CS230

Application of Deep Learning Models for Pricing Bonds

Andrey Popov
Department of Computer Science
Stanford University
apopov@stanford.edu

Abstract

Financial markets for fixed income securities are generally less liquid and transparent comparing to the equity markets. The lack of available tradable prices combined with the limited liquidity create problems for potential investors who are unable to trade these instruments. Availability of predicted tradable prices would improve opportunities for market participants. In this research it has been examined whether neural network models can predict prices of bonds.

1. Introduction

Financial markets and their participants rely heavily on the information about future prices of certain assets. The equity and FX markets have been the main drivers behind developing a number of fundamental and quantitative approaches for making predictions about asset pricing. They vary from each other significantly in terms of complexity, accuracy and input data required but most of them have a goal of providing an estimate of the future asset price, which can further be used in derivatives pricing models.

On a quantitative side equity asset pricing has been developing from basic regression models to more sophisticated ARMA/ARIMA models, better suited for forecasting real world time series data. ARCH/GARCH models developed by Tim Bollerslev and Robert Engle has become another important milestone in quantitative finance allowing to model autocorrelated nature of volatility so relevant to many financial assets. Later developments of binominal trees and Monte Carlo simulations have become increasingly popular for derivatives pricing and algorithmic trading.

The main focus on fixed income side has been related to the better modelling of the yield curves and impacts of the yield movements on the price of the security. While there are number of sophisticated models used for evaluating bonds and their derivatives, these securities are mostly less liquid and therefore far less observable for proper statistical modelling.

The increasing popularity of Machine Learning reached the financial industry alongside others. Algorithmic trading has been one of the biggest beneficiaries of applied machine learning for trading and portfolio rebalancing. Convolutional Neural Networks (CNN) and recurrent neural networks (RNN) have become most innovative approaches in the industry for doing analysis of the financial data, allowing to capture complicated relationships between variables and suitable for some of the most complex tasks, where human might fail.

In this paper several sequential neural network models have been applied to the large dataset of time series containing bonds prices and yields. The results of training and testing different NN models have been compared between each other and with those of the regression analysis.

2. Related work

As machine learning methods have been developing, more authors started to compare them with the traditional quantitative forecasting to compare achieved results. The main focus of researches has been the equity market. In [12] Kelly and Xiu performed overview of ML methods in comparison to the regression analysis and identified superior performance of neural networks (NN) for predicting US stock prices and constructing portfolios when compared to the various methods of regression analysis. However, outperformance of NNs is not guaranteed, as Lee et al. in [9] showed that Seasonal ARIMA model can outperform simple NN model when forecasting stock prices over long-term periods.

RNN models have become one of the main ML techniques employed by researchers. In [3] and [6] authors point out that one of the best models for modelling financial data is Long Short-term Memory (LSTM), a specific version of RNN. LSTM contains specific memory blocks with information about temporal state of the network. Zou and Qu [7] compared single-layer LSTM model with its modified versions known as Stacked-LSTM and Attention-LSTM and in case of the Attention-LSTM have been able to achieve significant reduction in MSE.

While there are few papers focusing on applying ML to fixed income securities, the research published by Ganguli and Dunnmon [1] is one of the most relevant to this project. Authors applied a variety of ML techniques to the large population of US corporate bonds (rather than a single asset), predicting future prices based on the previously observed trade levels. They managed to achieve the lowest prediction error by utilizing 2-layered NN. Following this, Bacas [2] performed similar work on the same dataset, achieving best results in terms of MAE for multi-layered NN and LSTM. Finally, Gaurav in [4] demonstrated that NN can be implemented for predicting yield-price relationship on a single bond with a high degree of accuracy.

Based on the reviewed papers, the following research areas have been identified:

- The ability of LSTM model to make an accurate forecast in comparison to other methods;
- Which version of LSTM model works best for predicting prices of the financial assets;
- Suitability of NN for making multivariate predictions with diverse time series dataset of financial assets.

