run for a set number of samples. Each sample selects hyper-parameters at random, either from a list of possible parameters or the distribution of parameters. While this approach appears to be less optimal compared to grid search, it performs better given the same amount of computation. 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

In this approach, the naive interpretation way is to use a support model to find the best hyperparameters. A hyperparameter optimization process based on a probabilistic model, often the 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.

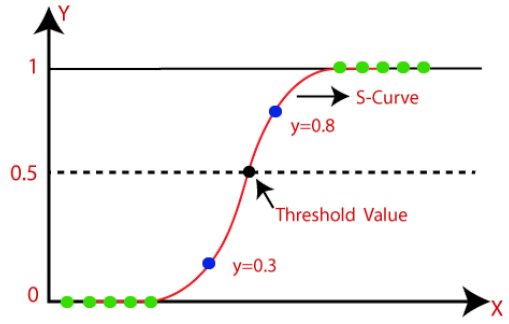
Model fitting

The following models have been studied and implemented on the given dataset:

- Logistic Regression
- Decision Trees
- Random Forest
- Support Vector Machine
- K-Nearest Neighbor

Logistic Regression

Two of the important parts of logistic regression are *Hypothesis and Sigmoid Curve*. With the help of this hypothesis, the likelihood of the event is derived. The data generated from this hypothesis can fit into the log function that creates an S-shaped curve known as "sigmoid". Using this log function, the category of the class is predicted.
Sigmoid/Logistic function represented as:



Logistic Function (Sigmoid Function) is given by **f(x)= 1/1+e ^(-x)**

- The sigmoid function is a mathematical function used to map the predicted values to probabilities.
- It maps any real value into another value within a range of 0 and 1. Values above/below the threshold are mapped to 1/0 respectively.

Assumptions for Logistic Regression:

- The dependent variable must be categorical in nature.
- The independent variable should not have multi-collinearity.

Decision Trees

It is a classification machine learning algorithm that involves dividing a dataset into segments based on certain feature variables from the dataset. Given data of attributes together with its classes, a decision tree produces a sequence of rules that can be used to classify the data.

A decision tree is composed of the following elements: A root, many nodes, branches, and leaves. The root node does the partition based on the attribute value of the class; the internal node takes an attribute for further classification; branches make a decision rule to split the nodes into leaf nodes; lastly, the leaf nodes give us the final outcome.

There are two widely used measures to test the purity of the split (a segment of the dataset is pure if it has data points of only one class).

The first one is the Gini index which measures total variance across the N classes.

$$G = -\sum_{k=1}^{k} p_{mk}\,(1 - p_{mk})$$

Another measure is cross-entropy, defined by

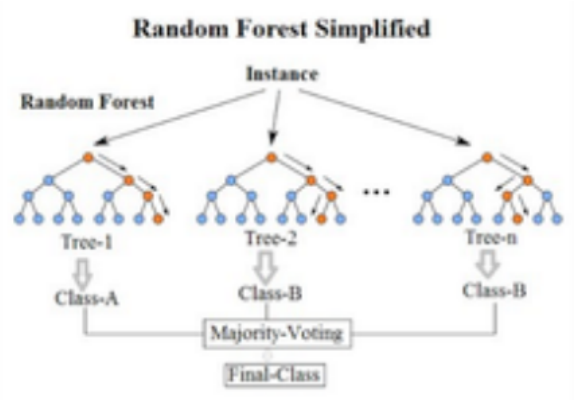
$$D = -\sum_{k=1}^{k} p_{mk} \log\,(p_{mk})$$

In both equations, 'pmk' represents the proportion of training variables in the mth segment that belongs to the kth class.

The dataset is split into segments based on that feature, giving rise to the minimum value of entropy or Gini index.

Random Forest

A random forest involves processing many decision trees where each tree predicts a value for the probability of target variables. Then the final class is decided by majority voting of the class as predicted by all the trees in the forest.



Each tree is evaluated as follows:

1. First samples of the dataset are created by selecting data points with replacements.

2. Next, only a subset of the available input variables is used.
3. Each tree is allowed to grow to the most considerable length possible, and no pruning is involved.

Random Forest is efficient with large datasets but requires high computational cost and high memory.

