Library Progress International *Print version ISSN 0970 1052*
Vol.44 No. 3, Jul-Dec 2024: P. 15018-15026 *Online version ISSN 2320 317X*

Original Article *Available online at www.bpasjournals.com*

Advanced Light Gbm Model Performance Analysis And Comparison For Coronary Heart Disease Prediction

L. Chandra Sekhar Reddy[1]*, Teegala Sandhya[2]

[1]Computer Science and Engineering (Data Science), CMR College of Engineering and Technology, Hyderabad, India.
chandunani@cmrcet.org
[2]M·Tech Scholar, Dept. of CSE, CMR College of Engineering and Technology, Hyderabad, India.
sandhyateegala17@gmail.com

How to cite this article: L. Chandra Sekhar Reddy, Teegala Sandhya (2024) Advanced Light Gbm Model Performance Analysis And Comparison For Coronary Heart Disease Prediction. *Library Progress International,* 44(3), 15018-15026.

Abstract

Coronary heart disease (CHD) is a serious cardiovascular disease that offers a huge wellbeing risk and, unfortunately, has no conclusive arrangement. Detecting coronary artery disease appropriately and right off the bat is basic for giving powerful treatment to patients. Early ID empowers early medicines and worked on understanding results. The recommended "HY_OptGBM" model depends on a better LightGBM classifier to foresee CHD. LightGBM is major areas of strength for a supporting structure that succeeds in prescient demonstrating productivity and precision. The LightGBM classifier is tuned by adjusting its hyperparameters and improving the misfortune capability. This streamlining strategy works on the model's preparation, expanding its precision and productivity. The model's presentation is evaluated utilizing information from the Framingham Heart Institute on coronary heart disease. Utilizing this information, the model succeeds at anticipating CHD, taking into account early ID and maybe prompting lower treatment costs by treating the ailment at a beginning phase. It likewise gives a Voting Classifier (RF + AdaBoost) with an astonishing close to 100% accuracy, which works on the distinguishing proof of CHD. This gathering model, which consolidates Random Forest and AdaBoost, is powerful in perceiving CHD-related designs. To accomplish pragmatic use, an easy to understand Cup system with SQLite network is utilized, improving on the register and signin processes for client testing. This worked on interface further develops openness, making ML moves toward more viable and easy to understand for the numerous partners engaged with CHD finding.

Index terms - coronary heart disease, hyperparameter optimization, Light, loss function, machine learning, OPTUNA.

INTRODUCTION

Atherosclerotic plaques in the coronary arteries reduce blood stream to the heart muscle, causing CHD, a typical cardiovascular disease. Side effects incorporate chest distress, angina, windedness, palpitations, and cardiovascular breakdown. CHD can cause a cardiovascular failure, which can forever harm the heart muscle and lower personal satisfaction. Perceiving and overseeing CHD by clinical mediation and way of life changes is vital [1].

Early CHD distinguishing proof increments fix rates and brings down treatment costs. Because of advances in ML calculations and lower information capacity costs, a few ML calculations and information mining advances have been generally applied in medication [2], [3], [4], [5], [6]. Illness determination, beneficial diagnostics, drug mining, and biomedicine require information mining advancements. We can separate inert ailment data from tremendous measures of unstructured clinical information, develop illness forecast models, and evaluate results utilizing information mining advances.

Medical services suppliers experience a few obstacles in giving superior grade, conservative therapy. Emergency clinics give quality medical services that expects doctors to have careful information and make precise patient analyses to forestall squandering cash. Information mining innovation is compelling and fundamental in clinical situations. The ideal hyperparameters [7], [8] for any arrangement technique extraordinarily influence execution. Picking the best hyperparameters further develops characterization calculation exactness. This study utilized OPTUNA [9] to upgrade LightGBM model hyperparameters. In this manner, our review chosen the best hyperparameters among the open ones. Random and framework searches can improve hyperparameters. Another methodology is OPTUNA hyperparametric search. Since LightGBM execution relies upon how much hyperparameters, standard Random and grid search techniques don't gain from past advancements, fooling around and being inefficient. OPTUNA gains from earlier enhancements and changes hyperparameters on a case by case basis. OPTUNA was picked for hyperparameter enhancement in this work.

The misfortune capability influences model exactness [10]. This work proposes a zeroed in misfortune capability in light of cross-entropy misfortune, including class weight α and test trouble weight changing element γ. This study tended to lopsided positive and negative example extents. Likewise, the center loss capability can help model execution. This study utilized the engaged loss capability to further develop the LightGBM [11] model's default misfortune capability to anticipate CHD.

1. LITERATURE SURVEY

Overweight and stoutness are connected to standard and modern CVD risk factors, which increment the gamble of cardiovascular disease (CVD) and CHD [1]. Cardiovascular disease is additionally connected to stoutness. Focal stoutness and metabolic condition are firmly connected to CVD, particularly CHD. There is solid epidemiologic proof connecting overweight and stoutness to CHD [2], [3], [4], [5], [6]. After death and coronary vein imaging examinations are less persuading. Late examination suggest a corpulent conundrum in CHD mortality. Actual activity and cardiorespiratory wellness decrease weight's CVD chances. There is little information on what deliberate weight reduction means for CVD results in overweight and hefty individuals.

Enormous assets are being utilized to apply software engineering and measurements to clinical difficulties in machine learning (ML). Advocates of ML say it can deal with huge, muddled, and dissimilar information found in medication and will advance biomedical examination, redid therapy, and PC supported diagnostics [12,13]. ML thoughts are unfamiliar to numerous clinical experts, and its exploration potential is underutilized. In this article [2], we cover ML hypothesis, clinical ML calculations, their disadvantages, and the fate of ML in medication.

AI is most normally utilized in drug treatment to match patients to the best medication or blend of medicines, expect drug-target or medication drug communications, and advance treatment regimens. Some ongoing artificial intelligence approaches for pharmacological treatment and organization are evaluated [3]. Joining patient information like hereditary qualities or proteomics with pharmacological information like compound synthetic attributes to assess treatment adequacy is normal in understanding medication choice. Comparability measures are utilized to foresee drug associations by assuming that prescriptions with comparable designs or targets would act much the same way or connect. Numerical models assess pharmacokinetic and pharmacodynamic information to enhance medicine portion. The newly built areas of strength for and for each occupation are examined and evaluated here [12].