3. Dataset

In [1] authors obtained a dataset of 827k observations of US bonds, and here the focus has been on obtaining population of the similar size. The trading data information have been gathered for 13,630 European and bonds quoted in EUR , CHF or GBP for the 4.5 months of 2022, making the total number of observations 991,326. The dataset contains information about traded price and yield (obtained from Bloomberg), quantitative information (e.g. time-to-maturity, coupon) and qualitative data most relevant for splitting the universe (currency, country, industry and credit rating).

Date	ISIN	Price	Yield	TTM	Currency	Country	Industry	Group	Credit rating
22/04/2022	XS1598103213	99.8019	1.72712	0.33	GBP	FR	Financial	Banks	AA-
19/04/2022	XS1605745261	99.917	7.80831	0.09	GBP	GB	Financial	Savings&Loans	NR
23/04/2022	BE0312770424	100.003	-0.26981	0.12	EUR	BE	Government	Sovereign	AA+

Table 1: Sample of the original dataset

The dataset has been split between train/dev/test set in the proportion 68/16/16. Due to the nature of sequential models, it has been decided to take into the test set only observations from the last month thus preventing model from being trained on the test set data.

As a first step, we have to convert qualitative fields into measurable variables by applying the logic of *OneHotEncoder* script. The script adds new binary feature for each possible category, in this case it resulted in a significant increase of total features to 154.

In order for selected LSTM models to perform well the data has been rescaled using *MinMaxScaler*. Also, while it is being shown in [4] that neural networks can be used to predict exact bond prices, the preferred output data to work would be yield to preserve comparability between different securities.

4. **Model selection and evaluation**

Based on the research studied the following models have been selected: 1) Time Series Model, 2) Single-layer LSTM, 3) Stacked-LSTM, 4) Attention LSTM and 5) CNN-LSTM. In order to achieve the highest accuracy, Mean Squared Error (MSE) have been selected as a primary measure to evaluate quality of the models.

a. Time series models: ARIMA

Autoregressive models gained significant popularity among economist and finance professionals a due to their simplicity and ability to accurately forecast trends. The formula for ARIMA with parameters (p, d, q) is the following:

$$\delta Y_t(p,d,q) = \mu + \sum_{p=1}^{p} (\varphi_p * \delta Y_{t-p}) - \sum_{q=1}^{q} (\theta_q * e_{t-q}), \; where \; \delta Y_t = Y_t - Y_{t-d}$$

While ARIMA is fairly simple to implement, the main limitation of the model is the lack of non-linear impact and limited response in case of changing patterns in the data as well. ARIMA (5,1,0) has been used to take into account prior week's yields and compare the results with other deep learning models.

b. Single-layer LSTM:

LSTM has demonstrated decent results when working with a sequential data, such as text recognition. It is also widely applied to the financial data, as the model allows to keep hidden states throughout time and do a reliable forecasting.

The main feature of the LSTM model is the existence of the memory cell – a specific state which is able to filter and assign weights to the new incoming information through neurons. The memory cell has the ability to update and amend the long-term dependencies learned by the RNN on the previous sequence steps by applying input, forget and output gates and defining their output through sigmoid function. Then this information about the memory cell state is passed to the next step.

LSTM module in *Tensorflow* requires specification of "lookback" window which defines how many previous observations and features we would need for predicting the current step. We selected 5 days history (a week of trading data) for all of our LSTM models. In line with the research reviewed, a single-layer LSTM with 64 hidden units and MSE optimization has been selected as a base model. In addition, early stopping mechanism has been added later to avoid running additional epochs once MSE reaches plateau level.

c. Stacked-LSTM

Often the impact of the change in the input data could be quite complex, in that case Stacking LSTM layers within NN model allows to achieve higher accuracy (although that doesn't always result in a better performance). Typically, there no more than three stacked layers and efficiency of additional layers falls quickly. We chose to employ the following structure of 2-layer LSTM:

