Prediction for Lethality of hepatitis with classification model

Hongming Fu

Brown University, DSI

https://github.com/martinezphu07/data1030project_final.git

1. Introduction

1.1 Background

Hepatitis is a condition characterized by having an inflamed liver. There can be many causes for hepatitis, including viral infections, autoimmune diseases, and exposure to certain medications or toxins, while the most common cause of hepatitis worldwide is viral infection.

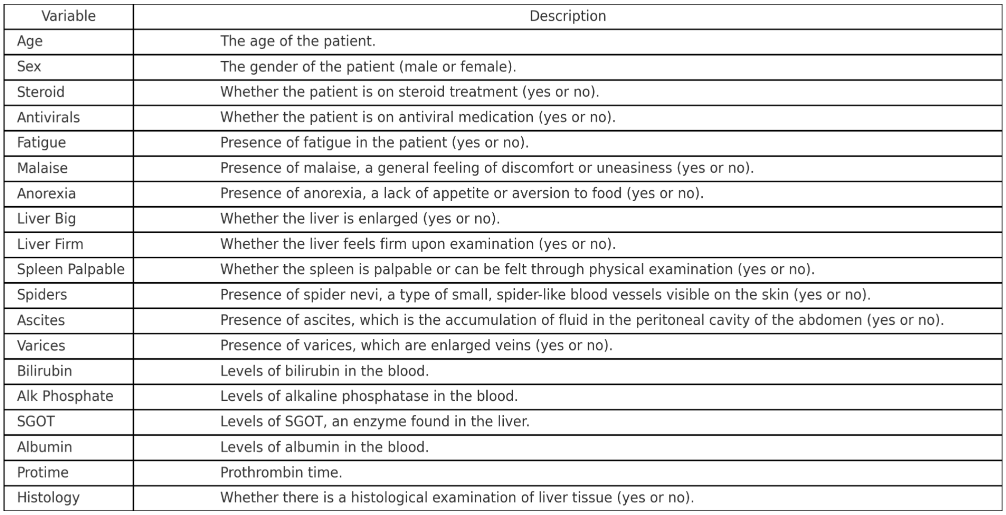
Prevention of hepatitis has been topic of discussion. Studies have found that an hundreds of death were found in the United States. The growth of hepatitis related deaths can be fast; nearly twice as many hepatitis A–related deaths were reported during 2016–2022 compared with 2009–2015. (Hofmeister MG, 2023)

1.2 Goal

This project aims to develop and test a machine learning pipeline that classifies if the hepatitis of a particular patient is a lethal one or not. The machine learning model will decide the lethality of one's hepatitis, send warnings to those whose prediction is lethal and suggest such a patient to seek for immediate medical care. This model, in proper circumstances, is designed to inform those who has no knowledge of their own condition and in turn decrease their deaths.

1.3 Data

The data involved in this project is selected from UCI Machine Learning Repository. (Hepatitis,1988)This dataset involves 155 instances of hepatitis patients and 19 features. There are 6 numerical features and 13 categorical features, while all categorical features are binary features. The dataset was originally gathered from Yugoslavia in 1988.Features and their description are described in Figure 1 below.



Variable	Description
Age	The age of the patient.
Sex	The gender of the patient (male or female).
Steroid	Whether the patient is on steroid treatment (yes or no).
Antivirals	Whether the patient is on antiviral medication (yes or no).
Fatigue	Presence of fatigue in the patient (yes or no).
Malaise	Presence of malaise, a general feeling of discomfort or uneasiness (yes or no).
Anorexia	Presence of anorexia, a lack of appetite or aversion to food (yes or no).
Liver Big	Whether the liver is enlarged (yes or no).
Liver Firm	Whether the liver feels firm upon examination (yes or no).
Spleen Palpable	Whether the spleen is palpable or can be felt through physical examination (yes or no).
Spiders	Presence of spider nevi, a type of small, spider-like blood vessels visible on the skin (yes or no).
Ascites	Presence of ascites, which is the accumulation of fluid in the peritoneal cavity of the abdomen (yes or no).
Varices	Presence of varices, which are enlarged veins (yes or no).
Bilirubin	Levels of bilirubin in the blood.
Alk Phosphate	Levels of alkaline phosphatase in the blood.
SGOT	Levels of SGOT, an enzyme found in the liver.
Albumin	Levels of albumin in the blood.
Protime	Prothrombin time.
Histology	Whether there is a histological examination of liver tissue (yes or no).

Figure 1, Description of columns in dataset

2. EDA

For the section of EDA, we started by putting together a correlation matrix that displays the correlation coefficient between all the numerical variables. From what we can observe from Figure 2, the correlation coefficient ranges from 0.42 to -0.38. This indicates that all features provide extra information for our machine learning model.