The dataset and preparing techniques incredibly influence ML model execution. The right preparation calculation can change a model's story. A few calculations succeed in some datasets yet battle in others. Execution can likewise be expanded by altering calculation hyperparameters that direct preparation. This work [7] utilizes Grey Wolf Optimization (GWO) and Genetic algorithm (GA) metaheuristics to streamline ML calculation hyperparameters. Additionally, 11 calculations like Averaged Perceptron, FastTree, FastForest, LGBM, and Limited Memory. Broyden Fletcher Goldfarb Shanno algorithm Maximum Entropy (LbfgsMxEnt), Linear Support Vector Machine (LinearSVM), and a Deep Neural Network (DNN) with four models are utilized on 11 natural, biomedical, and nature datasets about sub-atomic communications, disease, clinical determination, conduct forecasts, RGB pictures of human skin, and X-beams of Covid19 and cardiomegaly patients. We found that all preliminaries upgraded preparing stage execution. Likewise, GWO performs better with 2.6E-5 p-esteem. Most examinations in this study show that metaheuristic approaches beat Thorough Lattice Search and combine speedier. The proposed strategy takes a dataset and offers the best-investigated calculation with related contentions. Along these lines, datasets with unsure dispersion, ML calculations with complex way of behaving, and shoppers new to logical measurements and information science techniques can use it.

ML might be the best instrument for high-throughput sequencing genomic information examination because of its expectation limit. The complicated system of tweaking hyperparameters enormously blocks ML's utilization in creature and plant reproducing projects. To streamline genomic expectation utilizing ML, we integrated tree-structured Parzen estimator (TPE), an independent tuning hyperparameters approach. TPE improved KRR and SVR hyperparameters in this work [8]. To survey TPE execution, we analyzed KRR-TPE and SVR-TPE expectation accuracy to genomic best linear unbiased prediction (GBLUP) and KRR-RS, KRR-Grid, SVR-RS, and SVR-Grid, which tuned KRR and SVR hyperparameters utilizing random search (RS) and grid search (Gird) in reproduction and genuine datasets [47]. KRR-TPE anticipated all populaces well and was generally advantageous. For Chinese Simmental hamburger steers and Loblolly pine populaces, KRR-TPE exhibited a 8.73% and 6.08% normal increment above GBLUP in expectation exactness. Our work will help GP ML and breeding development.

2. METHODOLOGY

i) Proposed Work:

The proposed framework looks to further develop a LightGBM model for predicting coronary heart disease, evaluate its presentation, use gathering draws near, empower client input for expectation, and extend the framework with an easy to understand frontend and confirmation highlights. Streamlining and troupe approaches improve exactness, which is basic for reliable coronary illness expectation. Fine-tuning LightGBM brings about a viable expectation model with diminished boundaries and misfortune capabilities. The framework's adaptability makes it relevant to an extensive variety of medical services regions, exhibiting versatility and more prominent importance [11,26]. It likewise gives a Voting Classifier (RF

+ AdaBoost) with an astounding close to 100% accuracy, which works on the ID of Coronary Heart Disease(CHD). This ensemble model, which consolidates Random Forest and AdaBoost, is vigorous in perceiving CHD-related designs. To accomplish down to earth use, an easy to understand Jar structure with SQLite network is utilized, working on the register and signin processes for client testing. This worked on interface further develops availability, making ML moves toward more pragmatic and easy to use for all partners participated in CHD determination [2], [3], [4], [5], and [6].

ii) System Architecture:

While using ML models, the easier the engineering, the better, especially for enormous scope preparing and datasets. Each of this makes OPTUNA an extraordinary hyperparametric streamlining structure. The design of the superior LightGBM model is displayed in Figure 1. In Figure 1, every specialist plays out a case of the objective capability during the pursuit.

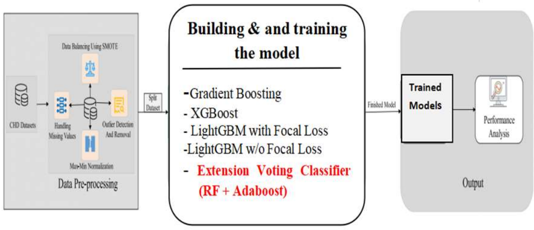
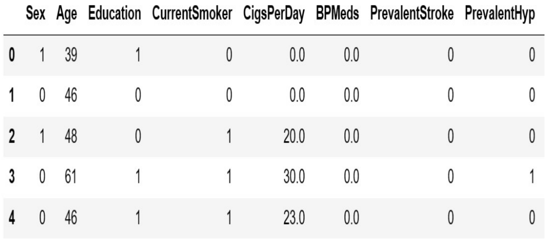


Fig 1: Proposed architecture

iii) Dataset collection:

The Framingham Heart Disease dataset is stacked and contemplated to more readily grasp its design, properties, and content. The Framingham Heart Study (FHS) intends to reveal normal factors or highlights that lead to cardiovascular illness. In 1948, a unique accomplice of 5,209 people matured 30 to 62 were enlisted from Framingham, Massachusetts. A Posterity Companion was laid out in 1971, an Omni Partner in 1994, a Third Era Companion in 2002, Another Posterity Mate Companion in 2004, and a Second Era Omni Companion in 2003. The dataset's center review is centered around cardiovascular and cerebrovascular diseases. The information comprise of organic examples, sub-atomic hereditary information, phenotypic information, tests, photos, member vascular capability information, physiological information, segment information, and ECG data.It is a joint effort between the National Heart, Lung, and Blood Institute and Boston University.



