Time to Default & Time to Prepayment Estimation Using LSTM and Deep Learning Techniques

Peter G. Drembelas
pdrembel@stanford.edu

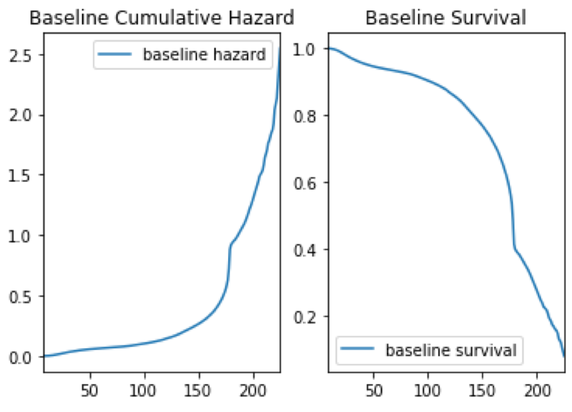
Vibhor V. Khar
vvkhar@stanford.edu

https://youtu.be/NlxVGXZfzuc

Introduction

- This model predicts the status of a mortgage in the future, given it has not already faced a hazard.
- A Long Short Term Memory (LSTM) model was developed with 45 features that include various time-insensitive / time-dependent loan level characteristics and macroeconomic variables.
- The model was trained to predict hazard events within 20 months.
- The model accuracy is then scored on an out-of-time development / test datasets which is comprised of 20 months of data from 2015.

Survival Rate of the First 200 months of a Mortgage



Baseline Models

- Historically, a Cox-Proportional Hazard (CPH) model was used to predict mortgage hazards and had an accuracy of 73% within the dataset used.
- Below is the definition of the CPH and its likelihood function.

$$\lambda(t|X_i) = \lambda_0(t)e^{\beta_1 X_{i1}+\ldots+\beta_p X_{ip}}e^{X_i\beta}$$

$$L_i(\beta) = \frac{\lambda(Y_i|X_i)}{\sum_{j:Y_j\geq Y_i}\lambda(Y_i|X_j)} = \frac{\lambda_0(Y_i)\theta_i}{\sum_{j:Y_j\geq Y_i}\lambda_0(Y_i)\theta_j} = \frac{\theta_i}{\sum_{j:Y_j\geq Y_i}\theta_j}$$

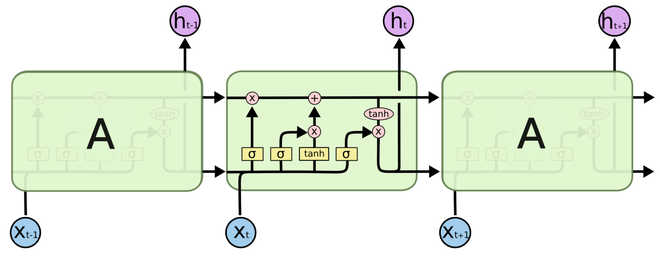
Dataset

- Mortgage Data from FannieMae consists primarily of US 15 and 30 year conforming fixed rate loans
- Macroeconomic data merged onto loan level data.
- Features normalized of one-hot encoded, if categorical.
- Development and Test data set were derived from out-of-sample and out-of—time data. Training contained 1MM loan records.

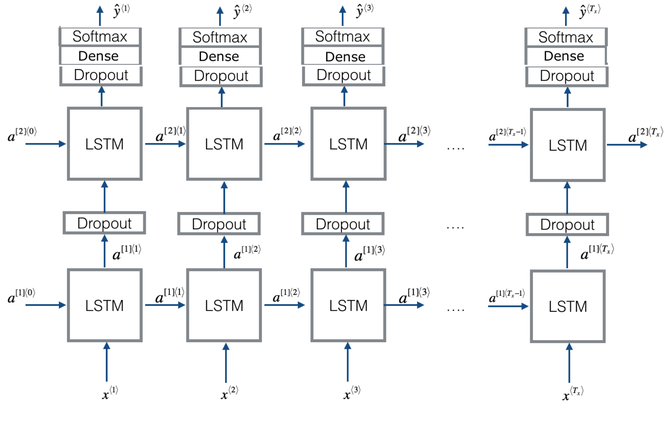
Method

- Sequential data lends itself naturally to the LSTM architecture.. Tensorflow with Keras was utilized. utput of each LSTM Cell (Cell shown below) is fed forward to the next cell in line.
- Two layers of LSTM tended to converge more rapidly and offered a greater degree of non-linear learning.
- Dropout greatly reduced variance and decreased overfitting, leading to improved dev and test performance.
- A softmax output was used in order to categorize the likely output hazard event at each time.

LSTM Cell



Complete RNN Architecture



Optimization

- The model uses RMSProp along with a weighted categorical cross entropy loss function shown below.
- Though hyperparameters could have been used to identify weights an initial value of the inverse proportion of category to total labels seemed to suffice.

$$\mathcal{L} = -\frac{1}{T_x}\frac{1}{m}\sum_{t=1}^{T_x}\sum_{j=1}^{N_{states}}\sum_{i=1}^{m} w_j y_{i,t}\log\hat{y}_{i,t}$$

Results

- We considered the same variables in the CPH method as well as the LSTM
- For sufficiently large number of neurons the RNN easily outperformed the baseline.
- We needed to ensure that the model was not over fitting on the train and dev datasets.
- Hyperparameter tuning was conducted on 5 hyperparameters.
- Final model was selected on the basis that it obtained a high accuracy in the training set over the baseline and ensured the model did not overfit over train and dev samples.
- We noted that our model not only predicts the occurrence of a hazard event with 94% accuracy in the train dataset/ 90% accuracy in dev / test datasets

Future Work

- increase train data accuracy by training on larger samples given higher computing power allowances.
- Implement the architecture used in RNN Surv and tailor to the mortgage domain.