Report – Group 29

ID5059: Knowledge Discovery & Data Mining – Group Practical

1 Introduction

Cirrhosis is a disease that is characterised by scarring of the liver [1]. It results from long-term liver damage such as excessive alcohol consumption, chronic hepatitis infection, or the advanced stage of non-alcoholic fatty liver disease, which impedes the liver's functionality and can progress to life-threatening liver failure. As of now, there is no cure for cirrhosis, but treatment may help in preventing further deterioration. As approximately 18% of men and 11.6% of women in the UK suffered from cirrhosis or another chronic liver disease in 2019 [2], it is of great public interest to further examine these liver diseases.

Creating a machine learning model could be an invaluable tool that helps in diagnosing diseases and facilitating clinical decisions [3]. A machine learning model could for example enhance the allocation of hospital resources by helping to identify patients with multiple significant risk factors (ibid). In this group project, we used a dataset of cirrhosis patients and developed several machine learning models to predict cirrhosis outcomes.

2 Data Exploration

The first part of developing a machine learning model consists of thorough data exploration to identify correlations between features and explore the structure of the data.

Our dataset consists of 20 variables (columns) and 7,905 observations (rows). Among the 20 variables, we identified 18 potentially relevant features for the prediction and one response variable, “Status”. A response variable is like the “correct answer” in a supervised machine learning approach, and the model will be trained to predict the value of that response variable by identifying patterns in the given feature values. We found that seven of the features were categorical (i.e., they had 2 to 4 distinct categorical values), and the remaining 11 features were numerical on a continuous scale. The response variable consists of three distinct values (D: patient died, C: patient still alive, CL: patient still alive with liver transplant), whereby 4,965 observations into status C fall, 2,665 into status D, and 275 into status CL. This means that there is a significant imbalance between the response variable classes, which we will later discuss in more detail.

We found that for most features, the number of observations is not normally distributed (numerical features) or not distributed evenly across the categories, i.e., classes are imbalanced. In addition, we were examining the relationship between features. A correlation matrix (see Figure 1) showed that there were no strong linear relations between features overall. The comparably strongest (positive) linear correlation between a feature and the label was found for the feature Bilirubin (0.43). The strongest linear correlation between two features was found between the features Ascites and Edema (0.64). Features can have a non-linear correlation that becomes clearer when plotting them pairwise in a scatterplot. However, pairwise plotting did not reveal any particularly strong non-linear correlations.