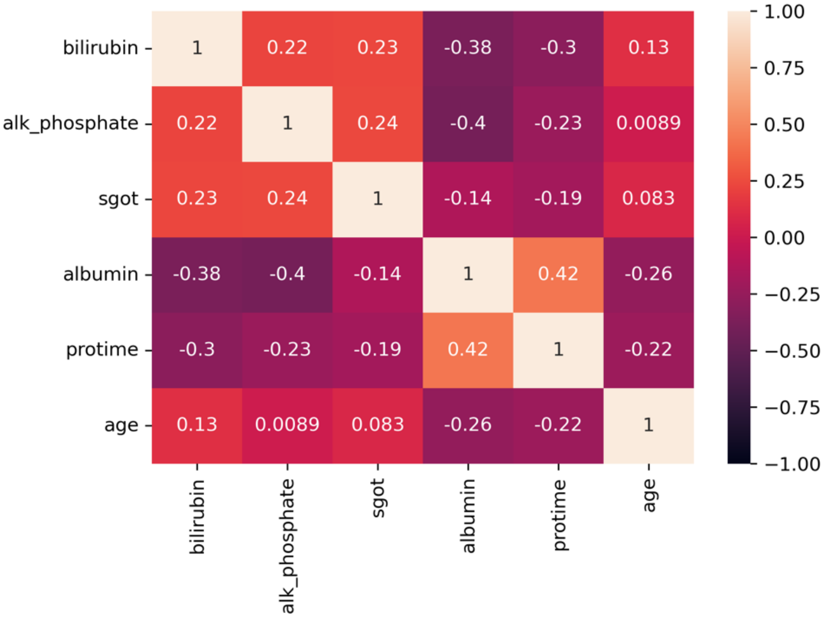


Figure 2, Correlation Confusion Matrix for all numerical Variables

We also generated the distribution of patients over age. From Figure 3.a, the blue shade distribution indicates the survival group, the group of patients with survived hepatitis. The orange shape distribution represents the terminal group, the group of patients that are diseased because of their hepatitis.

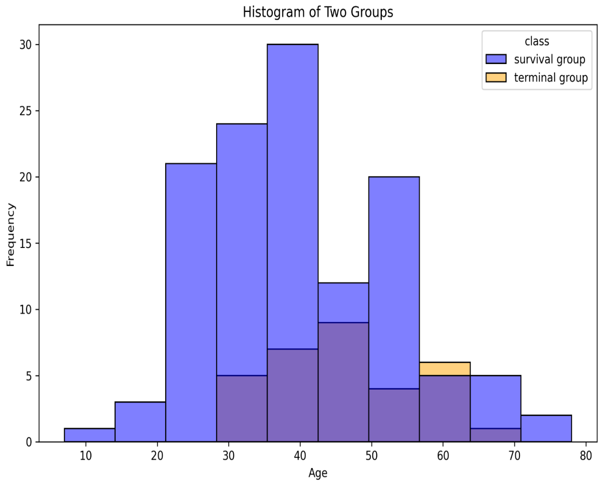


Figure 3.a, Histogram of Survival group and Terminal Group.

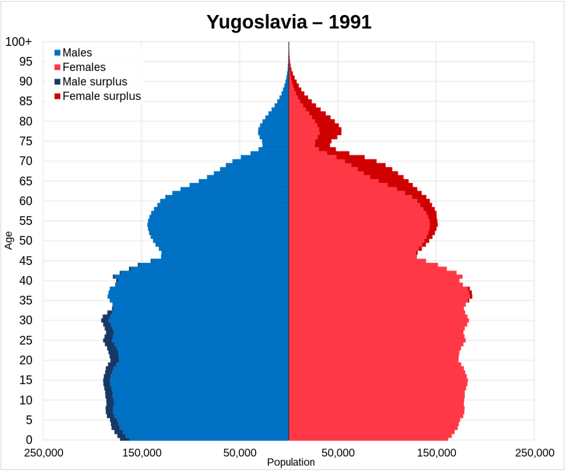


Figure 3.b Demographic structure of Yugoslavia

From Figure 3.b, we notice a shortage of population for people at the age group around 45. This shortage is due to the second world war. People in war did not incline to have children, while infants in war didn't intend to survive. A seemingly uncommon population distribution in the dataset is, in fact, a reflection of contemporary demographic structure, increasing the validity and credibility of this dataset.

Another EDA involved comparing the bilirubin level of survival group and terminal group, demonstrated in Figure 4. Bilirubin is a yellowish pigment that is made during the breakdown of red blood cells. Normally it will be excreted out of our body by our liver. Thus, a high bilirubin level implies a dysfunctional liver. The normal level of bilirubin is 0.3mg/L. From Figure 5, clearly both Survival Group and Terminal Group has an average bilirubin level significantly higher than the normal level, indicating that patients from both groups suffer from a malfunctioning liver. The mean for the terminal group is higher that of the survival group, hinting at a more significant liver issue. Medical domain knowledge can be reflected by our dataset, entrenched its validity.

