ADScAI
Applied Data Science & Artificial Intelligence Symposium

CatBoost and Random Forest Algorithms in Binary Classification Tasks.

Chathurangi Kahathota Liyanage
Department of Computer Science and Engineering
University of Moratuwa
Moratuwa, Sri Lanka
chathurangi.22@cse.mrt.ac.lk

Uthayasanker Thayasivam
Department of Computer Science and Engineering
University of Moratuwa
Moratuwa, Sri Lanka
rtuthaya@cse.mrt.ac.lk

***Keywords*—CatBoost, RandomForest, Bayesian Optimization, Model Architecture Selection**

I. Introduction

Among the many ML techniques, ensemble learning methods have grabbed significant attention due to their enhanced predictive performance achieved by combining multiple learning algorithms. Notably, ensemble methods like Random Forest, CatBoost have demonstrated superior capabilities in various predictive tasks, including numerous Kaggle competitions. In this study, I conduct an analysis of these two ensemble learning algorithms within the context of a binary classification task. This paper details the dataset selected, followed by the development of classification models using Random Forest, and CatBoost algorithms. I systematically evaluate and compare the performance of these models, analyzing the impact of hyperparameter optimization by Bayesian Optimization, model suitability based on features present, for both algorithms. The findings offer insights into the suitability of specific preprocessing techniques and model selections for each dataset, contributing to the optimization of ensemble learning applica tions in classification tasks.

II. Literature Review

CatBoost is an open source machine learning library implemented with the gradient boosting principle, but which can typically outperform other gradient boosting algorithms because CatBoost uses ordered boosting and a specific algorithm to handle categorical data.[1] Random forest is an ensemble learning method mainly used for classification and regression tasks and follows a bagging approach where multiple independent base learners are used. Since multiple decision trees are used in the model, overfitting is avoided. Baeysean optimization is an algorithm used to find the global optimum of a "black box" function. Recently it is being used for hyperparameter tuning of machine learning models.[3]

III. Materials and Methods

The dataset comprises responses from a broad survey conducted between January and June 2023, aiming to identify factors influencing depression risk among adults. Participants, aged between 18 and 60, anonymously provided information on various aspects. The survey encompassed individuals of diverse backgrounds across multiple cities, without involving professional mental health evaluations or diagnostic tests. The dataset contains 104 700 records, having the datatypes object, float and int. Since the presence of irrelevant unclean data could lead to wrong conclusions, the data was preprocessed well.

['Name', 'Gender', 'Age', 'City', 'Working Professional or Student', 'Profession', 'Academic Pressure', 'Work Pressure', 'CGPA', 'Study Satisfaction', 'Job Satisfaction', 'Sleep Duration', 'Dietary Habits', 'Degree', 'Have you ever had suicidal thoughts ?', 'Work/Study Hours', 'Financial Stress', 'Family History of Mental Illness', 'Depression']

Thereby I replaced irrelevant values and null values in the columns 'City', 'Working Professional or Student', 'Degree', 'CGPA', 'Profession'. The 'Sleep Duration' and 'Dietary Habits' columns were classified into 4 and 3 categories respectively, and were label encoded. The 'Profession' attribute of students were set to 'Student'.In addition to the present features, I added a new feature named ' Chronic Stress', which aggregated the features 'Academic Pressure', 'Work Pressure', 'Financial Stress' and 'Have you ever had suicidal thoughts?'. The dataset contains 14 677 records whose 'Profession' is 'Student' and the rest 126 023 records of data are of working professionals. Since the data representing students is less, the patterns of depression among students might not be well captured in this data. To calculate the accuracy, precision, recall of the model, I used train-test split of scikit learn and split train:test to a ratio of 4:1.