Support Vector Machine

It can be represented with training data as points in space separated into categories by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed linearly separable data, and classifier is used called Linear SVM classifier otherwise it is a Non-linear SVM classifier.

This algorithm utilizes support vector classifiers with enlarged feature variable space using special functions called *kernels* that make it suitable for evaluating a non-linear decision boundary.

The mathematical function that it uses for evaluating the boundaries is given by:

$$f(x) = \beta_0 - \sum_{i \in s} \alpha_i K(x, x_i)$$

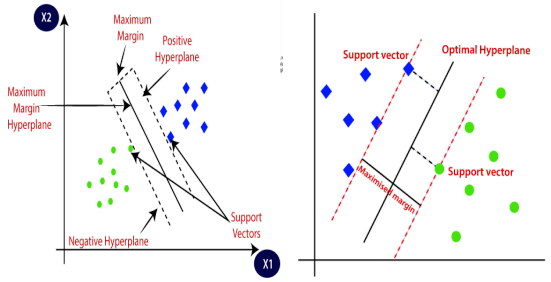
Where,

K represents the kernel function,

α and ***β*** are training parameters.

The dimensions of the hyperplane depend on the features present in the dataset. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme closest point of the lines are called support vectors, and hence algorithm is termed a Support Vector Machine.

The distance between the vectors and the hyperplane is called a margin. And the goal of SVM is to maximize this margin. The hyperplane with the maximum margin is called the optimal hyperplane.



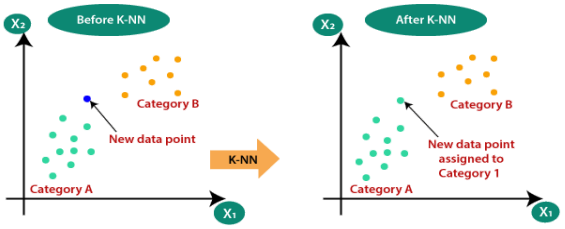
Advantages:

Effective in high dimensional spaces and uses a subset of training points in the decision function so it is also memory efficient.

Disadvantages:

The algorithm does not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation. SVM is not suitable for large data sets and when target classes are overlapping. So, it needs to be handled.

k-Nearest Neighbors



The K-NN working can be explained on the basis of the below algorithm: To classify a new data item,

1. Select the number K of the neighbors and calculate the *Euclidean* distance of K number of neighbors.

2. Take the K nearest neighbors as per the calculated Euclidean distance.
3. Among these K neighbors, count the number of the data points in each category and assign the new data points to that category for which the number of the neighbor is maximum.

K-NN is a non-parametric algorithm, which means it does not make any assumptions on underlying data.

It requires to be provided with a value of K initially also, the computation cost is high because of calculating the distance between the data points for all the training samples.

Evaluation Metrics and Visualizations

Choosing the right metric for a classification model is very crucial. For example, for our dataset, we can consider that achieving a high recall is more important than getting a high precision – we would like to detect as many heart patients as possible. For some other models, like classifying whether a bank customer is a loan defaulter or not, it is desirable to have a high precision since the bank wouldn't want to lose customers who were denied a loan based on the model's prediction that they would be defaulters.

In other cases, there are also a lot of situations where both precision and recall are equally important. For example, for our model, if the doctor informs us that the patients who were incorrectly classified as suffering from heart disease are equally important since they could be indicative of some other ailment, then we would aim for not only a high recall but a high precision as well.

Therefore for this project, ***the aim is to maximize the recall while keeping a reasonable ability to distinguish between the classes indicated by the AUC of the ROC curve.***

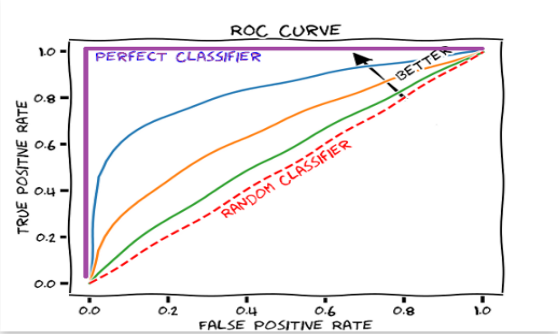
Confusion Matrix/ Error Matrix:

		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

The confusion matrix provides us with a matrix/table as output and describes the performance of the model. The matrix consists of predictions resulting in a summarized form, which has a total number of correct predictions and incorrect predictions.