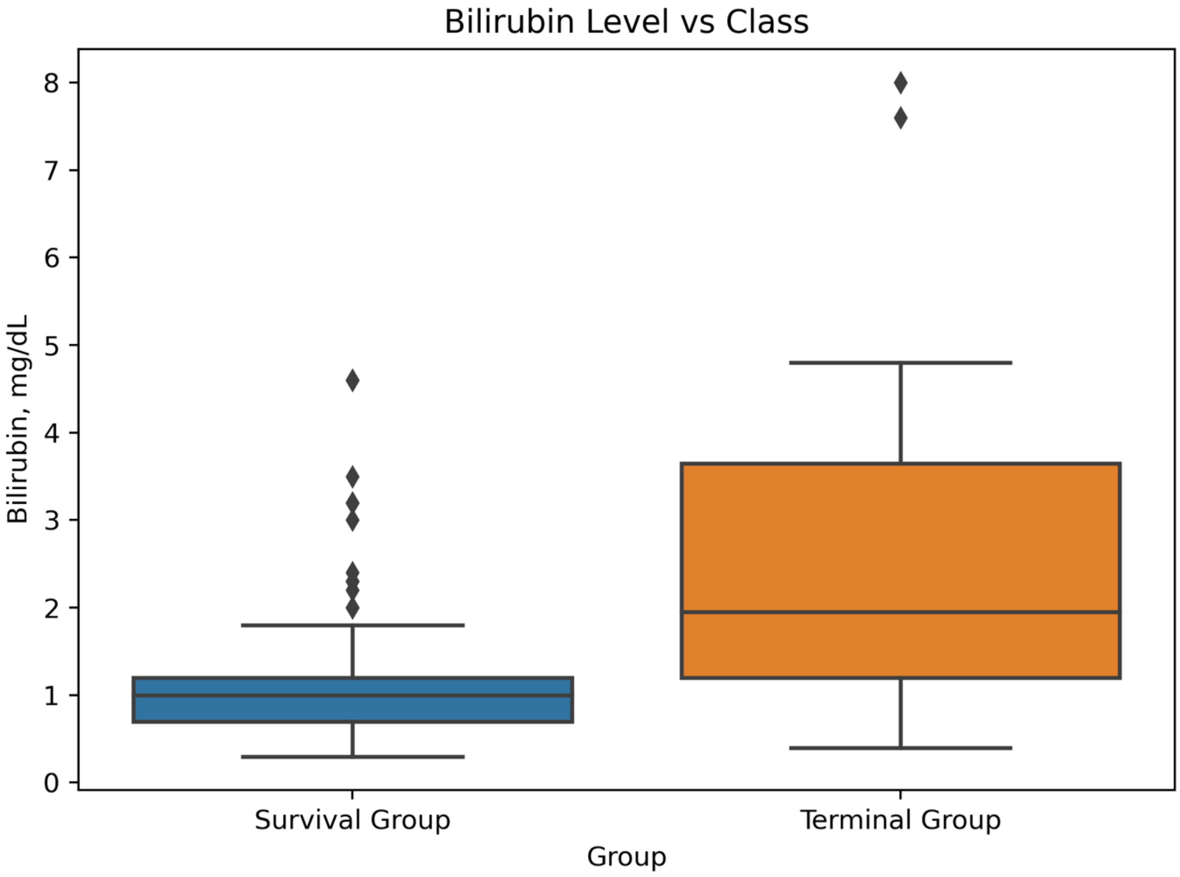


Figure 4, Bilirubin Level vs. Class

3. Method

3.1 Splitting

Since my dataset is fulfilled with individual patients and each of them are independent from each other, my dataset can be considered i.i.d. Thus, we have first extracted the test set with 20% as test size. In addition to that, my dataset can be considered imbalanced, since I have 20.6% of my datapoint belonging to the terminal group and 79.4% belonging to the survival group. Thus, we performed a stratified k-fold to ensure the proportion of survival and terminal group for each fold. The preceding process will be performed 10 times with 10 different random states, which will in turn provide me 10 sets of best models.

3.2 Preprocessing

We have used “OnehotEncoder” for 13 categorical variables, including "steroid", "antivirals", "fatigue", "malaise", "anorexia", "liver_big", "liver_firm", "spleen_palpable", "spiders", "ascites", "varices","histology". For 5 of the continuous variables, we have performed “Standardscaler” for them since they tend to have a sparse distribution, including "bilirubin", "alk_phosphate", "sgot", "albumin", "protime". For “age”, we have performed a minmax scaler since it contains a clear lower and upper bound.

3.3 Missing Values

This dataset contains missing values. Most of the medical features included have missing variables, but their missing proportions are usually no higher than 10%. This The two exceptions are “alk_phosphate” and “protime”; the former one missing about 15% and the latter one missing about 40% of them.

Considering that my dataset is composed of human data, especially medical test results, it will be inappropriate and ethically questionable to use imputation of any kind to solve the problem of missing values. Also, dealing with missing data might be easy for models like XgBoost that innately accepts missing value as proper input, but not for many models who don’t. Therefore, we have used the reduced-features model. The reduced-features model can use data points that have the same missing pattern in the train and validation set. As a result, we must train as many models as the number of distinct patterns in the test set, putting on a greater amount of computational burden.

3.4 ML Algos

In this project, 5 machine learning algorithms are used, including Logistic Regression Model, K nearest neighbor classifier, Support Vector Classifier, Random Forest Classifier and Xgboost Model.

For Logistic Regression Model, the parameter grid we have used is :

```
{'logisticregression__C': [ .1, 1, 10, 100],
'logisticregression__penalty': ['l1', 'l2', 'elasticnet'],
'logisticregression__solver': ['newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga']}
```

For K Nearest Neighbor Classifer:

```
{
 'kneighborsclassifier__n_neighbors': [40,50,60],
'kneighborsclassifier__metric': ['euclidean', 'manhattan', 'minkowski'],
'kneighborsclassifier__weights': ['uniform', 'distance'],
'kneighborsclassifier__leaf_size': [10, 20]
}
```

For SVC :

```
{
   'svc__gamma': [ 1e1, 1e3, 1e5],
   'svc__C': [1e-1, 1e0, 1e1]
}
```

For Random Forest Classifier:

```
{
   'randomforestclassifier__n_estimators': [3, 5,10],
   'randomforestclassifier__max_depth': [2, 3,5, 10]
}
```

For XgBoost Model:

```
{
   'xgbclassifier__learning_rate': [0.01, 0.1, 0.2, 0.3],  # or 'eta'
   'xgbclassifier__max_depth': [3, 5, 7],
   'xgbclassifier__min_child_weight': [1, 2, 3, 4],
   'xgbclassifier__colsample_bytree': [0.5, 0.7, 0.9, 1.0]
}
```

3.5 Pipeline

For the machine learning pipeline, we have constructed 5 pipelines, each one of them corresponding to a specific algorithm we involved. Since it is a classification problem, I used ‘accuracy’ and ‘f1_score’ as metrics for this machine project, since accuracy, precision and recall are all significant metrics for a classification problem, and ‘f1_score’, the harmonic mean of precision and recall, combines these two metrics. We looped through 10 different random states to minimize the effect of randomness. After the loops are completed, we will have lists that contain accuracy score and f1 score for each loop. The mean and standard deviation of each list will be calculated as a measurement of performance for this model.

