ML Contest on Liver Disease Prediction

By :- Birla Institute Of Technology (BIT), Mesra, Rachi

StillLearning

Team Members

Name	Email
Jatin Goyal	goyaljatin1102@gmail.com
Aman Kumar Raj	ar837232342@gmail.com

Problem Statement

In this Machine Learning contest, we have to create predictive models to predict the stage of liver Cirrhosis using 18 clinical features as Cirrhosis damages the liver from a variety of causes leading to scarring and liver failure.

Motivation Behind Problem Statement

Hepatitis and chronic alcohol abuse are frequent causes of the disease. Liver damage caused by cirrhosis can't be undone, but further damage can be limited. Treatments focus on the underlying cause. In advanced cases, a liver transplant may be required. Predicting the stage of cirrhosis and beginning the treatment before it's too late can prevent the fatal consequences of the disease.

Notebooks

- **EDA on Dataset:-** https://www.kaggle.com/raj401/liver-disease-eda/notebook

- **Create Folds:-** https://www.kaggle.com/raj401/liverdisease-create-folds/notebook

- **Our Complete Model:-**

  https://www.kaggle.com/raj401/liver-disease-complete-ensemble-2nd-place/notebook

Statistical Analysis

We went through many journals and articles to understand the problem of Liver Disease and what causes them. Few of our key findings are as follows:-

- ❑ ALP + GGT is used to discriminate between Liver and Bone disease. Both are higher in liver disease, while only ALP is higher in bone disease(This can be a very important feature).
- ❑ AST- ALT ratio or SGOT-SGPT ratio are good indicators of Liver disease.
- ❑ In practicality abnormal quantity of any of the chemical in Liver Function Test don't guarantee presence of Liver disease. We also need to consider patients complete medical history before coming to any conclusion.

After understanding the problem statement we did EDA of our dataset. Here we explored the dataset and tried to figure out important features and relationships.

Link of EDA notebook :- https://www.kaggle.com/raj401/liver-disease-eda/notebook

1.

Data Preprocessing

❑ **Categorical features :-**
'Status', 'Drug', 'Sex', 'Ascites', 'Hepatomegaly', 'Spiders', 'Edema'

❑ **Numerical features :-**
'N_Days','Age','Bilirubin','Albumin','Copper','Alk_Phos','SGOT','Platelets','Prothrombin',’Cholesterol’,’Triglicerieds’

❑ In our dataset there are many NaN values, so we removed them first.

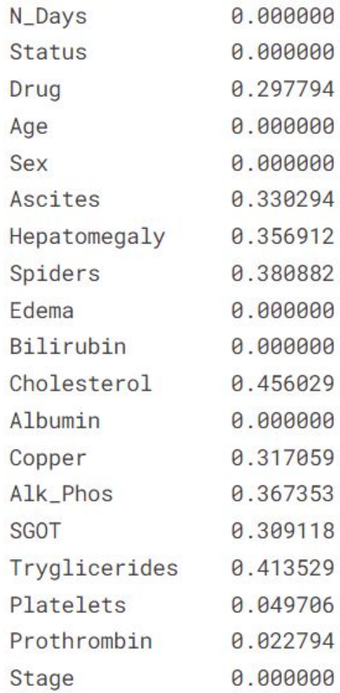
Dealing with nan value

❑ We calculated total % of nan value in each column.(right figure)

❑ We dropped ‘**Cholesterol**’ and ‘**Triglicerides**’ because they have large no of nan values.

❑ Then we looked at the distribution of all the features and found out that only ‘Alubmin’ and ‘Platelets’ are normally distributed while rest are **skewed** towards either right or left. So we filled ‘Albumin’ and ‘Platelets’ with the mean while for rest we filled with mode. (since mean is susceptible to outliers)

Note:- To avoid leakage we filled Test set columns with **mean** and **mode** of corresponding columns of Train set.



```
N_Days          0.000000
Status          0.000000
Drug            0.297794
Age             0.000000
Sex             0.000000
Ascites         0.330294
Hepatomegaly    0.356912
Spiders         0.380882
Edema           0.000000
Bilirubin       0.000000
Cholesterol     0.456029
Albumin         0.000000
Copper          0.317059
Alk_Phos        0.367353
SGOT            0.309118
Tryglicerides   0.413529
Platelets       0.049706
Prothrombin     0.022794
Stage           0.000000
```

Dealing with Categorical Variables

Categorical Variables :-

❑ We converted categorical variables like ‘Sex’, ‘Copper’ etc to one hot vector.