	Sex	Age	Education	CurrentSmoker	CigsPerDay	BPMeds	PrevalentStroke	PrevalentHyp
0	1	39	1	0	0.0	0.0	0	0
1	0	46	0	0	0.0	0.0	0	0
2	1	48	0	1	20.0	0.0	0	0
3	0	61	1	1	30.0	0.0	0	1
4	0	46	1	1	23.0	0.0	0	0

Fig 2: Framingham Heart Disease Data

iv) Data Processing:

Data processing is changing over crude information into valuable data for associations. Data researchers frequently process information by social occasion, coordinating, cleaning, approving, breaking down, and making an interpretation of data into justifiable portrayals like charts or papers. Data processing might be finished in three ways: physically, precisely, and electronically. The objective is to work on the worth of data and help direction. This permits associations to upgrade their tasks and pursue all the more opportune key choices. Mechanized data processing advancements, for example, PC programming, have a significant impact in this. It might assist with changing huge volumes of information, particularly enormous information, into valuable experiences for quality administration and navigation.

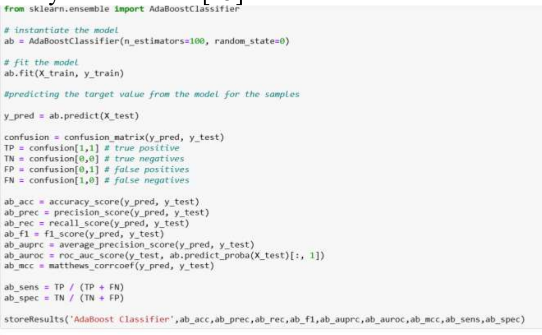
v) Feature selection:

Feature selection is the most common way of distinguishing the most predictable, non-excess, and applicable qualities for use in model creation. As data sets grow in amount and assortment, it is basic to purposefully diminish their size. The basic role of component choice is to build the exhibition of a prescient model while diminishing the computational expense of displaying.

Feature selection, one of the essential parts of component designing, is the demonstration of picking the main elements to take care of into ML calculations. Feature selection procedures are utilized to restrict the quantity of info factors by eliminating excess or superfluous elements and zeroing in on the qualities that are generally valuable to the ML model. The significant benefits of leading component determination early instead of depending on the ML model to figure out which qualities are huge.

vi) Algorithms:

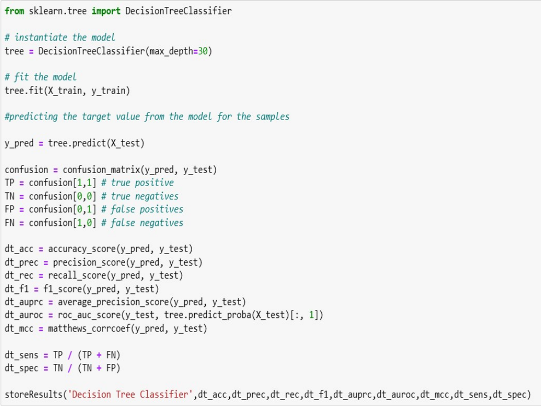
AdaBoost is an ensemble learning approach that joins feeble students (generally choice trees) to deliver a strong classifier. AdaBoost might be utilized to work on the presentation of feeble students (e.g., decision trees) in the group, bringing about better expectation exactness of coronary heart disease [25].



```
from sklearn.ensemble import AdaBoostClassifier

# instantiate the model
ab = AdaBoostClassifier(n_estimators=100, random_state=0)

# fit the model
ab.fit(X_train, y_train)

#predicting the target value from the model for the samples

y_pred = ab.predict(X_test)

confusion = confusion_matrix(y_pred, y_test)
TP = confusion[1,1] # true positive
TN = confusion[0,0] # true negatives
FP = confusion[0,1] # false positives
FN = confusion[1,0] # false negatives

ab_acc = accuracy_score(y_pred, y_test)
ab_prec = precision_score(y_pred, y_test)
ab_rec = recall_score(y_pred, y_test)
ab_f1 = f1_score(y_pred, y_test)
ab_auprc = average_precision_score(y_pred, y_test)
ab_auroc = roc_auc_score(y_test, ab.predict_proba(X_test)[:, 1])
ab_mcc = matthews_corrcoef(y_pred, y_test)

ab_sens = TP / (TP + FN)
ab_spec = TN / (TN + FP)

storeResults('AdaBoost Classifier',ab_acc,ab_prec,ab_rec,ab_f1,ab_auprc,ab_auroc,ab_mcc,ab_sens,ab_spec)
```

Fig 3: Adaboost

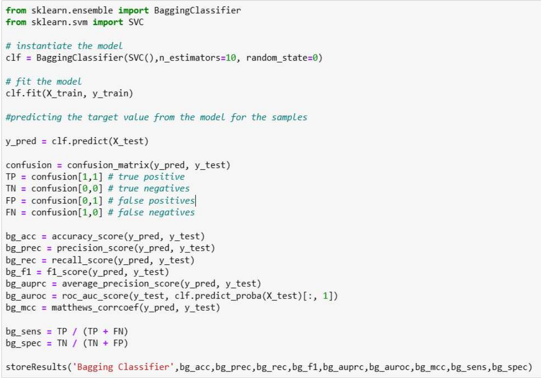
A **decision tree** is a flowchart-like design in which each center hub addresses a component, the branch addresses a choice rule, and the leaf hubs reflect results. Decision trees were utilized as base students in gathering approaches like as AdaBoost and Sacking to work on the expectation of coronary heart disease [22].



```
from sklearn.tree import DecisionTreeClassifier

# instantiate the model
tree = DecisionTreeClassifier(max_depth=30)

# fit the model
tree.fit(X_train, y_train)

#predicting the target value from the model for the samples

y_pred = tree.predict(X_test)

confusion = confusion_matrix(y_pred, y_test)
TP = confusion[1,1] # true positive
TN = confusion[0,0] # true negatives
FP = confusion[0,1] # false positives
FN = confusion[1,0] # false negatives

dt_acc = accuracy_score(y_pred, y_test)
dt_prec = precision_score(y_pred, y_test)
dt_rec = recall_score(y_pred, y_test)
dt_f1 = f1_score(y_pred, y_test)
dt_auprc = average_precision_score(y_pred, y_test)
dt_auroc = roc_auc_score(y_test, tree.predict_proba(X_test)[:, 1])
dt_mcc = matthews_corrcoef(y_pred, y_test)

dt_sens = TP / (TP + FN)
dt_spec = TN / (TN + FP)

storeResults('Decision Tree Classifier',dt_acc,dt_prec,dt_rec,dt_f1,dt_auprc,dt_auroc,dt_mcc,dt_sens,dt_spec)
```