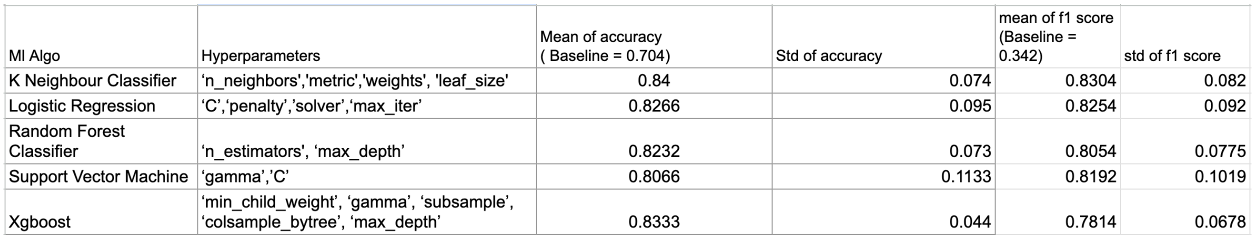
3.6 Metrics

The baseline test score for accuracy is the proportion of the most frequent element in the test set, which is 70.8%. The baseline for f1_score is a little more complex since if we make all predictions to be the most frequent element, precision p will be ill-defined since p will be 0/0. Thus, the baseline for f1_score will be predicting all input to be the second most frequent element, which will be 34.2% in this scenario.

4. Result

4.1 Overview

An overview of the machine learning algorithms and their results are listed in Table 1.



Ml Algo	Hyperparameters	Mean of accuracy (Baseline = 0.704)	Std of accuracy	mean of f1 score (Baseline = 0.342)	std of f1 score
K Neighbour Classifier	'n_neighbors','metric','weights', 'leaf_size'	0.84	0.074	0.8304	0.082
Logistic Regression	‘C’,‘penalty’,’solver’,‘max_iter’	0.8266	0.095	0.8254	0.092
Random Forest Classifier	‘n_estimators’, ‘max_depth’	0.8232	0.073	0.8054	0.0775
Support Vector Machine	‘gamma’,’C’	0.8066	0.1133	0.8192	0.1019
Xgboost	‘min_child_weight’, ‘gamma’, ‘subsample’, ‘colsample_bytree’, ‘max_depth’	0.8333	0.044	0.7814	0.0678

Table 1. The ML Algos, Hyperparameters and results from training

After getting the result of the all-machine learning models, the mean and standard deviation of accuracy and f1 score can also be calculated. We can observe that all my machine learning models outperform the baseline in both accuracy and f1_score. However, For the remainder of this report, we will use Xgboost model for further analysis. Although it comes with the lowest f1_score, its f1_score is not significantly lower than others, and it is the most most stable model in term of accuracy; it is 3 std above the baseline, while other models are at best 2 std above.

4.2 Interpretability

4.2.1 Global

We have taken three approaches to evaluate the importance of each feature in our XgBoost model.

The first approach we have taken is permutation feature importance,depicted in Figure 5. It involves randomly reshuffling one feature and measuring how much the predictive power, accuracy in this case, decrease. By the figure below, we can clearly observe that “alk_phosphate”, “fatigue”, “spleen_palpable” and “onehot_spiders” are among the top significant features.