❑ For this we first merged train and test set. We did this to maintain same classes in train and test set.

For example :-

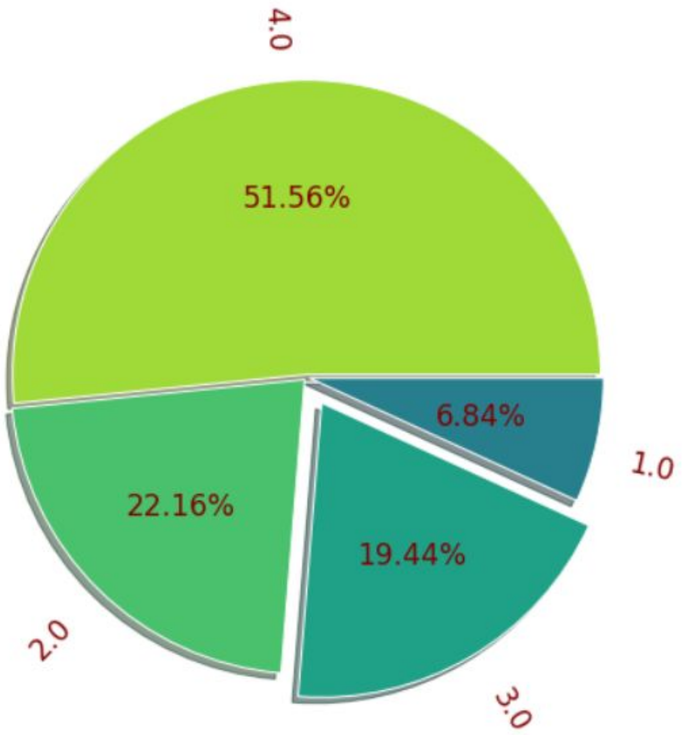
Train col1 :-- A,B,A,A,A

Test col1 :-- C,C,A

Now, if Train-Test kept separate then their one hot vectors with correspond to different things.

Target Column:-

- ❑ Almost 51% of sample belonged to Stage 4, 22% to Stage 2, 19% to Stage 3 and 7% to Stage 1.
- ❑ So there is a condition of **class imbalance** which we have to deal separately.
- ❑ We also plotted scatter plot of target variable with other variables to see if any feature has linear relationship with it. But there was no significant linear relationship.



Feature Engineering

- ❑ We also notice that there are certain ranges in which presence of a substance is considered normal and out of which it is considered abnormal. So we created new features based on this idea.
  The normal ranges are as follows:-
  - ❑ **Biluribin** 1-20 umol/L
  - ❑ **Albumin** 32-44 g/L
  - ❑ **Alk_Phos** 35-140 U/L
  - ❑ **SGOT** 5-30 U/L
- ❑ So for each of the above feature we created new feature as **normal_Biluribin**, **normal_Albumin** etc and assigned value of 1 if it was in the range else 0. (We also required to do unit conversion first)
- ❑ **Binning**: We created new feature **Age_bin** using Age column. (Binning converts numerical variable into categorical variable).
- ❑ The idea behind binning was that often doctors/practitioners divide patients in age groups. Like 4-10,11-17,18-29,30-50 and 50 above. They also give medicines and treatments based on patients age group.
- ❑ At the end we picked important features using LGBMClassifier.