Fig 4: Decision tree

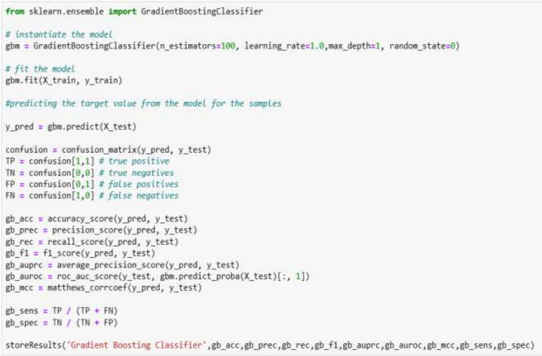
Bagging (Bootstrap Aggregating) involves building various models from particular subsets of the preparation dataset and averaging their forecasts to increment model exactness. Bagging was utilized to create an outfit of models, which further developed prediction accuracy with regards to CHD prediction [26].



```
from sklearn.ensemble import BaggingClassifier
from sklearn.svm import SVC

# instantiate the model
clf = BaggingClassifier(SVC(),n_estimators=10, random_state=0)

# fit the model
clf.fit(X_train, y_train)

#predicting the target value from the model for the samples

y_pred = clf.predict(X_test)

confusion = confusion_matrix(y_pred, y_test)
TP = confusion[1,1] # true positive
TN = confusion[0,0] # true negatives
FP = confusion[0,1] # false positives
FN = confusion[1,0] # false negatives

bg_acc = accuracy_score(y_pred, y_test)
bg_prec = precision_score(y_pred, y_test)
bg_rec = recall_score(y_pred, y_test)
bg_f1 = f1_score(y_pred, y_test)
bg_auprc = average_precision_score(y_pred, y_test)
bg_auroc = roc_auc_score(y_test, clf.predict_proba(X_test)[:, 1])
bg_mcc = matthews_corrcoef(y_pred, y_test)

bg_sens = TP / (TP + FN)
bg_spec = TN / (TN + FP)

storeResults('Bagging Classifier',bg_acc,bg_prec,bg_rec,bg_f1,bg_auprc,bg_auroc,bg_mcc,bg_sens,bg_spec)
```

Fig 5: Bagging

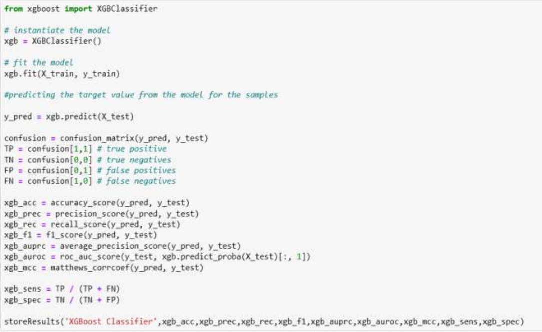
Gradient Boosting areas of strength for produces models by more than once consolidating the expectations of powerless models while limiting a loss capability. Gradient Boosting was utilized to create a bunch of models that iteratively further developed expectation accuracy for CHD [25].



```
from sklearn.ensemble import GradientBoostingClassifier

# instantiate the model
gbm = GradientBoostingClassifier(n_estimators=100, learning_rate=1.0,max_depth=1, random_state=0)

# fit the model
gbm.fit(X_train, y_train)

#predicting the target value from the model for the samples

y_pred = gbm.predict(X_test)

confusion = confusion_matrix(y_pred, y_test)
TP = confusion[1,1] # true positive
TN = confusion[0,0] # true negatives
FP = confusion[0,1] # false positives
FN = confusion[1,0] # false negatives

gb_acc = accuracy_score(y_pred, y_test)
gb_prec = precision_score(y_pred, y_test)
gb_rec = recall_score(y_pred, y_test)
gb_f1 = f1_score(y_pred, y_test)
gb_auprc = average_precision_score(y_pred, y_test)
gb_auroc = roc_auc_score(y_test, gbm.predict_proba(X_test)[:, 1])
gb_mcc = matthews_corrcoef(y_pred, y_test)

gb_sens = TP / (TP + FN)
gb_spec = TN / (TN + FP)

storeResults('Gradient Boosting Classifier',gb_acc,gb_prec,gb_rec,gb_f1,gb_auprc,gb_auroc,gb_mcc,gb_sens,gb_spec)
```

Fig 6: Gradient boosting

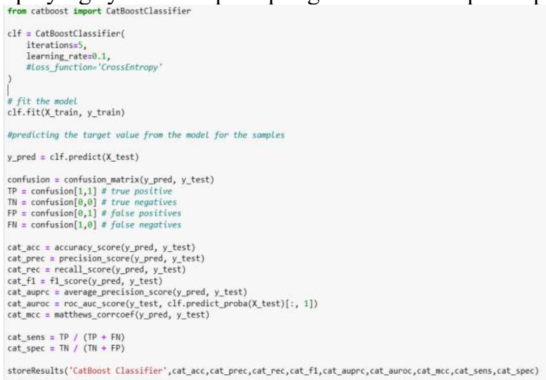
XGBoost (Extreme Gradient Boosting) is an exceptionally effective and scalable gradient boosting calculation. XGBoost was utilized as a boosting calculation to further develop the expectation accuracy of CHD [25].



```
from xgboost import XGBClassifier

# instantiate the model
xgb = XGBClassifier()

# fit the model
xgb.fit(X_train, y_train)

#predicting the target value from the model for the samples

y_pred = xgb.predict(X_test)

confusion = confusion_matrix(y_pred, y_test)
TP = confusion[1,1] # true positive
TN = confusion[0,0] # true negatives
FP = confusion[0,1] # false positives
FN = confusion[1,0] # false negatives

xgb_acc = accuracy_score(y_pred, y_test)
xgb_prec = precision_score(y_pred, y_test)
xgb_rec = recall_score(y_pred, y_test)
xgb_f1 = f1_score(y_pred, y_test)
xgb_auprc = average_precision_score(y_pred, y_test)
xgb_auroc = roc_auc_score(y_test, xgb.predict_proba(X_test)[:, 1])
xgb_mcc = matthews_corrcoef(y_pred, y_test)

xgb_sens = TP / (TP + FN)
xgb_spec = TN / (TN + FP)

storeResults('XGBoost Classifier',xgb_acc,xgb_prec,xgb_rec,xgb_f1,xgb_auprc,xgb_auroc,xgb_mcc,xgb_sens,xgb_spec)
```