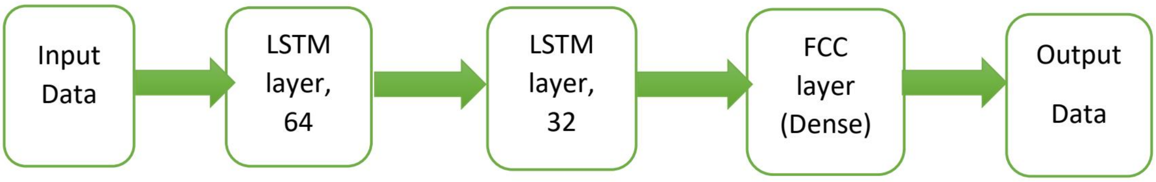

Figure 1: Stacked LSTM architecture

d. Attention-LSTM

Attention-LSTM was initially developed as part of the resolving encode-decoder issues in natural language processing, attention mechanism tries to replicate the behaviour of a human brain by assigning weights to the input vector. That helps neural network to focus on the important features of the incoming information. The mechanism has been implemented through applying '*SoftMax'* function to the weights e_t as shown below:

$$\alpha_t = \frac{exp\,(e_t)}{\sum_{k=1}^{T} exp\,(e_k)}, \text{ where } e_t = tanh\,(W_a[x_1, x_2, \dots, x_T] + b)$$

e. CNN-LSTM

Convolutional Neural Networks are primarily used in the computer vision and speech recognition as it allows to extract important features from a multi-dimensional input data. In CNN model convolution layer (responsible for extracting features) t is followed by the pooling layer, which is reducing dimensionality. Since CNN on itself hasn't demonstrated the best results when working with time series, LSTM layer is added on top of it to improve performance of the model. Following the approach of Lu et al. [6] it has been decided to use 1D convolutional layer with 64 Neurons, "same padding" and a kernel size of 3. After the Conv1D layer, Max Pooling layer with the pool size of 2 has been selected.

5. Results

The five selected models have been trained, in case of NN models epochs limit has been set to 100 and selected *Adam* optimizer with the default learning rate of 0.001.

The initial result of applying single-layer LSTM has demonstrated that test MSE has been 50% higher comparing to that of the training set (0.1504 vs 0.09523). That flagged potential overfitting problem hence it's been decided to incorporate a dropout layer with probability of 0.2, which helped reducing the difference in MSE between sets.

When reviewing residuals produced by ARIMA and LSTM models it's been identified that there are few significant outliers mostly belonging to junk or pre-defaulting bonds due to very high yield. These outliers represent less than 3%of the total population hence it's been decided to remove them from the dataset. As shown in the table below all models achieved significant reduction in MSE for both sets of data:

	With outliers		Without outliers			
	Train MSE	Test MSE	Train MSE	Test MSE	Epoch stop	Running time
ARIMA	0.217	0.372	0.064	0.078	n/a	6 minutes
Single LSTM	0.098	0.114	0.039	0.054	67	46 minutes
2-layer LSTM	0.1107	0.1231	0.040	0.057	78	58 minutes
2-Layer LSTM + Attention	0.072	0.094	0.029	0.035	100	1hr 45 minutes
CNN-LSTM	0.088	0.101	0.032	0.041	100	1hr 14 minutes

Table 2: MSE summary for the analysed models

It is clear that ARIMA performed way worse comparing to the NN, which was in line with the expectations. Surprisingly 2-layer LSTM didn't result in any improvements comparing to single-layer model. Attention-LSTM and CNN-LSTM demonstrated the best results, coming very close to each other, although the training time is longer for both of the.

6. Conclusions and future work

The achieved results demonstrated that bond prices can be fairly accurately predicted using sequential neural network models. Additional layers to the LSTM model like attention and CNN can significantly improve quality of the model, resulting in a big reduction of MSE. Moreover, multivariate LSTM models can effectively handle securities from different currencies and industry sectors, as shown in the chart below:

