Can deep learning predict Bitcoin next day price trends?

Martin Hughes. Snr: 2017786

MSc. Cognitive Science & Artificial Intelligence

Tilburg University

Supervisor: Dr. Henry Brighton

Second reader: Dr. Drew Hendrickson

July 2018

Abstract

This thesis investigates whether deep learning approaches can be used to predict next day price trends of the Bitcoin cryptocurrency. Many researchers have looked at predicting next day financial time-series trends using economic or public awareness information, evaluating their results via performance over a test set. This thesis investigates combining unique economic and public awareness measures and using Artificial Neural Networks for trend prediction. This thesis takes a different approach; not only will results be evaluated via performance over a test set, but by also trying to order predictions in terms of the confidence with which they are made. To this end, a novel use of a Bayesian approximation network is presented. This investigation finds that Bitcoin trend prediction accuracy of Bayesian approximation network predictions increases with prediction confidence. The investigation also demonstrates that a profitable Bitcoin investment strategy can be determined from a Bayesian approximation network trained only on public awareness features.

Can deep learning predict Bitcoin next day price trends?

Preface

My thanks to Dr. Henry Brighton for providing guidance during this investigation. I wish to acknowledge the Gdelt project (https://www.gdeltproject.org/), Google (https://trends.google.com), and Coin Market Cap (https://coinmarketcap.com/) for providing the data used in this thesis.

Contents

Introduction

Over the years Machine Learning approaches for predicting financial time-series trends have received a lot of attention from academics. A typical financial time-series prediction algorithm aims to predict whether the next day closing price of a stock or currency will either increase or decrease compared to the current day's closing price. In essence it is a simple binary classification task, a '1' predicts a next day price rise and a '0' predicts a next day fall in price compared to the current day's price. Artificial Neural Networks (ANNs) are often used to perform this classification task. ANNs have been demonstrated to predict next day stock index trends with reasonable accuracy when trained on features engineered from historical economic data (Kara, Boyacioglu, Baykan, 2011; Qiu & Song 2016; Senol & Ozturan, 2008). In addition to gleaning insight from historical economic data, model features extracted from various public awareness data have also been demonstrated to be very useful in financial time-series prediction (Bollen, Mao and Zheng, 2011; Borges, Goldstein, Ortmann and Gigerenzer, 1999; Guan-Ru Wu, Chieh-Tse Hou and Lin, 2018).

In 2017, Bitcoin burst into global public consciousness like never before due to it's wild price rise and turbulent volatility. The cryptocurrency started 2017 valued at $1000 yet reached over $14,500 per Bitcoin by the end of the year (Chaparro, 2017). Bitcoin was heavily talked about and discussed in the news, online and in social media. Today, the internet is awash with opinions, advice, and warnings from supposed Blockchain experts about the future price trends of Bitcoin. Researchers have looked in depth at potential drivers of cryptocurrency prices (Garcia, Tessone, Mavrodiev and Perony, 2014; Kristoufek, 2013, 2015) and there is an emerging body of work looking at predicting cryptocurrency behaviour (Jang & Lee, 2018; Kim et al., 2016 ; Shah & Zhang, 2014).

Despite academic research into predicting cryptocurrency behaviour, no work has yet been done on trying to order daily prediction results with regards to their accuracy. To tackle this problem an appropriate technique was sort out that would allow for daily model predictions to be ranked in terms of the confidence attributed to each prediction by the model. In 2016, Gal and Ghahramani introduced using Dropout in a specific way in ANNs so as to create a Bayesian approximation. Such a Bayesian approximation network allows for the confidence of each prediction made by a network to be retrieved. This thesis investigates using such a Bayesian approximation network for predicting next day price trends of the Bitcoin cryptocurrency and for ordering the predictions in terms of confidence. The network is trained on a combination of features engineered from historical Bitcoin economic data and features obtained from measures of public awareness. This investigation is unique in it's approach to Bitcoin trend prediction due to it's novel use of a Bayesian approximation network and unique combination of model features. Commonly, predictions of financial time-series are not ordered with regards to an increased accuracy likelihood (Jang & Lee, 2018, Kara, Boyacioglu, Baykan, 2011; Qiu & Song 2016; Senol & Ozturan, 2008). Being able to rank predictions in terms of those more likely to be accurate to those more likely to be less accurate would be incredibly useful from a financial perspective. A next day stock index or Bitcoin trend predictor with a poor general accuracy could still be very valuable to a trader or investor if confidence information is provided for each prediction (provided prediction accuracy is proven to increase with prediction confidence).