Fig 7: XGBoost

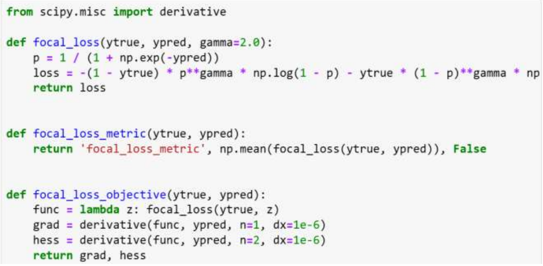
CatBoost is an gradient boosting toolbox that handles classification includes really. It consequently handles absolute information without the prerequisite for preprocessing, like one-hot encoding. CatBoost was used to deal with the dataset's all out qualities, working on the displaying system and prompting further developed expectations [24].



```
from catboost import CatBoostClassifier

clf = CatBoostClassifier(
    iterations=5,
    learning_rate=0.1,
    #loss_function='CrossEntropy'
)

# fit the model
clf.fit(X_train, y_train)

#predicting the target value from the model for the samples

y_pred = clf.predict(X_test)

confusion = confusion_matrix(y_pred, y_test)
TP = confusion[1,1] # true positive
TN = confusion[0,0] # true negatives
FP = confusion[0,1] # false positives
FN = confusion[1,0] # false negatives

cat_acc = accuracy_score(y_pred, y_test)
cat_prec = precision_score(y_pred, y_test)
cat_rec = recall_score(y_pred, y_test)
cat_f1 = f1_score(y_pred, y_test)
cat_auprc = average_precision_score(y_pred, y_test)
cat_auroc = roc_auc_score(y_test, clf.predict_proba(X_test)[:, 1])
cat_mcc = matthews_corrcoef(y_pred, y_test)

cat_sens = TP / (TP + FN)
cat_spec = TN / (TN + FP)

storeResults('CatBoost Classifier',cat_acc,cat_prec,cat_rec,cat_f1,cat_auprc,cat_auroc,cat_mcc,cat_sens,cat_spec)
```

Fig 8: Catboost

LightGBM is an gradient boosting structure, while Focal Loss is a modified loss capability that remedies class lopsidedness by zeroing in on hard to-order information. LightGBM with Focal Loss was utilized to increment responsiveness in predicting CHD, especially within the sight of uneven information, by zeroing in additional on testing circumstances.

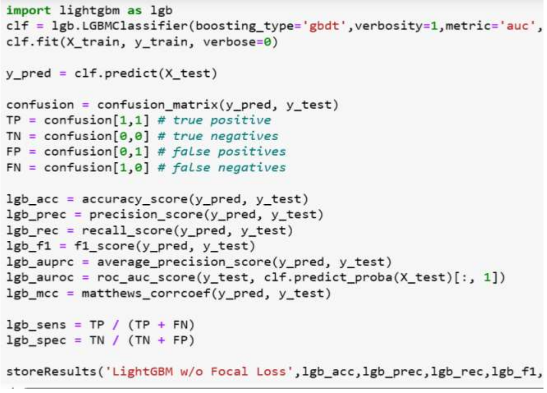


```
from scipy.misc import derivative

def focal_loss(ytrue, ypred, gamma=2.0):
    p = 1 / (1 + np.exp(-ypred))
    loss = -(1 - ytrue) * p**gamma * np.log(1 - p) - ytrue * (1 - p)**gamma * np
    return loss


def focal_loss_metric(ytrue, ypred):
    return 'focal_loss_metric', np.mean(focal_loss(ytrue, ypred)), False


def focal_loss_objective(ytrue, ypred):
    func = lambda z: focal_loss(ytrue, z)
    grad = derivative(func, ypred, n=1, dx=1e-6)
    hess = derivative(func, ypred, n=2, dx=1e-6)
    return grad, hess
```

Fig 9: Light GBM

This alludes to utilizing Light GBM's customary loss calculations instead of the Focal Loss capability. LightGBM without

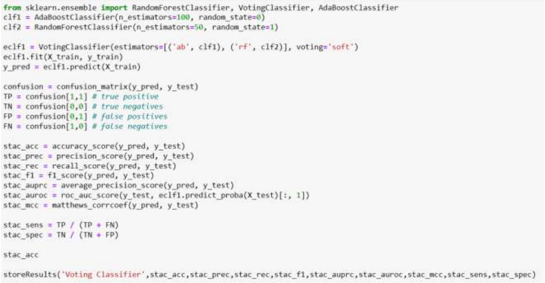
Focal Loss filled in as a benchmark for contrasting and assessing Focal Loss's effect on coronary heart disease prediction.



```
import lightgbm as lgb
clf = lgb.LGBMClassifier(boosting_type='gbdt',verbosity=1,metric='auc',
clf.fit(X_train, y_train, verbose=0)

y_pred = clf.predict(X_test)

confusion = confusion_matrix(y_pred, y_test)
TP = confusion[1,1] # true positive
TN = confusion[0,0] # true negatives
FP = confusion[0,1] # false positives
FN = confusion[1,0] # false negatives

lgb_acc = accuracy_score(y_pred, y_test)
lgb_prec = precision_score(y_pred, y_test)
lgb_rec = recall_score(y_pred, y_test)
lgb_f1 = f1_score(y_pred, y_test)
lgb_auprc = average_precision_score(y_pred, y_test)
lgb_auroc = roc_auc_score(y_test, clf.predict_proba(X_test)[:, 1])
lgb_mcc = matthews_corrcoef(y_pred, y_test)

lgb_sens = TP / (TP + FN)
lgb_spec = TN / (TN + FP)

storeResults('LightGBM w/o Focal Loss',lgb_acc,lgb_prec,lgb_rec,lgb_f1,
```