Algorithm & ML Approach

Link of Notebook containing Complete Approach:-
https://www.kaggle.com/raj401/liver-disease-complete-ensemble-2nd-place/notebook

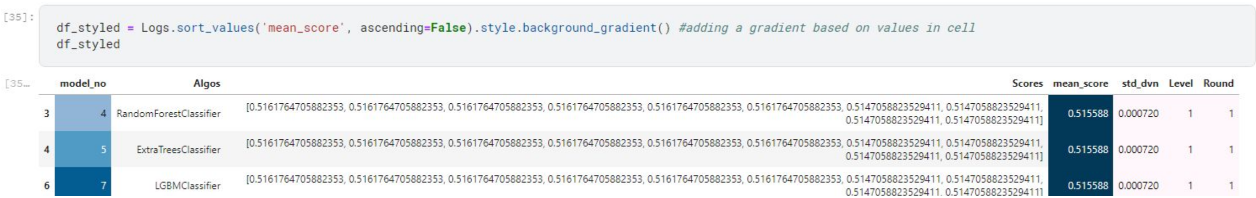
Ensemble Techniques

- ❑ First we started with various single models like catboost, Neural network and we were able to reach score of 25 pts. But after that it was not improving further, So we planned to use Ensemble of several models.
- ❑ Since we didn’t want to overfit to public leaderboard so we used cross validation method.
- ❑ Our aim was to maintain very strong local CV.
- ❑ We created 10 folds of our dataset using **StratifiedKFold** because our target is imbalance (Stratified divides data such that each group gets target classes in equal proportion).

Create Folds :- https://www.kaggle.com/raj401/liverdisease-create-folds/notebook

Model Architecture

- We created 2 levels for building models. In each level we trained various models and made TEST predictions and OOF predictions. (this gave us score of 50 pts)
  - In level 1 we had 3 Rounds each round 8 models , so total 24 models in level 1.
  - In level 2 we had 2 Rounds, 8 model in each round, so total 16 models.
- Top **most stable** models of level 1 were used to train meta models of level 2. (ex below)
- Then finally we used Random Forest Classifier on top of level 2 to make our final prediction. (this gave our final score of 61.13 - 60.88 on public and private lb.
- Each level trains on previous level and thus further improves the performance.



[35]:

```
df_styled = Logs.sort_values('mean_score', ascending=False).style.background_gradient() #adding a gradient based on values in cell
df_styled
```

[35…

	model_no	Algos	Scores	mean_score	std_dvn	Level	Round
3	4	RandomForestClassifier	[0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5147058823529411, 0.5147058823529411, 0.5147058823529411, 0.5147058823529411]	0.515588	0.000720	1	1
4	5	ExtraTreesClassifier	[0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5147058823529411, 0.5147058823529411, 0.5147058823529411, 0.5147058823529411]	0.515588	0.000720	1	1
6	7	LGBMClassifier	[0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5161764705882353, 0.5147058823529411, 0.5147058823529411, 0.5147058823529411, 0.5147058823529411]	0.515588	0.000720	1	1

