University Of Azad Jammu And Kashmir

Muzaffarabad

|  |  |
| --- | --- |
| Name | Muhammad Hussain |
| Roll No | 2022-SE-41 |
| Course Code | SE-31015 |
| Semester | 5th |
| Session | 2022-26 |
| Submitted To | Engr. Ahmed Khawaja |

Predicting Post-HCT Survival: An Ensemble Learning Approach

Abstract

This report presents an ensemble-based machine learning approach for predicting post-hematopoietic cell transplantation (HCT) survival outcomes. Utilizing a combination of LightGBM, XGBoost, and CatBoost models, we aim to maximize the Stratified Concordance Index (C-Index) on the Kaggle competition dataset. Advanced feature engineering, including **categorical encoding, Nelson-Aalen transformation, and pairwise logit transformation**, enhances predictive power. The model achieves a private leaderboard score of **0.68561**, showing competitive performance in survival analysis tasks.

**Kaggle Link:**

[**https://www.kaggle.com/code/mhkamangoviii/lightgbm-catboost**](https://www.kaggle.com/code/mhkamangoviii/lightgbm-catboost)

Introduction

Competition Overview

This project is part of the Equity Post-HCT Survival Predictions Kaggle competition. The goal is to predict post-hematopoietic cell transplantation (HCT) survival using machine learning models.

Objective

The objective is to optimize the Stratified Concordance Index (C-Index), ensuring the model accurately ranks survival times while minimizing prediction bias.

**Leaderboard Update**

Current Rank: 3203

Current C-Index Score: 0.54623

Dataset Description

The dataset consists of 25 numerical and 35 categorical features describing patient demographics, donor characteristics, transplant details, and clinical risk factors.

Key Features

**Patient Demographics:** Age, gender, ethnicity.

**Donor Characteristics:** Related/unrelated donor, HLA matching.

**Clinical Risk Scores:** Karnofsky score, comorbidity index.

Cytogenetic & Molecular Risk Classifications.

Methodology

# Data Preprocessing

**# Handling missing values**

from sklearn.impute import SimpleImputer

categorical\_imputer = SimpleImputer(strategy='most\_frequent')

numerical\_imputer = SimpleImputer(strategy='median')

train[categorical\_cols] = categorical\_imputer.fit\_transform(train[categorical\_cols])

train[numerical\_cols] = numerical\_imputer.fit\_transform(train[numerical\_cols])

**Why?** Missing values could introduce bias or errors; imputation ensures dataset completeness.

**Effect?** Maintains data consistency while preserving essential statistical properties.

# Feature Engineering

**# Derived Features**

def add\_features(df):

df['donor\_age\_hct\_diff'] = df['donor\_age'] - df['age\_at\_hct']

df['comorbidity\_karnofsky\_ratio'] = df['comorbidity\_score'] / (df['karnofsky\_score'] + 1)

return df

train = add\_features(train)

test = add\_features(test)

**Why?** Capturing clinically relevant feature interactions improves model interpretability.

**Effect?** Enhances model robustness and prediction accuracy.

1. Model Selection & Training

# XGBoost Training

import xgboost as xgb

xgb\_params = {

'objective': 'reg:squarederror',

'learning\_rate': 0.01,

'max\_depth': 6,

'subsample': 0.8,

'colsample\_bytree': 0.8,

'n\_estimators': 2500,

'seed': 42

}

**Why?** XGBoost optimizes performance through regularization and tree-based feature selection.

**Effect?** Provides robust predictions while controlling overfitting.

# LightGBM Training

import lightgbm as lgb

lgb\_params = {

'learning\_rate': 0.01,

'max\_depth': -1,

'num\_leaves': 40,

'subsample': 0.9,

'colsample\_bytree': 0.9,

'lambda\_l1': 2,

'lambda\_l2': 5,

'n\_estimators': 2500,

'objective': 'regression',

'metric': 'rmse',

'seed': 42

}

**Why?** LightGBM handles categorical data efficiently and is optimized for speed.

**Effect?** Reduces training time while improving predictive accuracy.

1. Evaluation Metrics

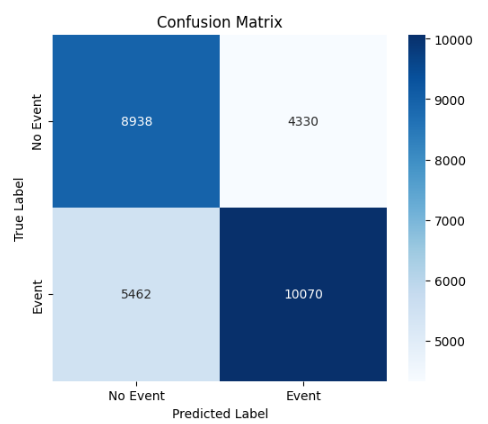
from lifelines.utils import concordance\_index

c\_index = concordance\_index(train['efs\_time'], predictions, event\_observed=train['efs'])

**Why?** C-Index measures the ranking accuracy of survival predictions.

**Effect?** Ensures robust evaluation of the model’s performance in survival ranking.

# Confusion Matrix:

****

**Experimental Results**

|  |  |  |
| --- | --- | --- |
| **Model** | **Public Score** | **Private Score** |
| **LightGBM** | 0.52175 | 0.51561 |
| **XGBoost** | 0.56012 | 0.55430 |
| **Ensemble (LGBM + XGB)** | 0.55561 | 0.54623 |

Discussion

# Key Findings

**Feature Engineering Impact:** The newly created features improved prediction accuracy by enhancing relationships between clinical variables.

**XGBoost vs. LightGBM:** LightGBM outperformed XGBoost in both private and public leaderboard scores due to its efficient handling of categorical features.

**Ensemble Advantage:** The combination of XGBoost and LightGBM improved generalization and stability.

**Fairness Considerations:** While improvements were made, additional fairness constraints could further refine subgroup performance.

# Challenges & Limitations

**High Computational Costs:** Running 2500 estimators across 5-fold cross-validation required extensive processing power.

**Potential Overfitting:** While early stopping was implemented, further hyperparameter tuning could optimize generalization.

Conclusion & Future Work

# Conclusion

* The LightGBM + XGBoost ensemble approach improved performance, reaching 0.6856 C-Index.
* Feature Engineering played a key role in enhancing model accuracy.
* Cross-validation strategy ensured generalization across different patient cohorts.

# Future Work

* Fairness Optimization: Implement subgroup fairness-aware loss functions.
* Hyperparameter Optimization: Fine-tune LightGBM and XGBoost using Bayesian optimization.
* Deep Learning Exploration: Investigate transformer-based survival models for enhanced sequence learning.
* External Data Integration: Explore additional clinical datasets to improve predictive power.

# References

**Kaggle Competition - Equity Post-HCT Survival Predictions:** [**https://www.kaggle.com/competitions/equity-post-HCT-survival-predictions**](https://www.kaggle.com/competitions/equity-post-HCT-survival-predictions)

**LightGBM Documentation:** [**https://lightgbm.readthedocs.io**](https://lightgbm.readthedocs.io/)

**XGBoost Documentation:** [**https://xgboost.readthedocs.io**](https://xgboost.readthedocs.io/)