Predicting Cirrhosis Patient Survival

Tyler Baker and Dominic Pennacchio
Group #: 6
Department of Computer Science and Engineering
University at Buffalo
Buffalo, NY 142603
{tjbaker; dmpennac}@buffalo.edu

Abstract

Using the Cirrhosis Patient Survival Prediction dataset available at the UC Irvine Machine Learning Repository, several models were trained on the data, which includes information such as triglyceride levels and cholesterol levels, among other variables/features. The models trained were: Support Vector Machine (with different kernels), Neural Network, and XGB Classifier. Then accuracy scores were used to evaluate the performance of the SVM models and the XGB classifier and tracked the accuracy over epochs to evaluate the Neural Network.

1 Introduction

The problem that this project aims to solve is to effectively detect the likelihood of survival of Cirrhosis patients based on their symptoms and treatment methods. This is relevant to society since machine learning can recognize patterns in data that humans may miss, which would catch early warning signs of imminent death that otherwise could go unnoticed by medical professionals. This project is attempting to find methods of approaching this problem that may have been overlooked by others in the field that could have better predictive accuracy than previous models created.

2 Related works

Others in the field have attempted to use Support Vector machines and XGB Classifiers to determine the survival state of the patients in the dataset Jaxongir et al. [2024], mlmosaic [2024]. Our work is mostly an extension by building off of the original implementations provided by the referenced sources. We then compare the different methodologies in an attempt to find the best model for our use case. In the "Prognosis in primary biliary cirrhosis: model for decision making" Dickson et al. [1989] paper, the authors used a method called the "Cox regression method" which is a statistical method used for survival analysis and doesn't use any form of artificial intelligence, although it does use the same dataset we are using for our models. Another similar article was "Predicting Progression of Type 2 Diabetes Using Primary Care Data with the Help of Machine Learning" Ozturk et al. [2023], which although didn't use the same data set, used a similar methodology (an SVM and they also used an RF model). They concluded, "that predicting the risk of developing hypertension for Type 2 diabetic patients using ML provides a promising stepping stone for preventing the Type 2 diabetes progression."

3 Data

The Dataset that we are utilizing for this project was found on the UCI Machine Learning Repository E. et al. [2023]. The dataset is divided into 20 sections labeled as follows in Table 1:

Table 1: Dataset columns

ID	N_Days	Status	Drug	Age
Sex	Ascites	Hepatomegaly	Spiders	Edema
Bilirubin	Cholesterol	Albumin	Copper	Alk_Phos
SGOT	Triglycerides	Platelets	Prothrombin	Stage

Table 2: Columns in the dataset.

Most of the data in the dataset was already numeric, but a few of the columns had to be preprocessed before training to convert the values to numeric values. Those columns are listed below in Table 3

Status	Drug	Age	Ascites	Hepatomegaly	Spiders	Edema

Table 3: Columns in the dataset that had to be preprocessed.

All of these columns were enumerated the same way by simply mapping integers to each classification in each column. The only column that was treated differently in the end was the *Status* column due to our findings in Section 5.1.2.

4 Methods

The project started out just using an SVM with a couple different kernels, but as we progressed further into the work the approach changed to incorporate different models and more kernels for our SVM. Our choice of other models were a neural network and the XGBoost classifier, (alongside our original SVM) based on their proven effectiveness in classification tasks and their adaptability to diverse datasets. The SVM, in particular, is well-suited for binary and multi-class classification, which made it a natural choice for our use case. Additionally, neural networks offer flexibility and can capture patterns in the data, while XGBoost excels in handling sequential weak models, which can be advantageous in this scenario. On top of these two alternative models we also looked at the statistical approach used in the "Prognosis in primary biliary cirrhosis: model for decision making" Dickson et al. [1989] paper. After deciding the models we chose to build (which came after creating the SVM), we decided to then compare the evaluation data of each model. This would allow us to actually determine the best model for our use case, with the ability to back it up with the statistics from our experiments.

5 Experiments and Results

We explored various machine learning models to classify patient survival status using the Cirrhosis Patient Survival Prediction dataset. We experimented with both multi-class classification, using the original labels, and binary classification, simplifying the labels to focus on survival. Our models included Support Vector Machine (SVM) with different kernels, neural networks with various layer configurations, and the XGBoost classifier. Evaluation metrics such as accuracy, score, and loss were used to gauge performance (or accuracy over epochs in the case of the neural Network). Our findings can be found in the following subsections.

