Comparing ANNs to traditional forecasting methods

Alexandros Antoniou

Centre for Computational Finance and Economic Agents

CSEE

University of Essex

Submitted in partial satisfaction of the requirements for the
Degree of Master of Science
in
Computational Finance

Supervisor	Dr Maria Kyropoulou
Second Supervisor	Dr John O'Hara

August 2020

Acknowledgements

I would like to thank my supervisor, Dr Maria Kyropoulou, for helping me with this thesis. I would also like to thank my friends and family for keeping me sane during this lock-down and overall worldwide panic. It has not been a fun time.

Abstract

TODO

Keywords: Deep Learning, Artificial Neural Networks, Stock Market, Time Series prediction, Neural Networks

Table of Contents

1 Introduction

The ability to predict changes in stock prices is extremely important to the financial world as it influences trading strategies and reduces risks in the market. Forecasting has long been a problem for the business and technology communities and has seen little advances until quite recently, with the advent of neural networks and deep learning.

Before artificial neural networks, the finance world used other methods to model the time series that arose from the continuous updating of stock prices. Models like the autoregressive moving average model (ARIMA) and the generalised autoregressive conditional heteroscedasticity model (GARCH) are key econometric methods for forecasting time series and are still widely used in finance.

The focus of this project is to provide a comparison between the traditional methods for time series forecasting such as the ARIMA model, and simple implementations of artificial neural networks, in the context of financial time series prediction.

The prices of stocks can be modelled as non-linear time series, which have been at the centre of attention in the finance world since the 1970s with George Box and Gwilym Jenkins popularised their Box-Jenkins method for finding the best-fit of a time series model[2].

2 Related Work

This section will introduce work related to stock market prediction, namely traditional asset pricing models and work related to the development of generative adversarial networks.

2.1 Stock Market Prediction models

ARIMA(1,1,1) data mining can predict stock prices [3]

2.2 Testing some math

Here are two equations:

$$a = b + 1 \tag{2.1}$$

$$\frac{\hbar^2}{2m}\nabla^2\Psi + V(\mathbf{r})\Psi = -i\hbar\frac{\partial\Psi}{\partial t} \tag{2.2}$$

And here is some text with some nice inline math, (x, y) wow γ so cool ρ.

2.3 Testing citations

This is Fama[1] and this is Goodfellow. This is another GAN citation.

3 Methodology

The following section provides details in the construction of the model for predicting stock prices, as well as a breakdown of the data used in the training of the network.

3.1 Time-Series Forecasting

Financial data are discrete in time and as such can be modelled as time-series with calculated means and standard deviations.

* time series analysis

* exploratory analysis

3.2 Autoregressive Models

3.3 Model Description

The network is a relatively simple network by most accounts, comprised of a single hidden layer. The simplicity of the model only goes to show the power of neural networks in fitting and forecasting time-series. The model consists of three layers, an input layer, a Long Short-Term Memory (LSTM) layer and the output layer, as shown in 3.1.

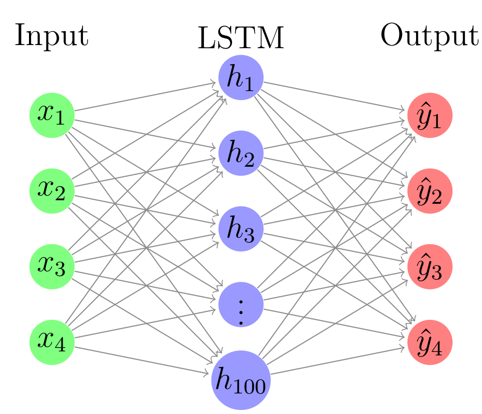

Figure 3.1: Model Architecture

The input layer receives a 2-dimensional array of historical stock values, including the previous day's opening, highest, lowest and closing stock prices. The network then allows the neurons to compete amongst each other and determing an appropriate output. Output is in the shape of a 1-dimensional array containing four values, the future day's predicted stock values for open, high, low and close. The LSTM layer contains 100 nodes and is trained for 100 epochs. The Rectifier Linear Unit was used as its activation function and the Mean Absolute Error was used as its loss function.

