Luke Bogacz
Vishal Sinha
Jason Long

Final Project, Group Proposal

Project Description:

As part of this project, we want to develop loan default model using the loan attributes and predict current delinquency status which could be 0,1, 2, 3 and so forth depending on how many months the loan is delinquent.

We want to select this problem as this is the foundation problem for any financial problem. As part of this project, we want to benchmark the old problem using neural network and techniques.

Data used:

We will use Fannie Mae public performance data. It consists of the last 18 years of performance data, which should be good enough for a deep neural network. Considering Fannie Mae monthly acquisition to be around 350000 on the average and 4 million loans in a year, we should have around 72 million loans to use.

Deep network design:

We will customize the network to get better performance.

We want to train a neural network model on the last ten years of data, and we can predict the normalized probability distribution on loan default, prepay and actuals using SoftMax function.

The input to the neural network will consist of three input classes, each of which will specify the default, prepay and actuals:

- A loan is **default** if the loan is not paid in full.
- A loan is **prepaid** if the principle and is paid in full before the loan term is expired.
- A loan is **actual** if the payment of the loan is coming as expected.

Framework:

A pytorch framework will be used for the project. The reason being we are familiar with the framework and can customize more easily.

Reference material:

https://pytorch.org
https://en.wikipedia.org/wiki/Softmax_function
https://pdfs.semanticscholar.org/9fc6/01d098eb16e10baae87b222cc0aec7bb5112.pdf

Judge the performance of the network

Split data into training, validation and test sets and use percentage correctly classified as a first means to understand accuracy. Additionally, we will use class-based ROC Curve plots and AUC to check for accuracy and performance. For tuning and general performance, we will monitor weight gradient change through histogram plots and use the standard deviation or mean to show changes in weight during iterations (changes should get smaller each time). We will use a confusion matrix to understand correlations between classes and false positives.

Schedule for computing the results:

Select the problem/data/model by March 29th, completed.

Write the project proposal, assign duties to the group members, submit via blackboard to Dr. Jafari by April 6th. Group effort

Semi-concurrent tasks to be completed each member by April 19th:

- Luke: Create the data loader for the mortgage data due to the size of the data not fitting on Github (~ 1GB per month), and the originating source requires password access.
- Vishal: Create the initial network using a sample set of data.
- Jason: Identity probable visualizations/analytics for Pytorch financial data models. Create the initial visualizations (plots) and analytics for the output/results using a sample of output/results.

Complete the training and tune the model by April 22h.

Complete visualization/analytics of the model by April 26nd.

Complete the 'Group Final Report' and the 'Individual Final Report's, stable and tested Github repo (able to run on unassociated accounts) by April 27th.

Validated satisfaction of the 'Deliverables,' (1.4) requirements, by April 29th.