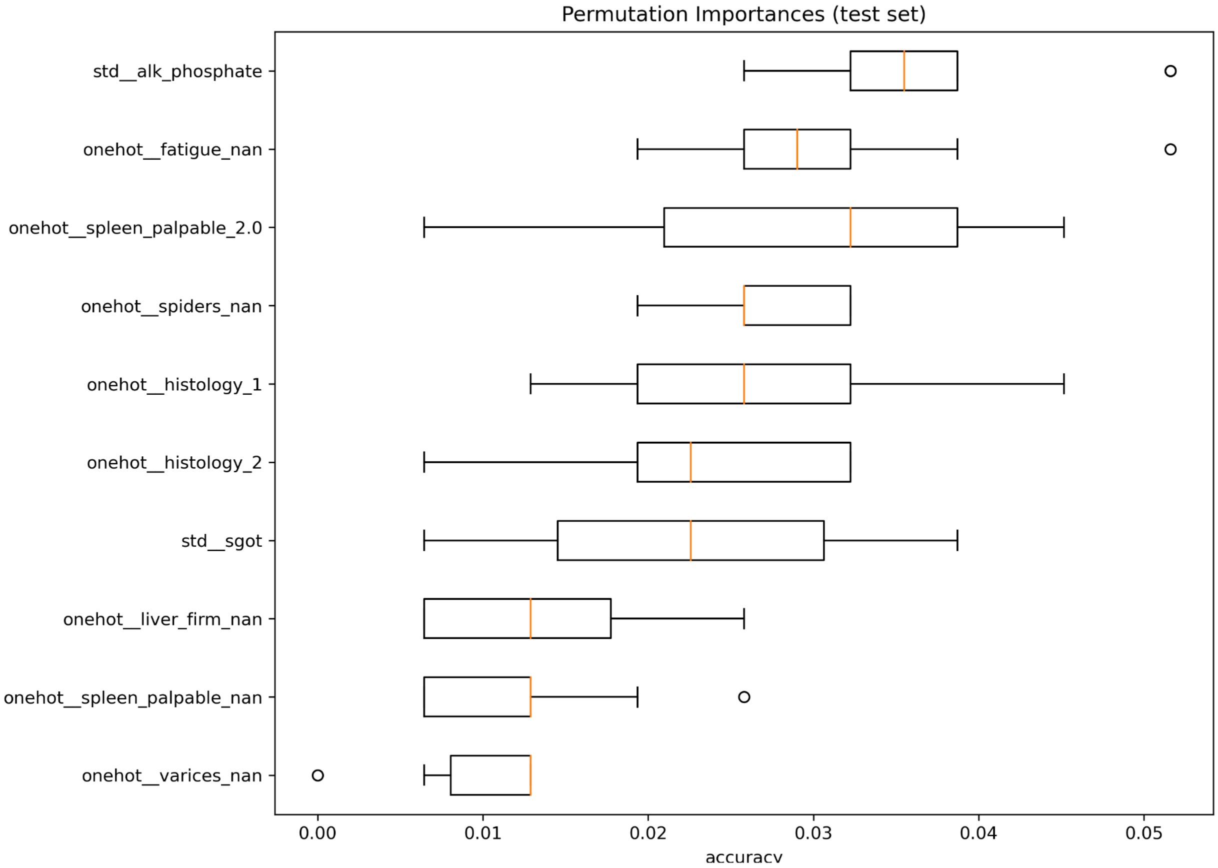


Figure 5, Permutation Importance of each features

The second approach involves 5 XGB metrics that come with our XgBoost model, depicted from Figure 6.a to Figure 6.e. There are five metrics provided by XgBoost, including "gain", "weight", "cover", "total_gain" and "total_cover". Each of these metrics measures the features from different perspectives, and the five figures below depicts their results.

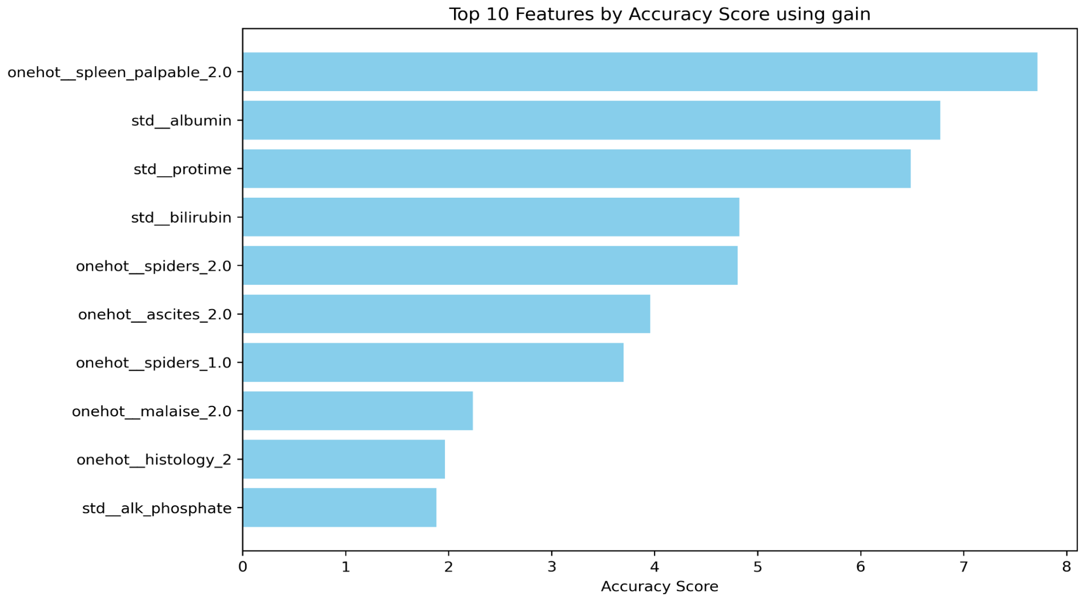
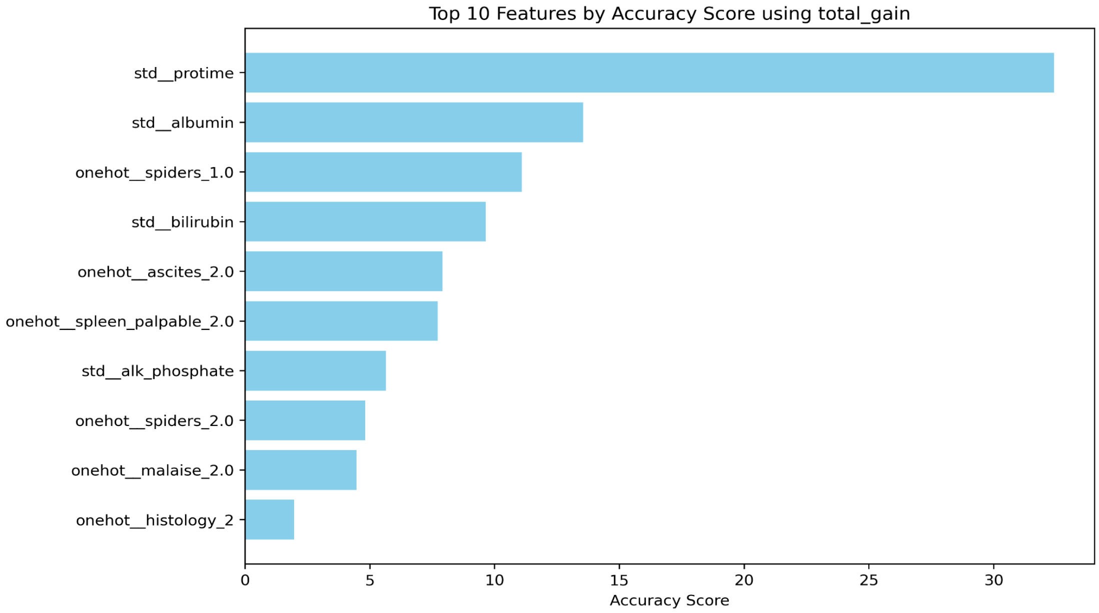