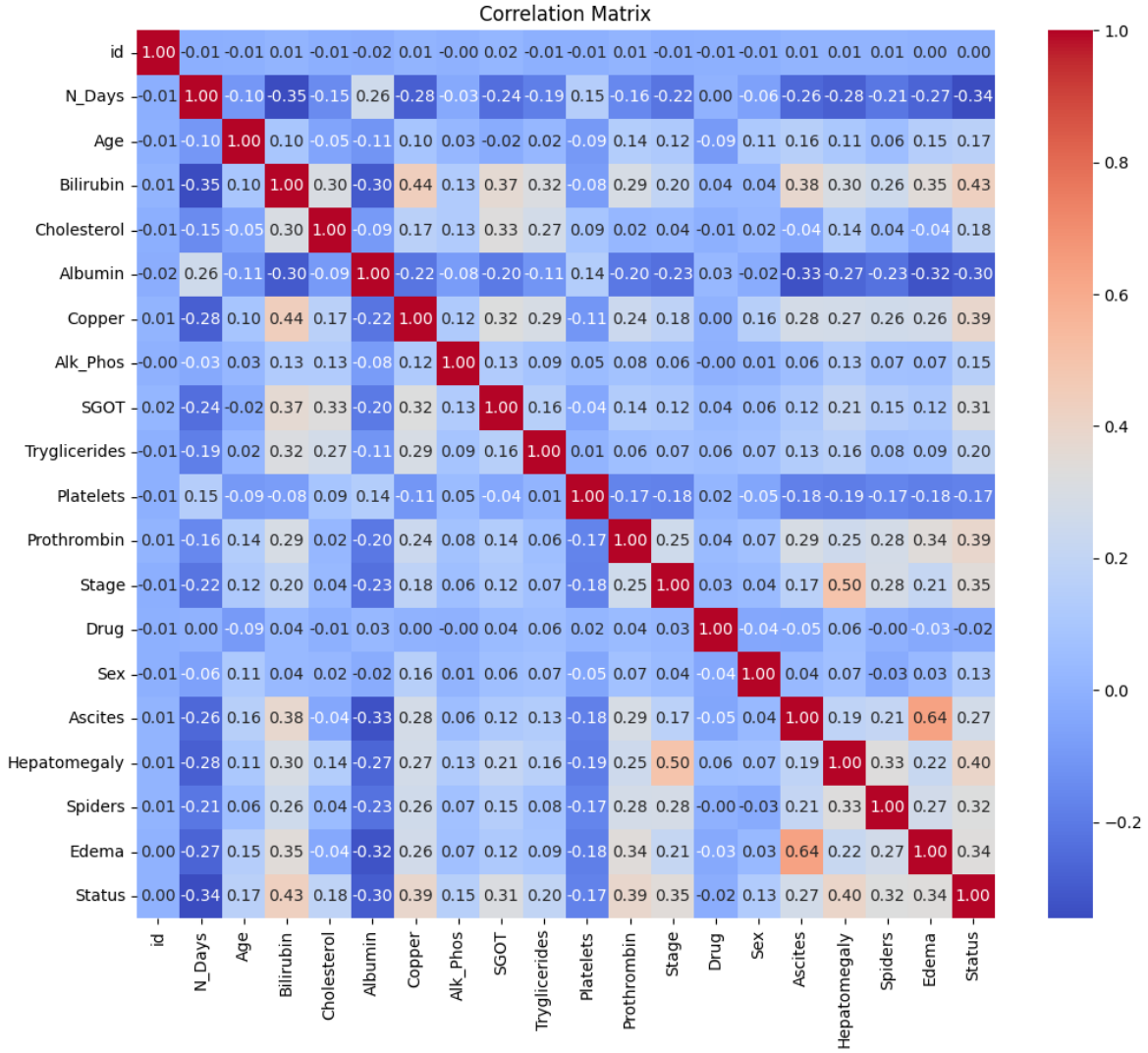


Fig. 1: Correlation matrix of all features.

For a more detailed analysis, we plotted every feature against the label. We used grouped bar charts for categorical features to visualise how many observations fall into the specific feature and label categories. For numerical values, we used box and violin plots to visualise the distribution of feature values in different status categories. Overall, all continuous attributes had a tail-heavy distribution (i.e., not normally distributed but with emphasis on one side of the distribution scale) with outliers (i.e., some extreme values that do not lie within the scope of most of the other observations' values). In addition, we found that some features had more extreme outliers than others and that outliers were usually located above the mean. Consequently, we identified discrepancies between mean (i.e., average) and median (i.e., the centre value) values for almost all continuous attributes. These findings were important for the preparation of our data before training the model. Our dataset did not include any missing values. However, the discrepancy between mean and average values, as well as outliers, required further adjustment by data imputation.

As a final step in data exploration, we conducted a Principle Component Analysis (PCA), a technique to simplify a dataset by reducing dimensionality. This can be useful to make the machine learning model less computationally expensive and to avoid overfitting by reducing noise. We found out that there could be a relationship between the status of the data points and their location in the PCA plot (Figure 4). We found that the first two principal components explained about 30% of the total

variance. However, 16 out of 18 components are necessary to explain 95% of the total variance. This implies that our dataset is complex and that dropping only two features would cause an information loss while not significantly reducing the computational complexity of the machine learning algorithm. This finding is also consistent with the result of the correlation matrix, which shows that most features have some correlation, but none of them have a particularly strong correlation with the response variable.

3 Imputation Methods

During the data exploration, we noticed that all continuous attributes have a tail-heavy distribution with outliers. Some attributes have more extreme outliers than others. Outliers are usually to the top of the scale and only sometimes to the bottom of the scale. Consequently, there is a discrepancy between mean and median values for almost all continuous attributes. The three imputation methods that we decided to assess are the following: mean/median imputation, k-nearest neighbour imputation and iterative imputation.

3.1 Mean/Median Imputation

This imputation method is simple and easy to implement. It preserves the mean/median/mode of the observed data. It can be effective for small amounts of missing data when the missingness is completely random. However, it may ignore relationships between features. From data exploration, there are no strong correlations between features. The data may be biased if it contains outliers. The data file is small, so there will not be an issue about performing this on the dataset.