5.1 Data Splitting

5.1.1 Multi Classification

One thing that we tried was treating the output as it is straight from the data set. The possible outputs of the dataset are either C, CL, or D. Those were mapped to 0, 1, or 2 respectively, and trained on our models as is. The results of this experiment compared to the following Section 5.1.2 can be seen here:

Table 4: SVM Kernel Performance with Multi Classification

Kernel	Score
poly	0.481
sigmoid	0.554
rbf	0.373
linear	0.457

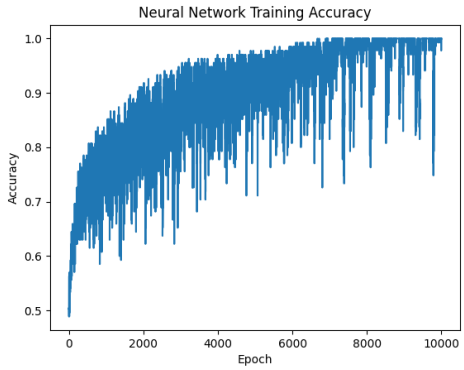

Figure 1: Accuracy over time of 18-10-1 NN with multi classification

5.1.2 Binary Classification

To achieve Binary Classification, we had to make some assumptions about the Status section (our models' prediction). We can reasonably assume that censored status means the patient is still alive. This means that we can treat both *C*, and *CL* statuses as *Survived*, and *D* as *Did Not Survive*. Assuming this results in 1 representing that the patient *Survived*, and 0 representing that the patient *Did Not Survive*.

Table 5: SVM Kernel Performance with Binary Classification

Kernel	Score
poly	0.445
sigmoid	0.566
rbf	0.457
linear	0.506

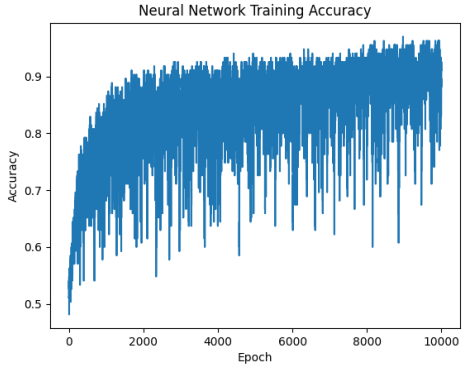

Figure 2: Accuracy over time of 18-10-1 NN with binary classification

5.1.3 Results Conclusion

Based on the data above, it is reasonable to conclude that the section on Binary Classification (Section 5.1.2) was more optimal for fitting the data present in the dataset over Multi Classification (Section 5.1.1

5.2 SVM

5.2.1 Choosing the right kernel

For our kernel on the SVM, we ran 10 training sessions on each of the four kernels (poly, sigmoid, rbf, linear) for a total of 40 training sessions. After running each session, the score values were calculated for each and then averaged for each kernel type. This was done to find the optimal kernel for future training sessions on the SVM. Here are the results before the Data Splitting experiment (Section 5.1):

Table 6: SVM Kernel Performance

Kernel	Score
poly	0.506
sigmoid	0.458
rbf	0.578
linear	0.711

After changing how the data was structured for the output (Section 5.1.2), the kernel scoring was as follows:

Table 7: SVM Kernel Performance

Kernel	Score
poly	0.422
sigmoid	0.482
rbf	0.458
linear	0.518

In both cases (even by a slim lead in the second case), the Linear kernel seemed to be the best choice for our data set on the SVM model.
It was interesting to see that the model performed worse overall with the new structure, but this was favored due to our findings in Section 5.1.2.

5.3 Neural Network

5.3.1 Layers and Sizes

The data we are training on has 20 total columns. One of which is simply an ID that counts up with every entry in the table. Excluding this value, and our target column for the model outputs, we get 18 inputs. The output of our neural network, following our findings in Section 5.1.2, is a single node, and the input layer has a size of 18.
We tried a few layer configurations, one of which is shown in figure 3:

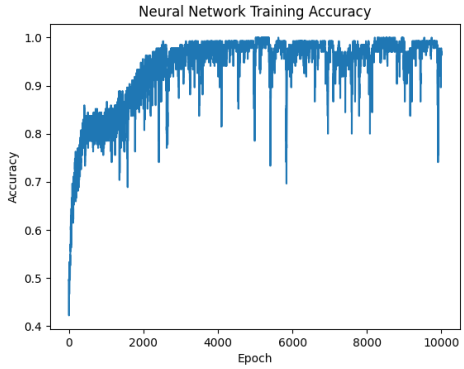

Figure 3: Accuracy over time of 18-10-10-1 NN

Following this training session, the resulting accuracy score from the testing data was only 48.19% accurate. We then tried using one fewer hidden layers after realizing that our model was severely overfitting to the training data.
Here is the resulting training plot:

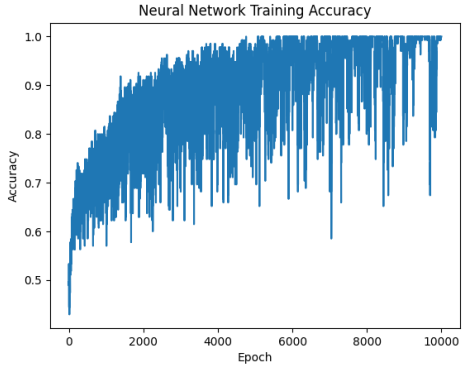

Figure 4: Accuracy over time of 18-10-1 NN

Unfortunately, even with the layer reduction, this model only achieved a slightly better accuracy on the test data of 51.81%. The model did appear to be less overfit than the first, but the model still reached 100% accuracy multiple times during the training period.

5.4 XBG Classifier

5.4.1 Scores for the XBG Classifier

An XGB model was also run on the data set because it applies weak models sequentially, focusing on errors of previous models mlmosaic [2024]. The scores are an average of 10 runs with the data.

Figure 5: Score average and quartiles for XGBClassifer

Using the XGB Classifier the max validation score achieved out of the 10 runs was 44.2835%. This leads us to conclude although the XBG Classifier could be better than some SVM kernels, using either a neural network or an SVM with the linear kernel is the best option available.

6 Conclusion and future work

In our study, we explored different approaches to classify patient survival status using the Cirrhosis Patient Survival Prediction dataset. The methods included treating the dataset as-is for multi-class classification and transforming it for binary classification. We experimented with both multi-class and binary classification. For multi-class classification, we used the original labels (C, CL, D) mapped to 0, 1, and 2, respectively. For binary classification, we restructured the labels to represent survival status, where 'C' and 'CL' were mapped to 'Survived' (1) and 'D' to 'Did Not Survive' (0). We conducted extensive experiments with SVM using different kernels (poly, sigmoid, rbf, linear). The linear kernel consistently outperformed the others, despite the overall lower performance post-restructuring, the linear kernel remained the optimal choice for our dataset. Our neural network models were configured with various layers and sizes. Initial experiments with an 18-10-10-1 configuration resulted in severe overfitting, achieving a test accuracy of only 48.19%. Reducing the hidden layers to an 18-10-1 configuration improved the test accuracy to 51.81%, though overfitting remained a challenge. The XGBoost classifier was evaluated over 10 runs, achieving a maximum validation score of 44.2835%. While this performance was better than some SVM kernels, it did not surpass the linear SVM or the neural network models.

6.1 Final Recommendations

Based on our experiments, the linear SVM and neural network models demonstrated the best performance. The linear SVM also provided a reasonable solution, the possibility of overtaking the SVM with enough testing of layer configurations. Therefore, we recommend using the linear SVM or a thoroughly tested neural network for predicting patient survival status in this dataset.

References

E. R. Dickson, P. M. Grambsch, T. R. Fleming, L. D. Fisher, and A. Langworthy. Prognosis in primary biliary cirrhosis: model for decision making. *National Library of Medicine*, 1989.

Dickson E., Grambsch P., Fleming T., Fisher L., and Langworthy A. Cirrhosis Patient Survival Prediction. UCI Machine Learning Repository, 2023. DOI: https://doi.org/10.24432/C5R02G.

Abduxamidov Jaxongir, Isoyev Erkin, and Muhriddin Doniyorov. Svm 1. `https://www.kaggle.com/code/abduxamidovjaxongir/svm-1`, 2024.

mlmosaic. S3e26 | xgbclassifer. `https://www.kaggle.com/code/markuslill/s3e26-xgbclassifer`, 2024.

Berk Ozturk, Tom Lawton, Stephen Smith, and Ibrahim Habli. Predicting progression of type 2 diabetes using primary care data with the help of machine learning. *National Library of Medicine*, 2023.