Figure 6.a ,top 10 features by accuracy score using gain



6.b, top 10 features by accuracy score using total_gain

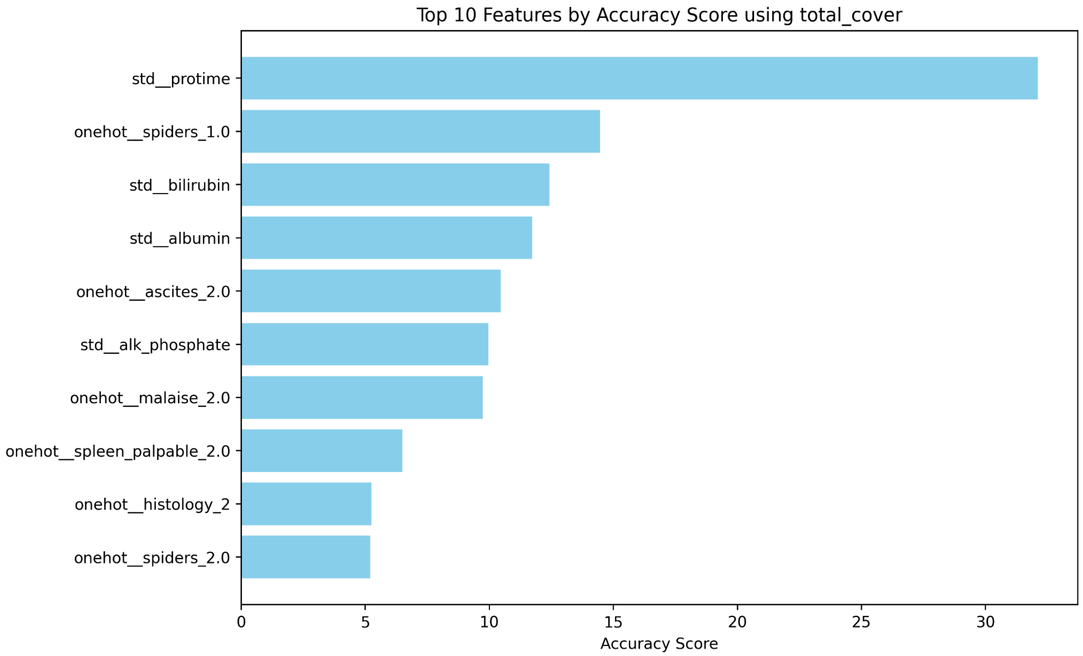


Figure 6.c, top 10 features by accuracy score using total_cover

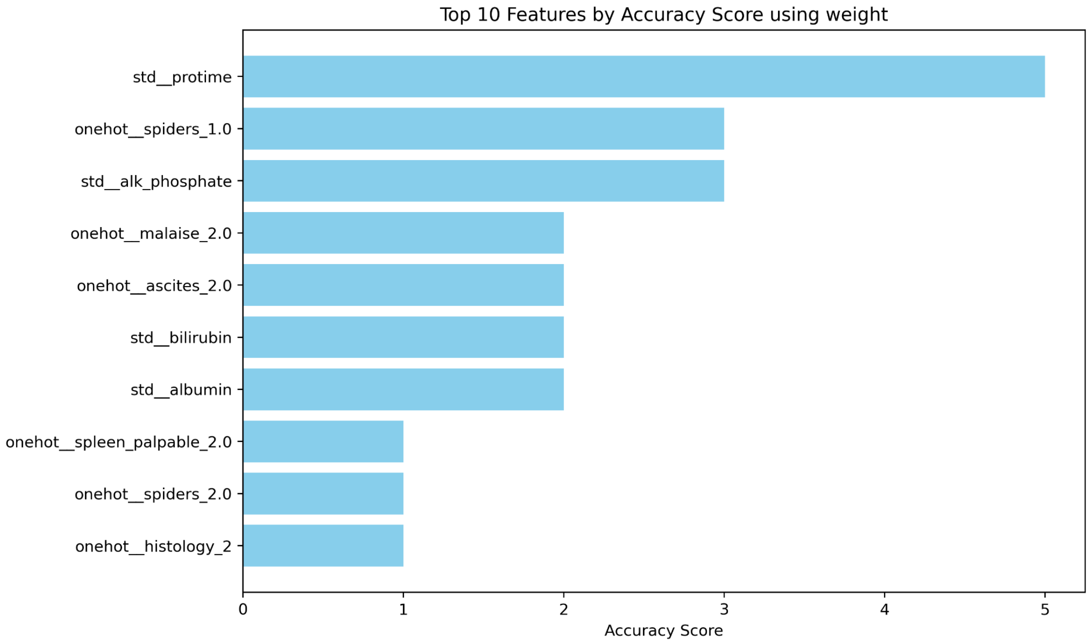


Figure 6.d, top 10 features by accuracy score using weight

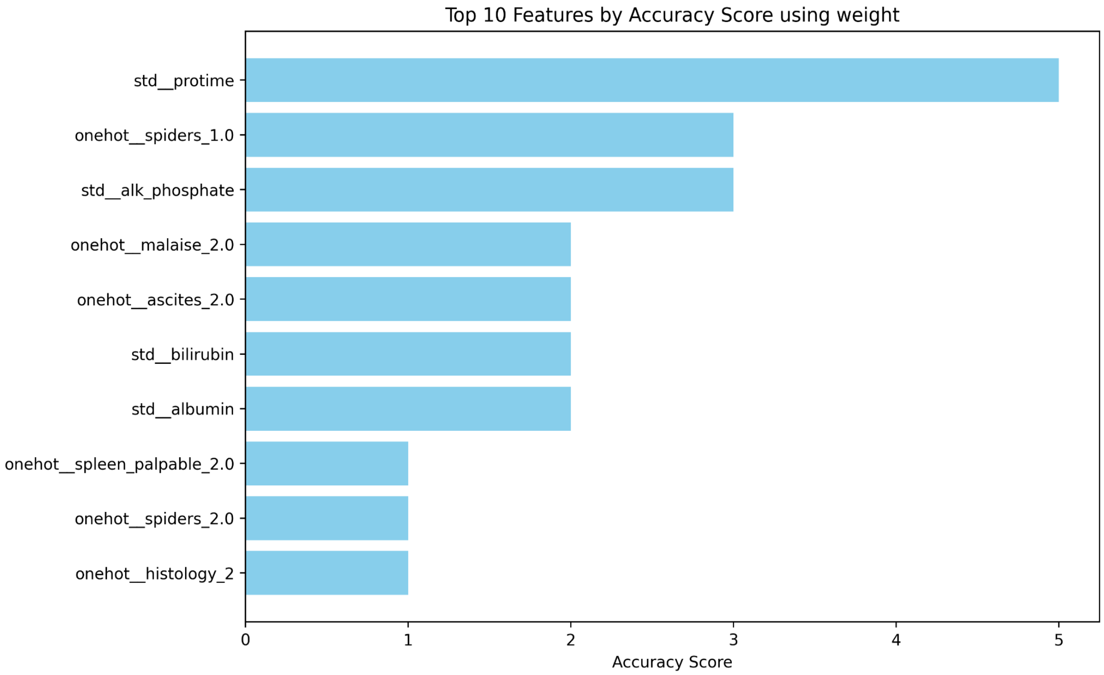


Figure 6.e, top 10 features by accuracy score using weight

It can be easily observed that “std_protime”, “onehot_spiders_1.0”, “alk_phosphate” and “bilirubin” are usually among the top 5 significant feature using any of these five metrics.