Fig 10: LightGBM without Focal Loss

A **Voting Classifier** is a group approach that joins expectations from various separate models to conjecture the class name in view of a greater part vote. In this trial, a Voting Classifier was utilized related to Random Forest (RF) and AdaBoost models to expand the qualities of the two models, fully intent on further developing expectation accuracy for CHD.



```
from sklearn.ensemble import RandomForestClassifier, VotingClassifier, AdaBoostClassifier
clf1 = AdaBoostClassifier(n_estimators=100, random_state=0)
clf2 = RandomForestClassifier(n_estimators=50, random_state=1)

eclf1 = VotingClassifier(estimators=[('ab', clf1), ('rf', clf2)], voting='soft')
eclf1.fit(X_train, y_train)
y_pred = eclf1.predict(X_train)

confusion = confusion_matrix(y_pred, y_test)
TP = confusion[1,1] # true positive
TN = confusion[0,0] # true negatives
FP = confusion[0,1] # false positives
FN = confusion[1,0] # false negatives

stac_acc = accuracy_score(y_pred, y_test)
stac_prec = precision_score(y_pred, y_test)
stac_rec = recall_score(y_pred, y_test)
stac_f1 = f1_score(y_pred, y_test)
stac_auprc = average_precision_score(y_pred, y_test)
stac_auroc = roc_auc_score(y_test, eclf1.predict_proba(X_test)[:, 1])
stac_mcc = matthews_corrcoef(y_pred, y_test)

stac_sens = TP / (TP + FN)
stac_spec = TN / (TN + FP)

stac_acc

storeResults('Voting Classifier',stac_acc,stac_prec,stac_rec,stac_f1,stac_auprc,stac_auroc,stac_mcc,stac_sens,stac_spec)
```

Fig 11: Voting classifier

3. EXPERIMENTAL RESULTS

Precision: Precision estimates the extent of precisely characterized cases or tests among those classified as certain. Hence, the precision can be determined utilizing the accompanying formula:

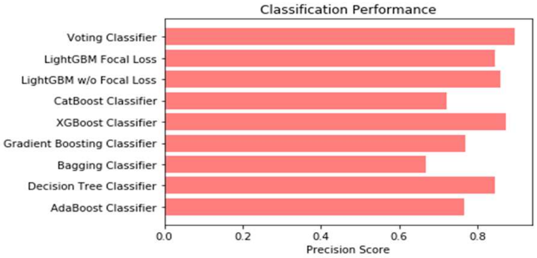


Fig 6: Precision comparison graph

Recall: Recall is an ML metric that evaluates a model's capacity to perceive all occasions of a given class. It is the proportion of accurately anticipated positive perceptions to add up to real up-sides, which gives data on a model's fulfillment in gathering instances of a particular class.

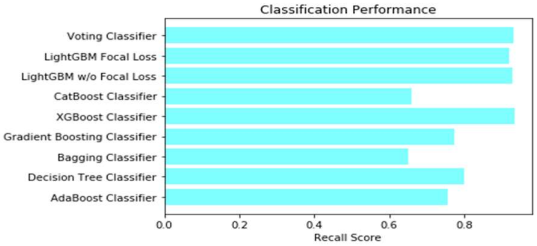


Fig 7: Recall comparison graph

Accuracy: Accuracy is characterized as the extent of right forecasts in a grouping position, which estimates a model's general accuracy.

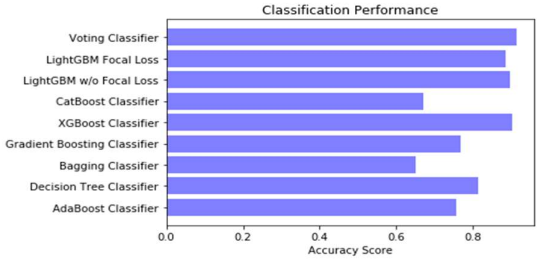


Fig 8: Accuracy graph

F1 Score: The F1 score is a symphonic medium of precision and validation, and is legitimate for unbalanced datasets because it represents a useful measure that captures both false positives and false negatives.

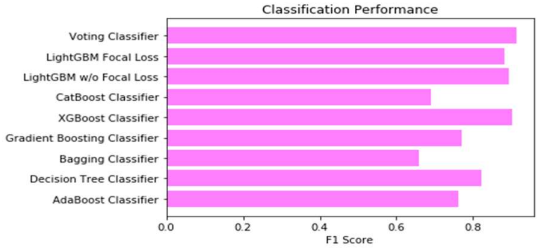
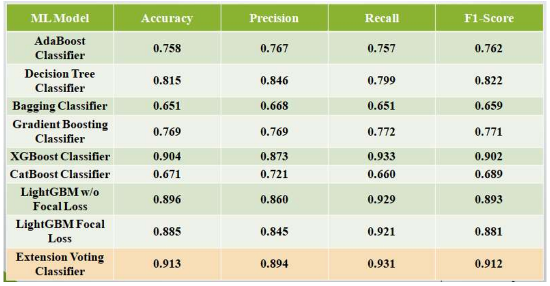


Fig 9: F1Score



ML Model	Accuracy	Precision	Recall	F1-Score
AdaBoost Classifier	0.758	0.767	0.757	0.762
Decision Tree Classifier	0.815	0.846	0.799	0.822
Bagging Classifier	0.651	0.668	0.651	0.659
Gradient Boosting Classifier	0.769	0.769	0.772	0.771
XGBoost Classifier	0.904	0.873	0.933	0.902
CatBoost Classifier	0.671	0.721	0.660	0.689
LightGBM w/o Focal Loss	0.896	0.860	0.929	0.893
LightGBM Focal Loss	0.885	0.845	0.921	0.881
Extension Voting Classifier	0.913	0.894	0.931	0.912