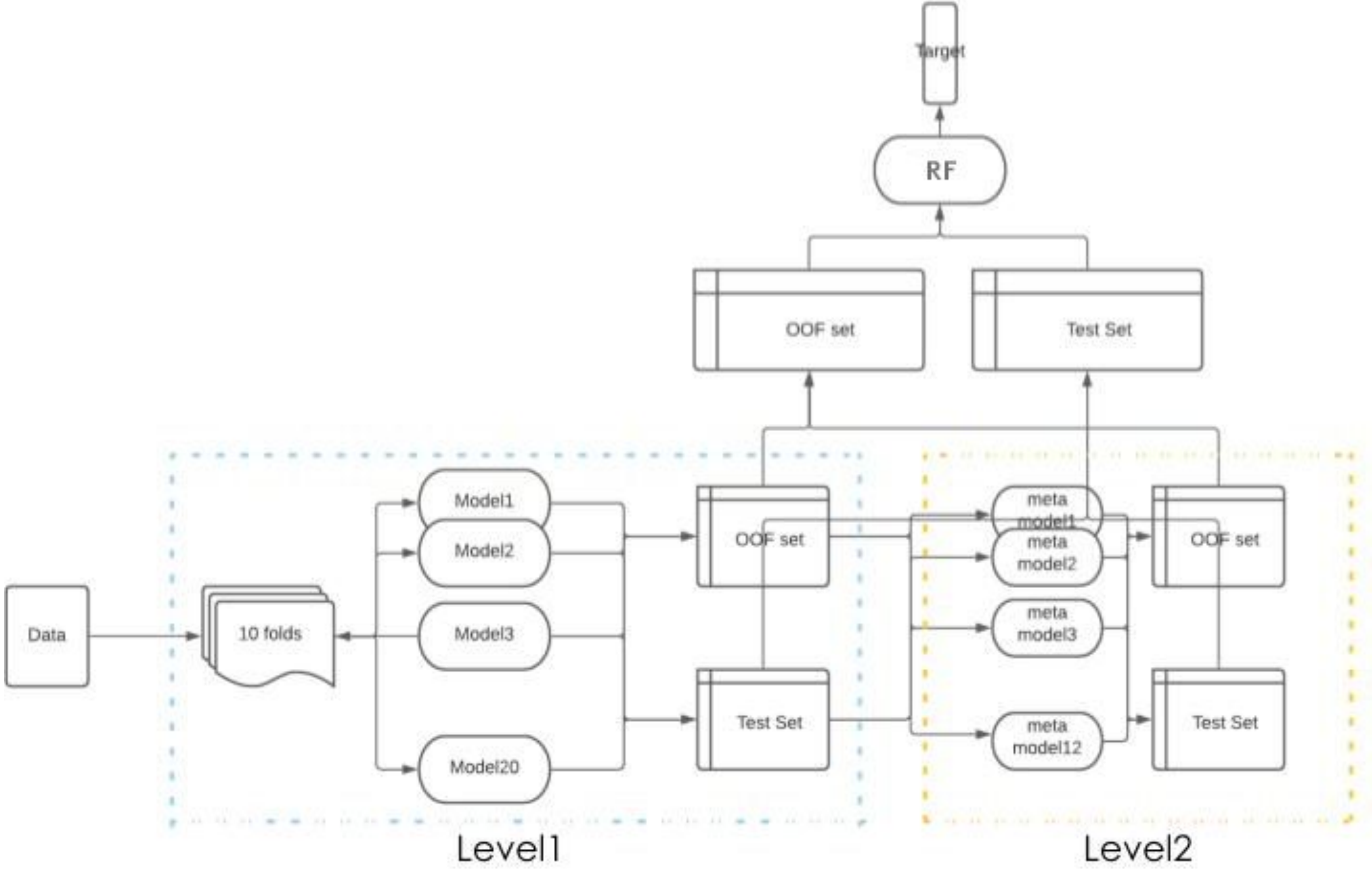


Fig:- Architecture of our Ensemble model.

Model Selection and hyperparameter Optimization

- ❑ Each of the model of level 1 and level 2 were optimized using Optuna.
- ❑ We ran Optuna for 50 trial. Optuna is a bayesian optimization algorithm. It is more efficient than GridSearch and RandomizedSearch in most of the cases because Optuna uses sampling algorithm to select next hyperparameter.

Evaluation matrix

- ➢ As this is multiclass classification problem so, we used Weighted-F1 as the evaluation metrics.
- ➢ F1 is the harmonic mean of precision and recall.
- ➢ We calculated the Weighted F1 because we have situation of target imbalance.

What worked and what didn’t work

Worked

- **Binning** Age column
- Creating new feature based on whether the chemicals present in blood are in **normal range** or not.
- **Optuna** to optimize Hyperparameter
- Using **StratifiedKFold** helped deal with target imbalance.
- **10 Folds** helped create strong local CV.

Didn’t work

- Using **single model** didn’t improve score much.
- Using **PCA** to select most important features didn’t help.
- Using **PolynomialFeatures** to generate polynomial and interaction features didn’t help.

Real World Outcomes

- It can help doctors in saving their time by predicting the stage of disease for patients.
- It can also help patients by predicting liver Cirrhosis at early stage so that they can start their treatment before it is too late as this disease can cause fatal consequences.
- It can also help in giving personalized prescription to every patients based on their clinical features.
- Today almost everybody above the age of 12 years has smartphones with them, and so we can incorporate these solutions into a mobile or web device which will be highly beneficial for a large section of society.

References

1. https://www.sciencedirect.com/topics/biochemistry-genetics-and-molecular-biology/ast-alt-ratio
2. https://www.mayoclinic.org/diseases-conditions/cirrhosis/symptoms-causes/syc-20351487
3. https://acadpubl.eu/jsi/2018-118-7-9/articles/9/72.pdf
4. https://arxiv.org/abs/1907.10902

THANK YOU