The third approach involves calculating the Shap value. More specifically, it involves calculating the average impact on model output magnitude with game theory as basis. Demonstrated in Figure 7, we can be observed that the top 5 features are “std_protime”,”onehot_spiders_1.0”, “std_albumin”, “std_bilirubin” and “onehot_malaise_2.0” .

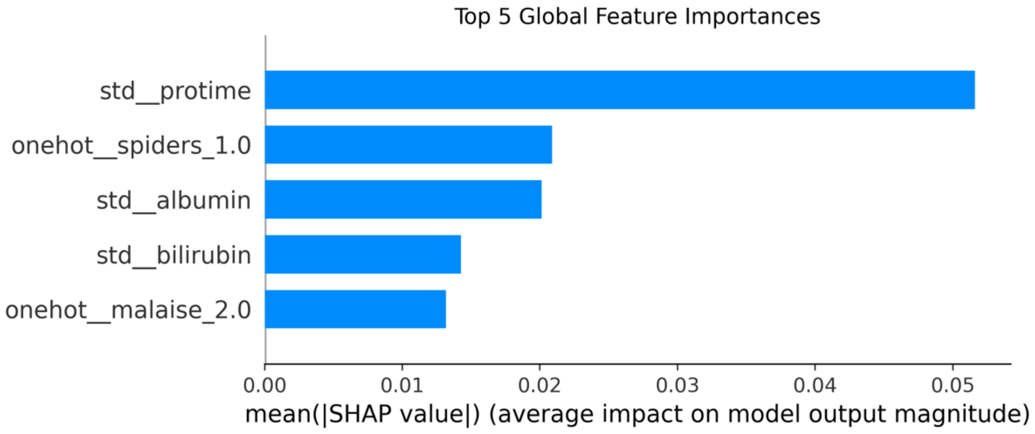


Figure 7, top 5 global features by absolute shap value

The least significant features, depicted by Figure 8 using permutation importance, are "onehot_spleen_palpable_1.0", "onehot_spiders_1.0", "onehot_spiders_2.0", "onehot_ascites_1.0" and "onehot_ascites_2.0". These features barely shows up in the most significant features.

