Maximizing Returns on Lending Club with Machine Learning

Drew Mathieson
(andrew4@stanford.edu)
Abiel Gutiérrez
(abielg@stanford.edu)
CS 229 Fall 2017

Introduction

We sought to maximize the investment opportunity provided in peer to peer lending. Using the Lending Club dataset, we developed a criteria for creating and evaluating a given portfolio. Our evaluation metric was the **Sharpe Ratio**:

$$S = \left(\frac{R_p - R_f}{\sigma_p}\right)$$

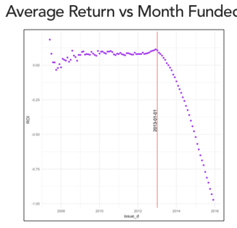
We maximized this metric via a linear and logistical regression, as well as with a neural network fully connected model.

Dataset

200,000 loans (either 3 or 5 year maturity) that were either paid or charged off between 2008 and 2013, obtained from Kaggle and the LendingClub platform.

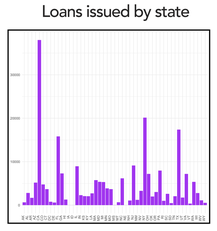
We removed 119 missing or irrelevant features
Also Removed all rows with missing data (<1% of total data)

Average Return vs Month Funded



We dropped loans that hadn't matured since their ROI was naturally lower.

Loans issued by state



We dropped categorical variables which occured with low frequency .

Approach

1. Maximize Portfolio Returns: Trained Linear Regression model to predict annual return on investment
2. Minimize Portfolio Variance: Trained Logistic Regression and Neural Network to assign probabilities for loan defaults.

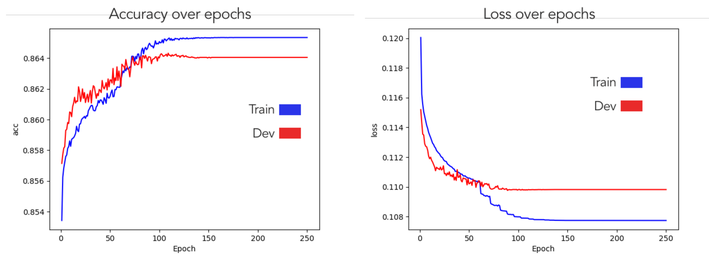
With this, we maximize the Sharpe ratio, which is the average return earned in excess of the risk free rate of return per unit of volatility.

Results

Models: Linear Logistic Neural Network

Test Baseline: 2.82% Test Baseline: 8.62% Test Baseline: 0.12%%

Loan Default Neural Net Classification



	Training	Validation	Test
Baseline	85.20%	85.22%	85.17%
Neural Net	86.54%	86.41%	85.09%

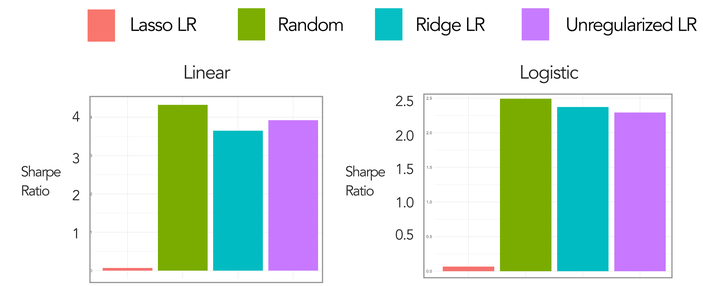
Significant Features

- Loan Subgrade
- Loan Purpose
- Interest Rate
- Debt to Income Ratio
- Employment Length

Models

Regressions

Done on Annural ROI using 32 features

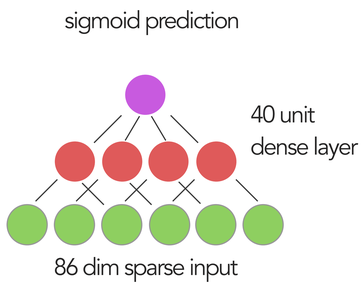


Neural Network

40 unit hidden layer with PReLU activation

SGD optimizer, 0.001 learning rate with 0.6 decay per 5 batch patience interval.

Mean squared error loss function



No regularization; dropout decreased validation accuracy

Implemented with Tensorflow's Keras high level API.

Next Steps

1. Train Neural Network to predict Return on Investment
2. Combine return maximizing and variance minimizing models in a meaningful way to build optimal portfolio
3. Iterate on current models to improve performance.