Fig 10: Performance Evaluation

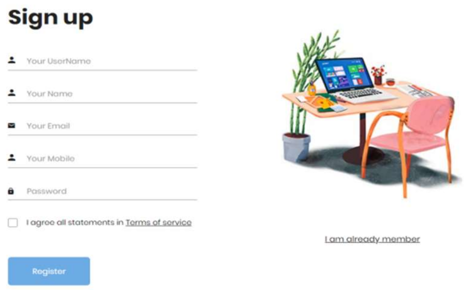
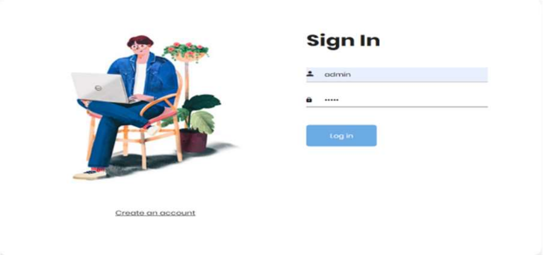


Fig 12 Signin page



Sign In

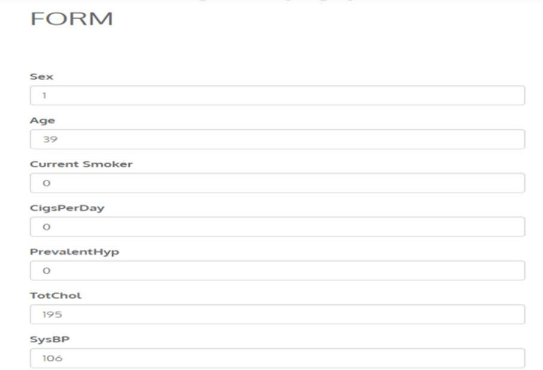
admin

•••••

Log in

Create an account

Fig 13: Login page



FORM

Sex
1

Age
39

Current Smoker
0

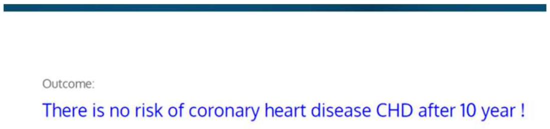
CigsPerDay
0

PrevalentHyp
0

TotChol
195

SysBP
106

Fig 14: User input



Outcome:

There is no risk of coronary heart disease CHD after 10 year !

Fig 15: Predict result for given input

4. CONCLUSION

Utilizing an improved LightGBM classifier and an updated loss function, the HY_OptGBM expectation model precisely predicts CHD. Precision, recall, F score, and accuracy are utilized to assess the model's expectation abilities. Enhancers utilize strong classifiers and misfortune capabilities to further develop the HY_OptGBM model. These progressions increment the model's CHD location and forecast accuracy [2], [3], [4], [5], [6]. A ensemble procedure joins expectations from various models to further develop framework accuracy and resilience. High level ensemble draws near, such the Voting Classifier, accomplish close to 100% accuracy, showing that shifted models support prescient execution. A easy-to-use Flask interface with secure confirmation further develops framework testing. This point of interaction works on information section for framework execution assessment, ensuring ease of use and security.

5. FUTURE SCOPE

To further develop the HY_OptGBM model's coronary heart disease prediction, future examination can add qualities or information sources. Medical data might be incorporated for a total comprehension. Further review ought to test the model's generalizability and flexibility on greater and more differed datasets. This will uncover how really the model adjusts to information circulations. Contrasting the HY_OptGBM model with other refined ML strategies [12,13] for CHD expectation can assist with deciding its viability and prevalence. You might utilize the proposed method to figure coronary heart disease as well as other cardiovascular diseases or circumstances. This expansion can change cardiology by giving an adaptable determining tool.

REFERENCES

[1] N. Katta, T. Loethen, C. J. Lavie, and M. A. Alpert, ‘‘Obesity and coronary heart disease: Epidemiology, pathology, and coronary artery imaging,’’ Current Problems Cardiol., vol. 46, no. 3, Mar. 2021, Art. no. 100655, doi: 10.1016/j.cpcardiol.2020.100655.

[2] G. S. Handelman, H. K. Kok, R. V. Chandra, A. H. Razavi, M. J. Lee, and H. Asadi, ‘‘EDoctor: Machine learning and the future of medicine,’’ J. Internal Med., vol. 284, no. 6, pp. 603–619, Sep. 2018, doi: 10.1111/joim.12822.

[3] E. L. Romm and I. F. Tsigelny, ‘‘Artificial intelligence in drug treatment,’’ Annu. Rev. Pharmacol. Toxicol., vol. 60, no. 1, pp. 353–