IV. Results and Discussion

The default hyperparameters of the CatBoost model are : iterations=1000,L2 leaf regularization=3,loss function=Logloss, depth=6,bootstrap type=Baeysean. I also used the pool data structure in CatBoost with 50 early stopping rounds. Table I shows the statistics received for the CatBoost model with default parameters. Fig 2 shows the variation of accuracy of

https://doi.org/10.31705/ADScAI.2025.29

predictions with learning rate and tree depth. Following are the hyperparameters that were selected as best parameters by Bayesian optimization.

```
{ 'iterations': 2688,'depth': 5,
 'learning_rate': 0.046635440720980324,
 'l2_leaf_reg': 0.1649522723758651,
 'subsample': 0.7957835059926914,
 'colsample_bylevel': 0.5331739178037092}
```

Table II shows the statistics received upon training the model with the above parameters. Table III shows the statistics obtained by applying Random Forest model with default parameters. The best hyperparameters obtained from Bayesian optimization for RF are :

```
OrderedDict([('bootstrap',True),('max_depth'
,41),('max_features','log2'),
('min_samples_leaf,20),('min_samples_split'
,3),('n_estimators',431)])
```

Based on Fig 1,we can observe that for lower tree depths , higher learning rates give higher accuracy, but for higher tree depths, lower learning rates give better accuracy.This is because lower tree depths causes underfitting and higher tree depths can cause overfitting. Table IV shows the statistics obtained from the Random Forest model trained accordingly. The performance differences observed in the two models reveal the importance of model architecture in handling categorical data, and the importance of selecting a model suitable to the nature of data in the dataset under analysis[3]. I can observe that CatBoost gives more accuracy for models having more categorical features. On the other hand, Random Forest uses encoding methods like label encoding, target encoding, mean encoding to handle categorical features. This will lead to an increase of dimensionality and bias in assigning feature importance . The post-BO improvement in RF can be attributed to hyperparameters that constrained model complexity thereby reducing overfitting on noisy splits induced by one-hot encoding. Conversely, CatBoost's slight accuracy drop after BO suggests that its default hyperparameters—already finetuned for regularization and categorical robustness—were dropped into suboptimal configurations. Notably, CatBoost's sustained superiority post-BO highlights the limits of hyperparameter tuning in overcoming architectural limitations. While RF's bagging mechanism averages predictions from independent trees, it struggles to capture interactions in high-dimensional categorical spaces. CatBoost's gradient boosting, iteratively corrects errors using gradient-guided splits, which are more effective in such settings. RF's reliance on one-hot encoding increases the "curse of dimensionality," diluting the signal from categorical features. These results emphasize that model selection should prioritize architectural alignment with data characteristics before optimization.

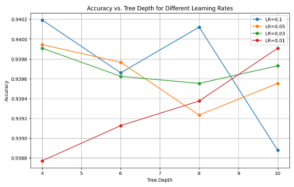

Fig. 1. Variation of CatBoost model accuracy with tree depth and learning rate.

TABLE I
CATBOOST MODEL WITH DEFAULT HYPERPARAMETERS

Class	Precision	Recall	F1-score	Support
0	0.96	0.97	0.96	22986
1	0.85	0.82	0.83	5154
Accuracy	0.9399			

TABLE II
CATBOOST MODEL AFTER BAYESIAN OPTIMIZATION

Class	Precision	Recall	F1-score	Support
0	0.96	0.97	0.96	22986
1	0.85	0.82	0.83	5154
Accuracy	0.9396			

TABLE III
RANDOM FOREST MODEL WITH DEFAULT HYPERPARAMETERS

Class	Precision	Recall	F1-score	Support
0	0.94	0.95	0.94	23027
1	0.75	0.71	0.73	5113
Accuracy	0.9042			

TABLE IV
RANDOM FOREST MODEL AFTER BAYESIAN OPTIMIZATION

Class	Precision	Recall	F1-score	Support
0	0.94	0.96	0.95	23027
1	0.80	0.75	0.77	5113
Accuracy	0.9199			

V. CONCLUSION

This study demonstrates that CatBoost outperforms Random Forest in this binary classification task with a categoricaldominated dataset, both before and after Bayesian Optimization. Native categorical handling of CatBoost provides improved accuracy over RF in binary classification, for a model with considerably large fraction of categorical variables. RF increases accuracy after BO, but it is not as accurate as CatBoost, tuning cannot fully compensate for architectural limitations. Hyperparameter tuning should be applied to models with caution as it might lead to decrease in accuracy of models.

REFERENCES

[1] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased Boosting with Categorical Features," *arXiv preprint arXiv:1706.09516*, 2017. Available: https://github.com/catboost/catboost

[2] P. I. Frazier, "A Tutorial on Bayesian Optimization," *arXiv preprint arXiv:1807.02811*, 2018. Available: https://arxiv.org/abs/1807.02811