Nodes	**Epochs**	**Optimizer**	**Learning Rate**	**Activation**	**Loss**
100	100	Adam	0.001	ReLU	MAE

Table 3.1: Description of the hidden LSTM layer

In this project we made use of the Keras API to implement the network. Keras is written in Python and operates ontop of TensorFlow, which is an extremely popular and widely used Deep Learning library. A simple sequential model using LSTM cells built using Keras would look like this:

```
model = Sequential()
model.add(LSTM(100, activation="relu", input_shape=(n_steps, n_features)))
model.add(Dense(n_features))
opt = Adam(learning_rate=0.001)
model.compile(optimizer=opt, loss="mae", metrics=["mse"])
```

With as little as 5 lines of code, we have a working Long Short-Term Memory model ready for training. The LSTM cell easily remembers the long term dependencies in the data and outputs a 1-dimensional array containing future values.

Here is a sentence, and you can see a nice picture in Figure 3.2.

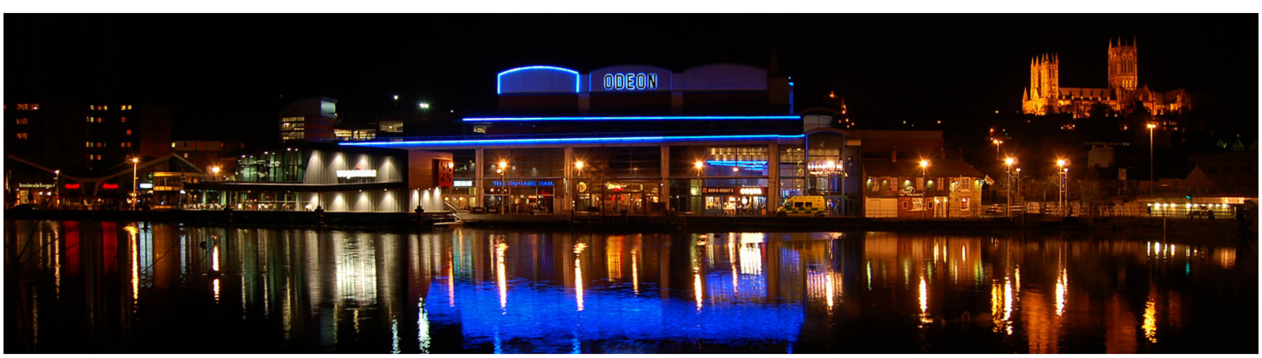

Figure 3.2: A picture of the Brayford from Google Images.

Also, a table can be found in Table 3.2. You should use a LaTeX table generator like `https://www.tablesgenerator.com/` if you want to make your life easier.

Table 3.2: Here is a table. The caption goes above like this.

First name	Last name	Age
Bob	Bobbington	24
Joe	Bloggs	37
Billy	Bob	10

4 Results

4.1 Dataset

We retrieve data on tickers available at `https://www.tiingo.com/`. The stock exchanges targeted in this project are the New York Stock Exchange (NYSE), the National Association of Securities Dealers Automated Quotations (NASDAQ) and the Financial Times Stock Exchange (FTSE). The data comprise of companies from the technology and the financial services industries.

The data consist of daily values for the period starting January 1, 2016 and ending December 31, 2019. The dataset includes daily information for the `close`, `open`, `high` and `low` prices of each trading day. Below is a table listing all stocks considered for the project:

Stock Name	**Symbol**	**Stock Exchange**
Apple Inc.	AAPL	NASDAQ
Amazon Inc.	AMZN	NASDAQ
American Express Company	AXP	NYSE
Boeing Co	BA	NYSE
Bank of America Corp	BAC	NYSE
Citigroup Inc	C	NYSE
Ford Motor Company	F	NYSE
Facebook Inc.	FB	NASDAQ
General Electric Company	GE	NYSE
Alphabet Inc. Class C	GOOG	NASDAQ
Goldman Sachs Group Inc.	GS	NYSE
JPMorgan Chase & Co.	JPM	NYSE
Morgan Stanley	MS	NYSE
Microsoft Corporation	MSFT	NASDAQ
Wells Fargo & Co.	WFC	NYSE

Table 4.1: Stocks included in the study

4.2 Evaluation Metrics

4.3 Results

5 Conclusions

...

References

[1] E. Fama, 'Efficient capital markets: A review of theory and empirical work,' *Journal of Finance*, vol. 25, no. 2, pp. 383–417, 1970 (cit. on p. 2).

[2] G. J. George Box, 'Time series analysis, Forecasting and control,' 1970 (cit. on p. 1).

[3] M. M. Sasan Barak, 'Developing an approach to evaluate stocks by forecasting effective features with data mining methods,' *Expert Systems with Applications*, vol. 42, pp. 1325–1339, 3 2015 (cit. on p. 2).

List of Tables

List of Figures