AUC-ROC curve:

ROC curve stands for Receiver Operating Characteristics Curve and AUC stands for Area Under the Curve. It is a graph that shows the performance of the classification model at different thresholds.



A ROC curve is a two-dimensional graph to depict trade-offs between benefits (true positives) and costs (false positives). It displays a relation between sensitivity and specificity for a given classifier.

F1-Score:

F1-score is given as:

(2 x Precision x Recall) / (Precision + Recall)

F1-Score is the weighted average of Precision and Recall used in all types of classification algorithms. Therefore, this score takes both false positives and false negatives into account. F1-Score is usually more useful

than accuracy, especially if you have an uneven class distribution.

Where,

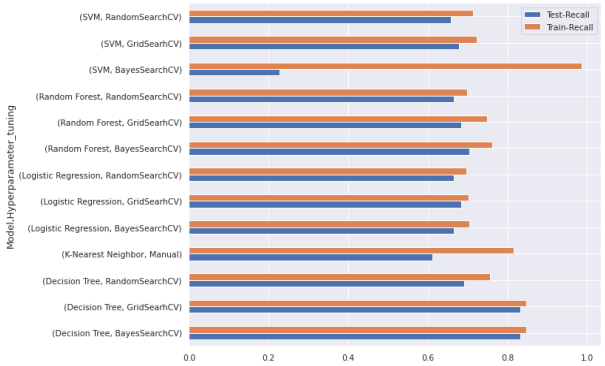
- Precision: When a positive value is predicted, how often is the prediction correct?
- Recall: When the actual value is positive, how often is the prediction correct?

Results

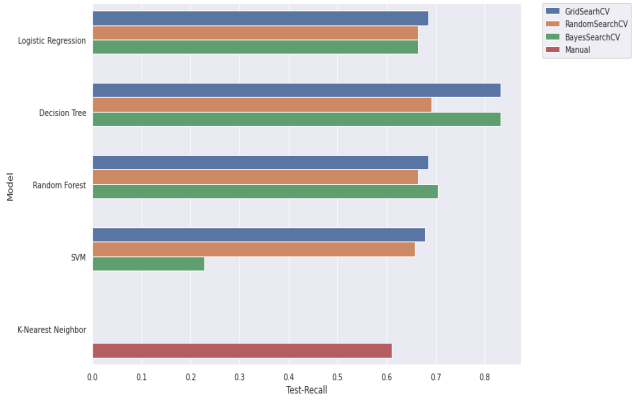
Recall-Scores tabulated for different models.

Model	Hyperparameter_tuning	Train-Recall	Test-Recall
Logistic Regression	GridSearhCV	0.7016	0.6846
Logistic Regression	RandomSearchCV	0.6977	0.6644
Logistic Regression	BayesSearchCV	0.7041	0.6644
Decision Tree	GridSearhCV	0.8468	0.8322
Decision Tree	RandomSearchCV	0.7558	0.6913
Decision Tree	BayesSearchCV	0.8468	0.8322
Random Forest	GridSearhCV	0.7479	0.6846
Random Forest	RandomSearchCV	0.6987	0.6644
Random Forest	BayesSearchCV	0.7618	0.7047
SVM	GridSearhCV	0.7225	0.6779
SVM	RandomSearchCV	0.7131	0.6577
SVM	BayesSearchCV	0.9861	0.2282
K-Nearest Neighbor	Manual	0.8165	0.6107

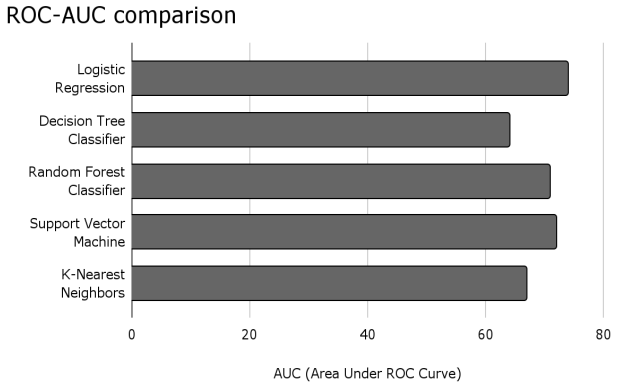
Recall scores are plotted in a barplot for models with different hyperparameter tuning.



