Cardiovascular Risk Prediction

Ashik Kumar

Project Description

Heart disease is the major cause of morbidity and mortality globally: it accounts for more deaths annually than any other cause. According to the WHO, an estimated 17.9 million people died from heart disease in 2016, representing 31% of all global deaths. Over three quarters of these deaths took place in low- and middle-income countries.

Of all heart diseases, coronary heart disease (aka heart attack) is by far the most common and the most fatal. In the United States, for example, it is estimated that someone has a heart attack every 40 seconds and about 805,000 Americans have a heart attack every year (CDC 2019).

Doctors and scientists alike have turned to machine learning (ML) techniques to develop screening tools and this is because of their superiority in pattern recognition and classification as compared to other traditional statistical approaches.

In this project, We will be giving you a walk through on the development of a screening tool for predicting whether a patient has a 10-year risk of developing coronary heart disease(CHD) using different Machine Learning techniques.

Evaluation-metric

The most important evaluation metric that we go with, to address this problem statement, is the recall. Since we want to decrease the number of false negatives and predict all the true cases for 10-year risk of having CHD correctly, the focus of our ML models is to improve the recall.

Data Summary

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

Data Description

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, yet are considered continuous because of a large number of possible values.)
- Glucose: glucose level (Continuous)

Predict variable (desired target)

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

Project Workflow

1. Splitting to Train and Test sets to avoid Data bleeding.

2. Simultaneously Data Cleaning of Train and Test sets

3. EDA on features

4. Feature cleaning and engineering

5. Solving Class Imbalanced problem

6. Base Model and Candidate Models

7. Hypertuning of parameters

8. Final Predictions

Train-Test split

We split the data in 80-20 ratio before doing any sort of data cleaning or EDA so as to avoid data beeliding. This ensures that our test data is unseen throughout our training process.

Data cleaning

The null values in the categorical values are imputed using a simple imputer which replaces the null values with the value that is most frequent in the particular feature. While the numerical features are imputed with a KNN imputer that imputes the null values with values that are close to the values of K nearest neighbors of that particular sample. Imputation like these ensure that none of the data is significantly lost and data is safely preserved in the data set.

EDA

The exploratory data analysis that we performed on our train dataset helped us to realize how different features in our dataset influence the target variable. We used a chi-square test to find whether a certain categorical feature was dependent on our binary target variable. We also used a one-way ANOVA test to find out if the distribution of our continuous variables for different classes are similar. This helped us to decide whether certain features need to be included in our final model or not.

Feature Selection and engineering

- From the above EDA we try to establish some patterns which influence the cause of heart disease. We have tried to engineer some new features based on existing features by bucketing some of the continuous variables. We have created age buckets of population e.g 18-25 -> **20s**, 25-40 -> **Mid30s** etc. In this way we might be able to target a particular age group which has a high risk of Coronary Heart disease.

- We also tried to reduce the multicollinearity from the dataset by removing features that are highly correlated, such as diabetes and glucose level, smoking and number of cigarettes per day, hypertension and systolic blood pressure.

Class Imbalanced issue

In this problem we have a dataset of patients where we have to find out whether the given features or symptom a person has he/she has a Cardiovascular disease in future.

But here’s the catch… the risk rate is relatively rare, only 15% of the people have this disease.

The Metric Trap

One of the major issues when dealing with unbalanced datasets relates to the metrics used to evaluate our model. Using simpler metrics like accuracy score can be misleading. In a dataset with highly unbalanced classes, the classifier will always “predict” the most common class without performing any analysis of the features and it will have a high accuracy rate, obviously not the correct one.

Random Over-Sampling

Oversampling can be defined as adding more copies to the minority class. Oversampling can be a good choice when you don’t have a ton of data to work with.

A con to consider when undersampling is that it can cause overfitting and poor generalization to your test set.

Model Training and Evaluation

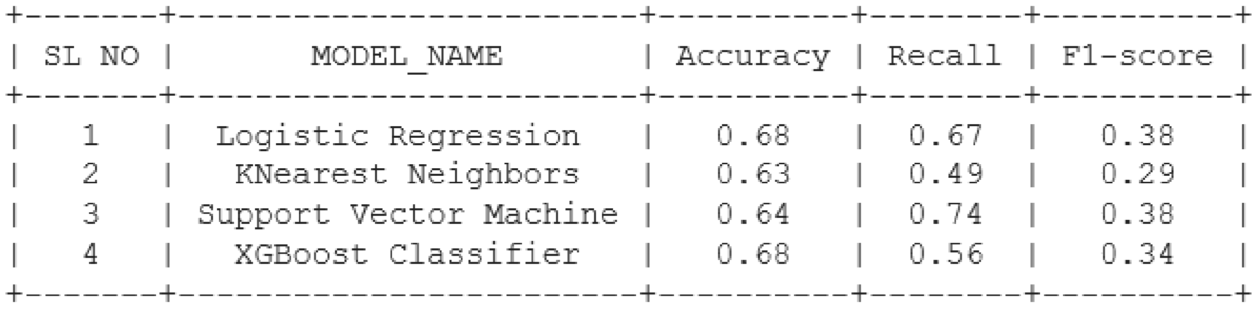
The various candidate models that are used to train our dataset are :

1. Logistic Regression
2. K-Nearest Neighbors
3. Support Vector Machine
4. XGBoost classifier

After training each model and tuning their hyper-parameters using grid search, we evaluated and compared their performance using the following metrics:

- **The accuracy score:** which is the ratio of the number of correct predictions to the total number of input samples. It measures the tendency of an algorithm to classify data correctly.
- **The F1 Score:** Which is defined as the weighted harmonic mean of the test’s precision and recall. By using both precision and recall it gives a more realistic measure of a test’s performance. (Precision, also called the positive predictive value, is the proportion of positive results that truly are positive. Recall, also called sensitivity, is the ability of a test to correctly identify positive results to get the true positive rate).
- **The Recall:** Which provides an aggregate measure of performance across all possible classification thresholds. It gives the probability that the model ranks a random positive example more highly than a random negative example.

Results:



```
+-------+------------------------+----------+---------+----------+
| SL NO |       MODEL_NAME       | Accuracy | Recall  | F1-score |
+-------+------------------------+----------+---------+----------+
|   1   |  Logistic Regression   |   0.68   |  0.67   |   0.38   |
|   2   |   KNearest Neighbors   |   0.63   |  0.49   |   0.29   |
|   3   | Support Vector Machine |   0.64   |  0.74   |   0.38   |
|   4   |   XGBoost Classifier   |   0.68   |  0.56   |   0.34   |
+-------+------------------------+----------+---------+----------+
```

Conclusion

We have used Logistic Regression, KNN, SVC and XGBoost for modelling. Based on our observations, the Support vector classifier seems to have performed better with a recall of 74%. Based on the recall metrics, the model performance is really good, which was our objective from the very beginning i.e. we wanted to correctly predict all the positive cases of high risk CHD.

However, we sometimes also don’t want to flag somebody with no risk of CHD as positive, which might eventually increase the operational cost. We need to ensure that our precision is not too low as well.Thats where our best model still lags. The current precision of the SVC model is around ~0.26. Further work needs to be done that might possibly improve the precision of our model on minority class as well.