3.2 K-Nearest Neighbours (KNN) Imputation

This imputation method will consider the relationship between features by using the nearest neighbour. It can handle both numerical and categorical data. It remains robust to outliers and non-linear relationships. It will also provide a more accurate imputation than single imputation methods. Since our dataset is small, this should not be a problem.

3.3 Iterative Imputation

This method of imputation focuses on imputing the missing data in a multivariate manner. It will consider relationships among variables in the dataset. This method will impute missing values for all variables simultaneously; this conserves correlations and dependencies to improve accuracy. The iterative imputation approach involves iteratively imputing missing values for all variables using models that incorporate information from other variables in the dataset.

3.4 Results

We assessed all three of these imputation methods. Once the data had been imputed with each method, we plotted histograms of the distribution for numerical attributes and computed correlation matrices. Stage has one of the strongest correlations with status, so we looked at the distribution of stage using each imputation method and compared it with the original data.

3.5 Conclusions

Based on the analysis and assessment of the three imputation methods, the more favourable imputation method is KNN Imputation or Iterative Imputation. The imputed data closely matches the original data. For the mean/median imputation method, there are clear discrepancies between the imputed data. This can be explained from the data analysis.

We assessed all three of these imputation methods. Once the data was imputed with each method, we plotted histograms of the distribution for numerical attributes and computed correlation matrices. Stage has one of the strongest correlations with status. Therefore, we looked at the distribution of stage using each imputation method and compared it with the original data.

4 Dealing with Unbalanced Data

In addressing the prognosis prediction for patients with cirrhosis, we encountered a severe imbalance in the corresponding features. Therefore, we conducted comparative experiments with different sampling techniques to balance the response variables and explored the impact of various methods for handling unbalanced data on model performance. We applied three sampling methods to the data and subsequently fitted a gradient boosting decision tree classifier. Then, we compared ROC curves and other metrics.

Firstly, we implemented Synthetic Minority Over-Sampling Technique (SMOTE), to augment the number of samples in minority categories, thus optimizing the decision boundaries. After sampling, we observed a significant improvement in model performance, particularly in the prediction of minority classes, where recall and precision were both enhanced.

Then, we tried Random Under-Sampling and Neighborhood Cleaning Rule (NCL), two undersampling techniques [4]. Although these methods balance the class distribution by reducing the samples of the majority classes, they may lead to information loss and thus affect the model's performance in certain cases. Test results indicated that despite undersampling improving the prediction capability for minority classes, it also increased the risk of misclassification for majority classes.

Finally, we combined the undersampling and oversampling approaches. We combined the NCL and the SMOTE sampling methods. This approach maintains balance between classes while avoiding the problem of information loss due to excessive imbalance. However, in our experiments, this combined approach did not result in better classification performance compared to SMOTE alone.

5 Classification Models

Firstly, we randomly generated data with two numerical features and one categorical label with three classes, i.e. three clusters. We applied a range of classifiers to this data and visualised the decision boundaries to assess their performance. We found AdaBoost, Gradient Boosting Decision Tree, and XGBoost classifiers the most promising.

5.1 Selected Classifiers

AdaBoost enhances the model's focus on hard-to-classify samples by increasing the sample weights of those misclassified in the previous round, while GBDT improves model performance by reducing residuals in each iteration, progressively increasing classification precision. XGBoost, as an efficient version of GBDT, not only optimizes speed and performance but also reduces the likelihood of overfitting. Moreover, all three models support the requirements of multiclass classification, making them ideal choices for prognosis analysis.

5.2 Model Training

We fitted the three classifiers using the following features: N_Days, Bilirubin, Albumin, SGOT, Copper, Prothrombin, Stage, Ascites, Hepatomegaly, Spiders, and Edema. These are the 11 features

with the highest correlation with the Status label. Categorical columns were one-hot encoded. Stage was the only ordinal attribute. Numerical features were neither scaled nor normalised, as this is not required for the chosen classifiers. Each model was optimised using an exhaustive grid search and 5-fold cross-validation. Model performance was evaluated using multiple metrics such as log loss, accuracy, recall, precision, and an AUC score. We also plotted ROC curves for each label and computed confusion matrices. Log loss was used as the scoring metric during model fitting.

5.3 Model Evaluation

Figures 2-4 show the ROC curves achieved on unseen data. All three models could predict labels "C" and "D" reasonably well. The GBDT model performed slightly better that the AdaBoost model. As expected, the XGBoost model, with its optimized algorithms, performed the best the in terms of both speed and predictions. It achieved the highest accuracy, log loss, and AUC scores. Log loss scores are shown in Table 3.