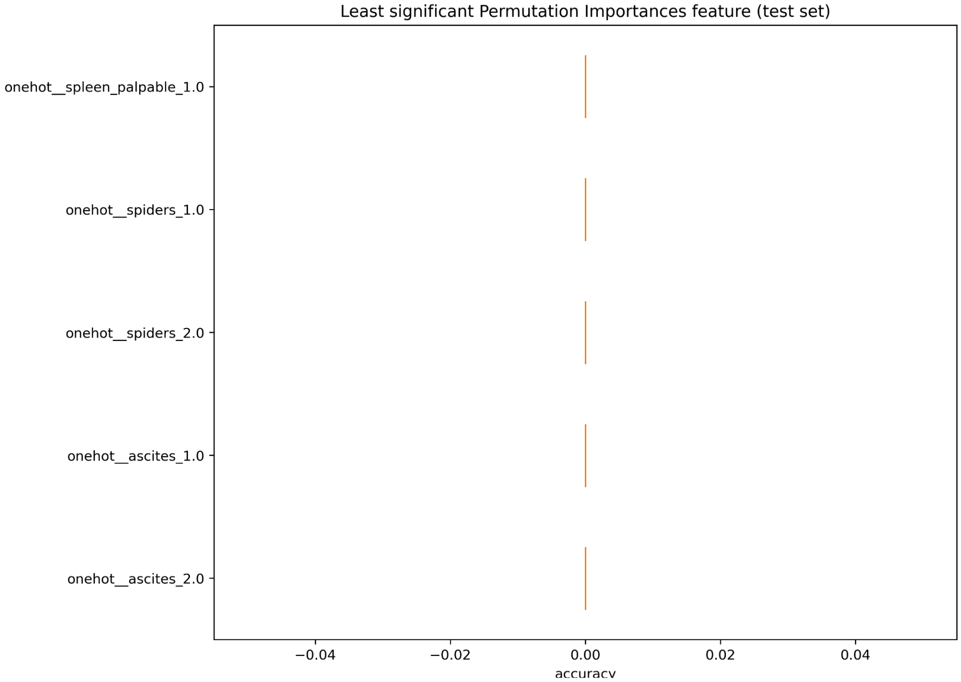


Figure 8, Least significant features by permutation importance

4.2.2 Local

For local importance, we have calculated the shap value for a particular datapoint in the test set. We have constructed a force plot, using probability as reference to show how each feature affect the probability of some datapoint to be predicted as lethal or non-lethal.

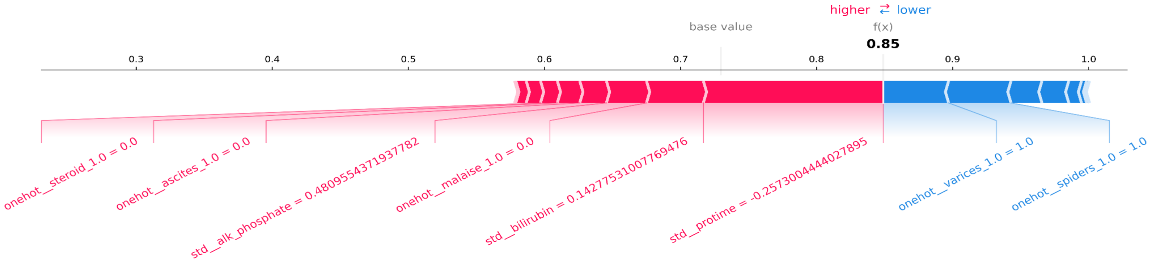


Figure 9.a, Force plot of a likely non-lethal hepatitis patient

In Figure 9.a, force plot gives an example of a patient that has a 0.85 chance of having non-lethal hepatitis. We can see that "std_protime", "std_bilirubin" and "onehot_malaise" all pushes the probability of this patient having non-lethal disease higher, while "onehot_varices" and "onehot_spiders" lowers the probability.

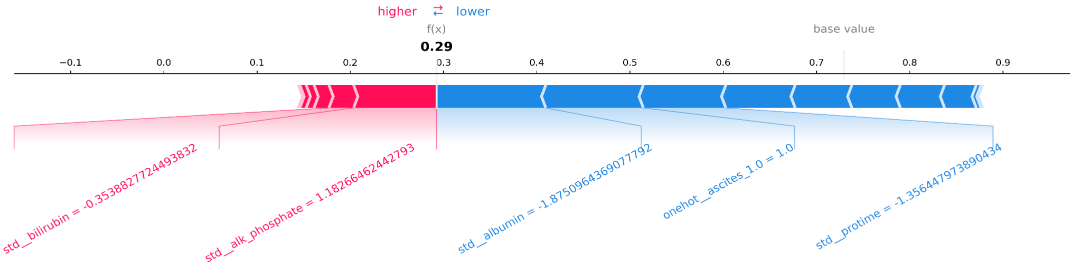


Figure 9.b, Force plot of a likely lethal hepatitis patient

The other force plot, depicted in 9.b, is an example of a datapoint that is most likely having lethal hepatitis. The patient has a 29% chance of having non-lethal hepatitis. We can observe that "std_albumin", "onehot_ascities_1.0" and "std_protime" have significantly lower the probability of this patient to have non-lethal hepatitis to 29%.

5. Outlook

One aspect that can be improved in further studies is the metric we used training the models. In this project, we have taken accuracy as the main metrics when training our machine learning models and tuning hyperparameters. However, such metrics might lead to an even distribution between false non-lethal and false lethal cases. In practice, false lethal can be accepted since it involves sending someone who is not in an urgent need of medical care to the hospital. On the other hand, false non-lethal is far less acceptable, since it essentially asks someone who is in a life-threatening situation and in urgent medical care not to seek assistance, which can be extremely dangerous and very likely to cost lives. As a result, we can consider using f1 score, or f-beta score with beta other than 1 to retrain the models and hyperparameter, for ethical considerations.

Reference:

1. Hofmeister MG, Gupta N. Preventable Deaths During Widespread Community Hepatitis A Outbreaks — United States, 2016–2022. MMWR Morb Mortal Wkly Rep 2023;72:1128–1133. DOI: http://dx.doi.org/10.15585/mmwr.mm7242a1

2. Hepatitis. (1988). UCI Machine Learning Repository. https://doi.org/10.24432/C5Q59J