369, Jan. 2020, doi: 10.1146/annurev-pharmtox-010919-023746.
[4] L. Lo Vercio, K. Amador, J. J. Bannister, S. Crites, A. Gutierrez, M. E. MacDonald, J. Moore, P. Mouches, D. Rajashekar, S. Schimert, N. Subbanna, A. Tuladhar, N. Wang, M. Wilms, A. Winder, and N. D. Forkert, ''Supervised machine learning tools: A tutorial for clinicians,'' J. Neural Eng., vol. 17, no. 6, Dec. 2020, Art. no. 062001, doi: 10.1088/1741-2552/abbff2.
[5] S. Rauschert, K. Raubenheimer, P. E. Melton, and R. C. Huang, ''Machine learning and clinical epigenetics: A review of challenges for diagnosis and classification,'' Clin. Epigenetics, vol. 12, no. 1, p. 51, Apr. 2020, doi: 10.1186/s13148-020-00842-4.
[6] Y. Arfat, G. Mittone, R. Esposito, B. Cantalupo, G. M. De Ferrari, and M. Aldinucci, ''Machine learning for cardiology,'' Minerva Cardiol. Angiol., vol. 70, no. 1, pp. 75–91, Mar. 2022, doi: 10.23736/s2724- 5683.21.05709-4.
[7] S. Nematzadeh, F. Kiani, M. Torkamanian-Afshar, and N. Aydin, ''Tuning hyperparameters of machine learning algorithms and deep neural networks using metaheuristics: A bioinformatics study on biomedical and biological cases,'' Comput. Biol. Chem., vol. 97, Apr. 2022, Art. no. 107619, doi: 10.1016/j.compbiolchem.2021.107619.
[8] M. Liang, B. An, K. Li, L. Du, T. Deng, S. Cao, Y. Du, L. Xu, X. Gao, L. Zhang, J. Li, and H. Gao, ''Improving genomic prediction with machine learning incorporating TPE for hyperparameters optimization,'' Biology, vol. 11, no. 11, p. 1647, Nov. 2022, doi: 10.3390/ biology11111647.
[9] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, ''OPTUNA: A nextgeneration hyperparameter optimization framework,'' in Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Anchorage, AK, USA, 2019, pp. 2623–2631.
[10] M. Yeung, E. Sala, C.-B. Schönlieb, and L. Rundo, ''Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation,'' Computerized Med. Imag. Graph., vol. 95, Jan. 2022, Art. no. 102026, doi: 10.1016/j.compmedimag.2021.102026.
[11] G. Ke et al., ''LightGBM: A highly efficient gradient boosting decision tree,'' in Proc. 31st Int. Conf. Neural Inf. Process. Syst., Long Beach, CA, USA, 2017, pp. 3149–3157.
[12] O. Goldman, O. Raphaeli, E. Goldman, and M. Leshno, ''Improvement in the prediction of coronary heart disease risk by using artificial neural networks,'' Qual. Manage. Health Care, vol. 30, no. 4, pp. 244–250, Jul. 2021, doi: 10.1097/qmh.0000000000000309.
[13] Z. Du, Y. Yang, J. Zheng, Q. Li, D. Lin, Y. Li, J. Fan, W. Cheng, X.-H. Chen, and Y. Cai, ''Accurate prediction of coronary heart disease for patients with hypertension from electronic health records with big data and machine-learning methods: Model development and performance evaluation,'' JMIR Med. Informat., vol. 8, no. 7, Jul. 2020, Art. no. e17257, doi: 10.2196/17257.
[14] J. K. Kim and S. Kang, ''Neural network-based coronary heart disease risk prediction using feature correlation analysis,'' J. Healthcare Eng., vol. 2017, Sep. 2017, Art. no. 2780501, doi: 10.1155/2017/2780501.
[15] C. Krittanawong, H. Zhang, Z. Wang, M. Aydar, and T. Kitai, ''Artificial intelligence in precision cardiovascular medicine,'' J. Amer. College Cardiol., vol. 69, no. 21, pp. 2657–2664, 2017, doi: 10.1016/j.jacc.2017.03.571.
[16] A. Akella and S. Akella, ''Machine learning algorithms for predicting coronary artery disease: Efforts toward an open source solution,'' Future Sci. OA, vol. 7, no. 6, Jul. 2021, Art. no. FSO698, doi: 10.2144/fsoa-2020- 0206.
[17] L. J. Muhammad, I. Al-Shourbaji, A. A. Haruna, I. A. Mohammed, A. Ahmad, and M. B. Jibrin, ''Machine learning predictive models for coronary artery disease,'' Social Netw. Comput. Sci., vol. 2, no. 5, p. 350, Sep. 2021, doi: 10.1007/s42979-021-00731-4.
[18] C. A. U. Hassan, J. Iqbal, R. Irfan, S. Hussain, A. D. Algarni, S. S. H. Bukhari, N. Alturki, and S. S. Ullah, ''Effectively predicting the presence of coronary heart disease using machine learning classifiers,'' Sensors, vol. 22, no. 19, p. 7227, Sep. 2022, doi: 10.3390/s22197227.
[19] Captainozlem. Framingham_CHD_Preprocessed_Data. Version 1. Accessed: May 5, 2020. [Online]. Available: https://www.kaggle. com/-datasets/captainozlem/framingham-chd-preprocesseddata/download?datasetVersionNumber=1
[20] V. Voillet, P. Besse, L. Liaubet, M. San Cristobal, and I. González, ''Handling missing rows in multi-omics data integration: Multiple imputation in multiple factor analysis framework,'' BMC Bioinf., vol. 17, no. 1, p. 402, Oct. 2016, doi: 10.1186/s12859-016-1273-5.
[21] G. Douzas and F. Bacao, ''Geometric SMOTE a geometrically enhanced drop-in replacement for SMOTE,'' Inf. Sci., vol. 501, pp. 118–135, Oct. 2019, doi: 10.1016/j.ins.2019.06.007.
[22] D. Che, Q. Liu, K. Rasheed, and X. Tao, ''Decision tree and ensemble learning algorithms with their applications in bioinformatics,'' in Software Tools and Algorithms for Biological Systems (Advances in Experimental Medicine and Biology), H. Arabnia and Q. N. Tran, Eds. New York, NY, USA: Springer, 2011, pp. 191–199.
[23] L. Yang, H. Wu, X. Jin, P. Zheng, S. Hu, X. Xu, W. Yu, and J. Yan, ''Study of cardiovascular disease prediction model based on random forest in eastern China,'' Sci. Rep., vol. 10, no. 1, p. 5245, Mar. 2020, doi: 10.1038/s41598-020-62133-5.
[24] J. T. Hancock and T. M. Khoshgoftaar, ''CatBoost for big data: An interdisciplinary review,'' J. Big Data, vol. 7, no. 1, p. 94, Nov. 2020, doi: 10.1186/s40537-020-00369-8.
[25] W. Wenbo, S. Yang, and C. Guici, ''Blood glucose concentration prediction based on VMD-KELM-adaboost,'' Med. Biol. Eng. Comput., vol. 59, nos. 11–12, pp. 2219–2235, Sep. 2021, doi: 10.1007/s11517-021-02430- x.