Alexander Noah King-Bailey

Introduction to Machine Learning (DAT2025)

Adam Forland

Bootstrap Neural Net Reflection

May 9, 2023

The project involved the development of a predictive model to identify patients with liver disease. The dataset comprised of medical records of liver patients from India, and the objective was to build a machine learning model that could determine whether a patient has negative health signs based on various clinical and demographic features including Bilirubin, Phosphatase, and Proteins.

The types of tools we used for this project were all written in python. But the way we used to create the neural network was a script written by the instructor titled “rrccml.Neural_Net”. From this script, we imported the network tool to officialize the structure of a neural net in our workspace. We included a layer tool so the neural net could process the data using the layers. We also used the script’s activation function so the neural net would produce the correct results.

The first stage of the project involved data pre-processing. Once we imported the necessary tools to display and use the dataset, we made sure it could be properly displayed along with all the right keys. Upon initial inspection of the data, we found a significant amount of null datapoints and irrelevant features. We used this first stage to clear the dataset of such elements in order to make the set easier for us to view as a team and to eliminate potential discrepancies bad points may cause in the neural net’s conclusions.

The next stage was feature engineering, where new features were created by combining or transforming existing features with the intention of training a neural network model using the bootstrapping method. The bootstrapping method involves sampling the original dataset with replacement to generate multiple training sets, each of which is used to train a separate neural net model. The final prediction is obtained by averaging the predictions of each model.

The method we used to divide patients between ‘sick’ and ‘healthy’ was called Binary Classification. At the end of the sets was a column known as ‘dataset’ comprising simply ones and twos. We used ones to classify sick patients and twos to classify healthy patients. From there, we pulled all our raw dataset rows labeled with one and placed them into a new dataset. We did the same with healthy patients, taking all raw dataset rows labeled with two and placing them into their own separate sample. These two new datasets served as our first samples. The sick liver comprised 75 percent of the original raw data and the healthy liver comprised 25 percent.

Once new features were added, we devised a set of targets with which the neural net would use the features to reach for as an objective. The targets represented the binary classifiers in the raw data set. From there, the features and targets were defined as the training data, and this was what we fed through the neural net. Once all layers process the data, then the data was compiled and given a time-span of 500 epochs to formulate it’s conclusion.

The neural net comprised four different layers, each one running through a different a different amount of neurons at a progressively decreasing value; from 500 to 100 to 50 to 1. All neurons in each layer have a connection to the layer that came before. This connection is what the neural net uses to create a weight. A weight is used to improve the conclusion of the network with each iteration it runs through to fulfill it’s purpose. For the sake of this project, we set the iteration to one thousand, which means we made the neural net run through the layers one thousand times until it published its conclusion.

My main contribution to the project was adding comments to other team member's codes. I labeled each individual line of code with what its purpose was in the overall project scheme and how it communicates with other lines that come before and after it. My reasoning behind the comment contents I include was ensuring the code was well-documented, readable, and easy to understand. Despite not possessing as high of a proficiency of writing code compared to my colleagues, I was able to help my team members by reviewing their own codes and providing feedback. I ensured that the codes followed best practices such as consistent naming convention, manageable repository structure, and the capacity to test the code as both good and bad users. By adding comments, I helped other team members understand the code better and make modifications quickly, thereby speeding up the development process. Furthermore, should the coding be consulted later or by non-team members, efficiency of understanding for all parties concerned is maximized.

After training the neural net model, we evaluated its performance using various metrics such as accuracy, precision, recall, and F1 score. All such factors were encountered for in the cost error function. We set the error for zero as a baseline and used the test data sets we created to come up with a percentage for how off-target the neural net was in formulating it's conclusion. The model showed high accuracy and performed well in predicting the likelihood of liver disease in patients. However, we also noticed that the model was biased towards certain demographics, which could lead to misdiagnosis in certain groups of patients. To address this issue, we further analyzed the data and used techniques such as oversampling and undersampling to balance the dataset. Undersampling is a way to render datasets with unbalanced point amounts even by maintaining the size of minority class data while decreasing the majority. Oversampling comes in handy when datasets present limited information; not enough for the neural net to formulate or contribute to the overall conclusion it creates.

As a way of visualizing the progress of the neural net as it came up with a conclusion, we devised a scatter plot using the python tool, matplotlib. We used "epochs" as a measurement of time at which the neural net operates to fulfill it's purpose. The Y-axis of the scatter plot was labeled "score", which was used as a placeholder to tell display the amount of weights it pulls from the different data samples we created as a result of bootstrapping. As a result of creating the scatter plot, we were able to see how far or how close the neural net gets to learning about how to come to the desired conclusion. In this case: which liver patients are sick and which patients are healthy.

Overall, the machine learning project was a success, and my contribution to the project was significant, despite not directly writing any code. By adding comments to other team member's codes, I helped to ensure that the code was not only well-documented and easy to maintain, but presentable to a less technical audience. The project also taught me the importance of working in a team, as each team member had a unique contribution to the project's success. Additionally, it highlighted the importance of addressing issues such as bias and imbalance in the dataset to ensure that the model was fair and accurate.