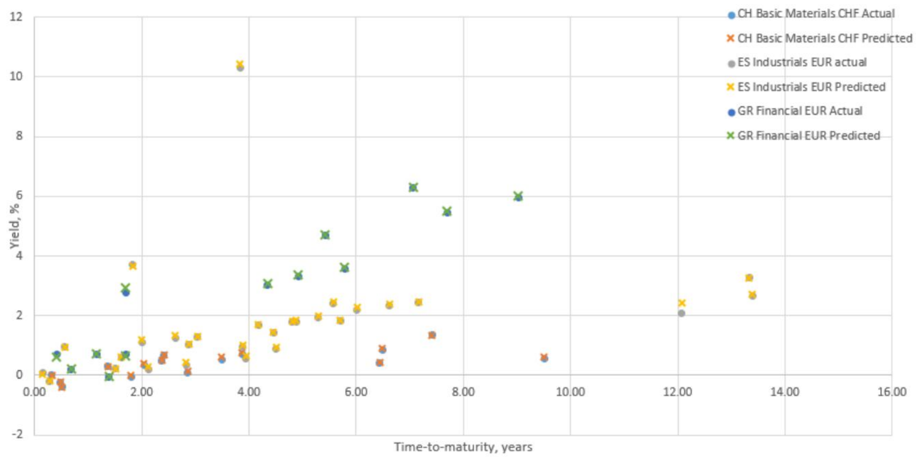

Figure 1: Actual vs predicted bond yields as of 01/05/2022

However, required exclusions of high-yield/defaulted securities highlighted that even NN have limitations of their explanatory powers and potentially require additional qualitative/quantitative features to be added. In addition, one has to be careful when applying LSTM models over longer periods of time as it's predicting power fades quickly and more complex, ensemble models would be required in that case.

7. Contributions

I would like to thank my Teaching Assistant Fenglu Hong for providing guidance and advices when doing this research as well as sharing her own work on the related topic.

References

[1] Ganguli, S., Dunnmon, J. (2017) Machine Learning for Better Models for Predicting Bond Prices. [Online]. Available: https://arxiv.org/pdf/1705.01142.pdf

[2] Bacas, T. (2018) Predicting Accurate Bond Prices with Neural Networks. [Online]. Available: https://medium.com/@thomasbacas/predicting-accurate-bond-prices-with-neural-networks-ff46c051c25c

[3] Bao, Wei, Jun Yue and Yulei Rao. (2017) A deep learning framework for financial time series using stacked autoencoders and longshort term memory. PloS one 12.7 (2017): e0180944

[4] Gaurav S. (2017) Deep Learning Neural Networks for Bond Pricing. [Online]. Available: https://www.linkedin.com/pulse/deep-learning-neural-networks-bond-pricing-gaurav-singh/

[5] Vithayapalert, W. (2019) Deep Learning prediction on Price Movement for NASDAQ. [Online]. Available: http://cs230.stanford.edu/projects_spring_2019/reports/18639473.pdf

[6] Wenjie Lu, Jiazheng Li, Yifan Li, Aijun Sun and Jingyang Wang. (2020) A CNN-LSTM-Based Model to Forecast Stock Prices. [Online]. Available: https://www.hindawi.com/journals/complexity/2020/6622927/

[7] Zou Zhichao, Qu Zihao. (2020). Using LSTM in Stock prediction and Quantitative Trading. [Online]. Available: https://cs230.stanford.edu/projects_winter_2020/reports/32066186.pdf

[8] Hamoudi, H., Elseifi, M. (2021). Stock Market Prediction using CNN and LSTM. [Online]. Available: http://cs230.stanford.edu/projects_winter_2021/reports/70667451.pdf

[9] CK Lee, Y Sehwan, and J Jongdae. Neural network model versus sarima model in forecasting korean stock price index (kospi). Issues in Information System, 8(2):372–378, 2007

[10] Gautama, E., Widiputra, H., Mailangkay, A. (2021). Multivariate CNN-LSTM Model for Multiple Parallel Financial Time-Series Prediction. Complexity, 9903518, 10/2021.

[11] Chen, L., Petler, M., Zhu, J. (2019). Deep Learning in Asset Pricing. [Online]. Available at: https://economics.yale.edu/sites/default/files/deep_learning_in_asset_pricing.pdf

[12] Kelly, B., Xiu, D. (2020). Empirical Asset Pricing via Machine Learning. *The Review of Financial Studies*, Vol. 33, Issue 5, May 2020.