These models were trained on 80% of the "train.csv" dataset and evaluated on the other 20%. For Kaggle submission, we retrained the XGBoost classifier with the best previously found parameters using all labelled data. This model achieved log loss of around 0.46 on both the Kaggle's public and private testing sets (see the Appendix).

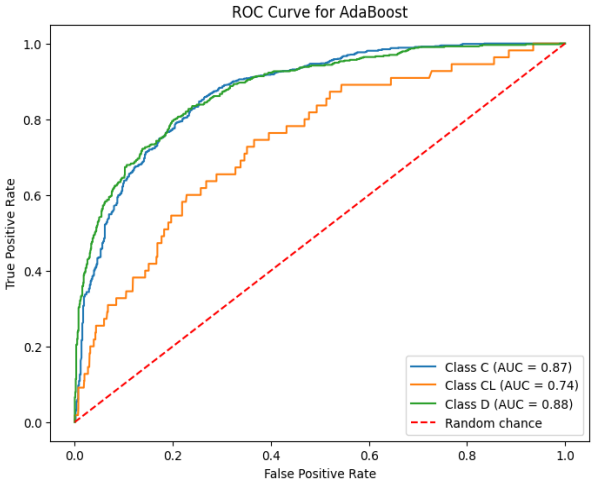


Fig. 2: AdaBoost ROC curves.

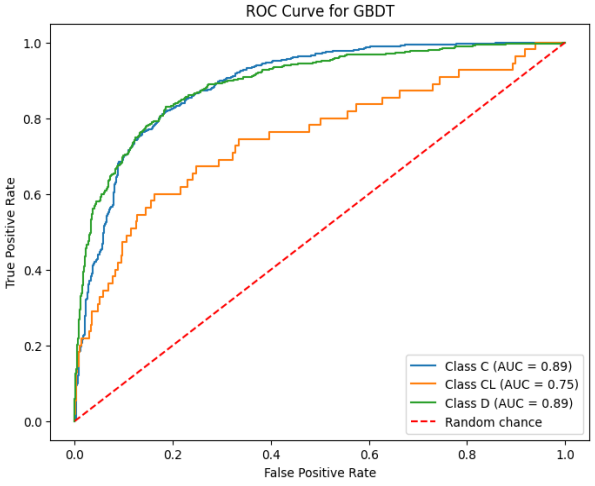


Fig. 3: GBDT ROC curves.

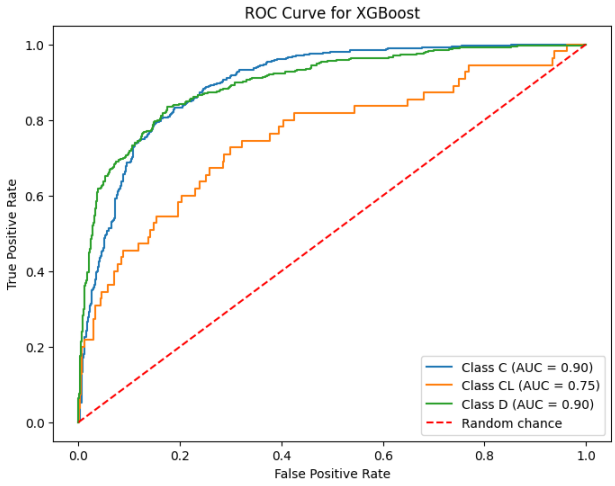


Fig. 4: XGBoost ROC curves.

6 Summary

We analysed 17 clinical features for predicting the survival state of patients with liver cirrhosis. We performed data exploration, assessed three imputation methods for missing data, tested three methods for dealing with unbalanced classes, and developed three classification models for the prediction of the patient's survival state.

The results obtained from data exploration show that there are some correlations between the clinical features in the dataset but that these relationships are not very strong. This is consistent with the findings from Principle Component Analysis, which shows that the dataset is relatively complex as most components are needed to explain 95% of the variance in our dataset.

Investigation methods for missing data, we found the K-Nearest Neighbour imputation method to be the most successful as it creates an imputed dataset most similar to the true data.

We found that oversampling methods work better for smaller datasets with unbalanced data. In particulat, the SMOTE oversampling method achieved good classification results.