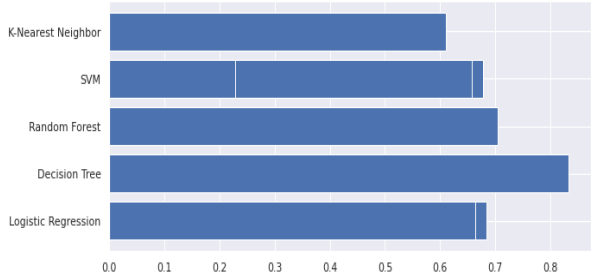
Test Recall Scores



ROC-AUC scores



Test-Recall scores for the optimally tuned models.



Summary

Considering the reliable data available, that indicates disease incidence rates/risk factors, exploratory and predictive analysis is done.

Starting with the data pre-processing that includes missing values analysis, wherein the missing values in data are imputed with appropriate aggregate values. The outliers are left unhandled considering the possibility of them being the potential risk factors. The correlation between the dependent variable and the features is checked. Also, the dependency between the risk factors is exploited to handle the multicollinearity by imputing a new feature- Pulse Pressure to replace the dependent ones- Systolic and Diastolic Blood pressure.

The exploratory data analysis explains the type of classification and imbalance in class. It also shows the vulnerability of patients to disease based on the risk factors. The analysis helped understand the contribution of risk factors in predicting the risk.

A null hypothesis is proposed and important features are identified and selected based on statistical methods- Chi-Square and ANOVA F-test score p_values that quantify the prospective relation between features and the target variable. Feature importance study shows that age is one of the most important factors in predicting the risk of the disease. Followed by pulse pressure, glucose, and soon. Further oversampling is done in order to balance the data using one of the data augmentation techniques called SMOTE. Skew transformation, encoding, and scaling are considered to prepare the data for modeling.

Various classification algorithms like the following are experimented with to bring in the prediction along with various hyperparameter tuning techniques namely *GridSearch*CV, *RandomSearch*CV, *and BayesSearch*CV to fine-tune the parameters.

- Logistic Regression
- Decision Tree
- Random Forest
- Support Vector Machine
- K-Nearest Neighbors

Finally, and importantly, the utility of risk prediction algorithms is assessed in the context of the clinical environment, in order to do justice to the imbalance and the type of problem at hand. Therefore considered to maximize the **recall score** while having a reasonable ability to distinguish between the classes as indicated by the **ROC_AUC score**.

The conclusion is that the Random Forest model with Bayesian optimization can be considered the best among the ones that are experimented with. It predicts a pretty high recall score**(0.7046)** while having a reasonable capacity to distinguish between the classes. (**ROC_AUC= 0.71).**

Since it seems right to apply the oversampling techniques only on train split and go for prediction rather than including the synthesized samples in test data to overestimate the results. However future improvements may include attempting to collect more data on minority class that helps the analysis to be improved further.

Conclusion

After the risk model development, the reliability of the predictions needs to be tested, both in the dataset used for model development and in other populations (external validation). **Although the recall and roc_auc for discrimination are most often presented, it could be argued that model calibration ('goodness of fit') is a more clinically relevant measure of model performance, as clinician and patient want to know if the predicted risk resembles the actual risk**. If baseline event risk (ie, which percentage of people in a population will

experience a CVD event) is very different in a population compared with the study population in which a model was derived, the predicted risks will systematically underestimate or overestimate the actual individual risks in that population. In that case, it is necessary to recalibrate the model to better reflect the baseline risk in the external population. This may, for example, be necessary for different countries (geographical recalibration), or when a model is older to better reflect a contemporary population.

Predictions of CVD risk thus support informed treatment decisions for preventive treatment and aid in choosing the right treatments and right treatment goals for the right patient.

Therefore, its believed that this project may support the process of shared decision-making and may increase patient motivation to adhere to lifestyle changes, and consider treatment and medication use.

References

1. Pulse Pressure
2. Sampling
3. Imbalanced Classifications
4. Cardiovascular Risk Prediction In India
5. Feature Selection- statistical methods
6. P-value
7. SMOTE
8. ANOVA- feature selection
9. Analytics Vidhya Articles