Although we found that almost all features are required to explain most of the variance in the data, we managed to fit three solidly performing classifiers using only 11 attributes: N_Days, Bilirubin, Albumin, SGOT, Copper, Prothrombin, Stage, Ascites, Hepatomegaly, Spiders, and Edema. In particular, the XGBoost classifier achieved good accuracy, log loss, and AUC scores. However, all three models struggled with identifying the least represented label, even when an oversampling technology was utilised.

REFERENCES:

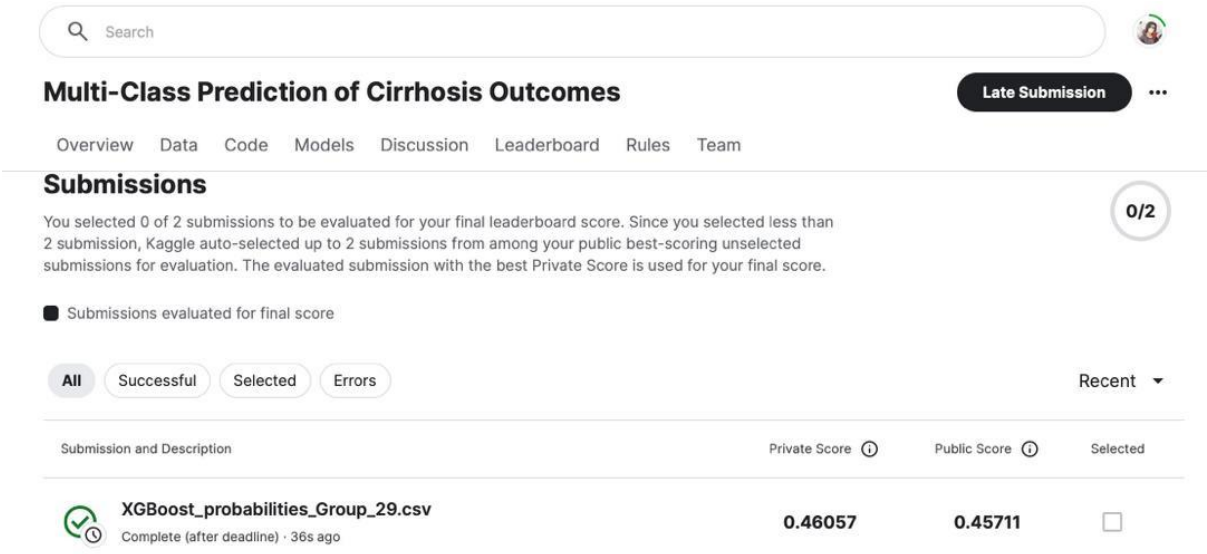
[1] NHS (2024). Cirrhosis. https://www.nhs.uk/conditions/cirrhosis/

[2] IHME. (2020). Prevalence of cirrhosis and other chronic liver diseases in the United Kingdom from 2000 to 2019, by gender (per 100,000 population) [Graph]. In Statista. Retrieved April 01, 2024, from https://www.statista.com/statistics/1036415/prevalence-of-cirrhosis-in-the-uk-by-gender/

[3] Wu, C., Yeh, W., Hsu, W., Islam, M., Nguyễn, P. A., Poly, T. N., Wang, Y. C., Yang, H., & Li, Y. (2019). Prediction of fatty liver disease using machine learning algorithms. Computer Methods and Programs in Biomedicine, 170, 23–29. https://doi.org/10.1016/j.cmpb.2018.12.032

[4] Junsomboon, N., & Phienthrakul, T. (2017). Combining Over-Sampling and Under-Sampling Techniques for Imbalance Dataset. In Proceedings of the 9th International Conference on Machine Learning and Computing (ICMLC '17) (pp. 243–247).

APPENDIX:


Search
Multi-Class Prediction of Cirrhosis Outcomes
Late Submission
Overview Data Code Models Discussion Leaderboard Rules Team
Submissions
You selected 0 of 2 submissions to be evaluated for your final leaderboard score. Since you selected less than 2 submission, Kaggle auto-selected up to 2 submissions from among your public best-scoring unselected submissions for evaluation. The evaluated submission with the best Private Score is used for your final score.
0/2
Submissions evaluated for final score
All Successful Selected Errors
Recent
Submission and Description
Private Score
Public Score
Selected
XGBoost_probabilities_Group_29.csv
Complete (after deadline) · 36s ago
0.46057
0.45711