This thesis reports some interesting results and makes some novel contributions to cryptocurrency trend prediction. It is demonstrated that confidence information from a Bayesian approximation network can, to a certain extent, be used to order predictions in terms of confidence (whereby accuracy is higher for the most confident predictions compared to less confident predictions). This is an original and new approach to tackling the problem of ordering financial time-series daily predictions. This thesis also reports the exciting finding that a profitable Bitcoin investment strategy can be determined from a Bayesian approximation network trained only on public awareness features. Furthermore, this thesis also highlights the pitfalls of inadequate financial time-series evaluation and suggests the appropriate combination of evaluation criterion to be used in order to communicate the robustness of prediction results.

Research questions

- Some financial time-series prediction research looks at predicting next day prices (continuous variables) rather than a simple up or down trend prediction. Often only root mean square error (RMSE) and mean absolute percentage error (MAPE) are used to evaluate such models (Jang & Lee, 2018; Ticknor, 2013). RMSE and MAPE can however be misleading, as trend prediction can still be very poor despite low RMSE and MAPE results. This thesis sets out to clarify whether evaluating financial time-series prediction with only RMSE and MAPE criterion is sufficient, and if not, what additional evaluation criterion are required to prove the robustness of results.

- An ANN will be built for next day Bitcoin price trend binary classification prediction. The network will be trained on features derived from both historical economic and public awareness data. Can the network predict Bitcoin price trend with an accuracy score better than the majority class for a specified test set?

- A Bayesian approximation network will be created from the ANN architecture via the Dropout technique (Gal & Ghahramani, 2016). When next day Bitcoin price trend predictions are ordered in terms of the network's confidence in each prediction; does prediction accuracy increase with prediction confidence?

- The training features of the Bayesian approximation network can be grouped into economic features and public awareness features. Which of these features is most important to the Bitcoin price trend prediction task?

Cryptocurrencies

Cryptocurrencies are digital, decentralized currencies that work as a form of exchange. Bitcoin, the world's first cryptocurrency, and it's underlying blockchain technology was first introduced in a 2008 article by Satoshi Nakamoto. Block chain technology allows for the movement of digital currency without a third trusted party and represents an immutable ledger of transactions that is distributed over a network of users (Nakamoto, 2008). In many ways, the Blockchain is a superior and more democratic transaction system than traditional transaction methods as it represents a borderless and open transaction system that makes fraudulent activity almost impossible. Approximately nine years after the introduction of Blockchain technology, the total value of all cryptocurrencies reached an all-time high of $707 billion in January 2018 (Martin, 2018). Despite price volatility and frequent questions regarding their legitimacy (Hagen, 2018), cryptocurrencies seem set to play an ever-increasing economic role in society.

Bitcoin

Bitcoin is a digital currency. It has no physical form and, due to the decentralized blockchain technology, there is no controlling authority. Bitcoin owners can own Bitcoin anonymously and the total number of Bitcoins that can exist is limited to 21 million. Bitcoins come to existence through a process known as Mining, whereby computer processors are deployed to solve cryptographic problems with bitcoins being rewarded for solving such problems.

Mining cryptocurrencies, such as Bitcoin, is one method of obtaining digital currency. As mining becomes computationally more difficult as the total amount of currency in existence reaches it's limit, it is not a viable method for everybody. Another method for obtaining cryptocurrencies is to simply buy them from exchanges with regular currency. Which cryptocurrency exchange a person buys digital currency from depends on their location and local laws and regulations. Interestingly, as there is no central authority issuing digital currencies, their price varies according to location. If one is purchasing Bitcoin in a country where there are multiple cryptocurrency exchanges available, the price of Bitcoin from these exchanges for local currency tends to be less than the price of Bitcoin in another country with fewer exchanges or opportunities to purchase Bitcoin (Mamoria, 2018). There is therefore, no unified global price index for Bitcoin (as there is for normal currencies).

Background

Bitcoin is notoriously volatile with significant price swings within a few hours a normal occurrence. Bitcoin experienced a meteoric rise in value during 2017 reaching over $15,000 per Bitcoin in December 2017 before loosing half it's value within just 40 days of 2018 (Adkisson, 2018). Due to this volatility, having some intuition about future price trends is critical in order to make informed Bitcoin investment decisions. Figure 1 provides an overview of Bitcoin prices during the investigated time-frame of this thesis.

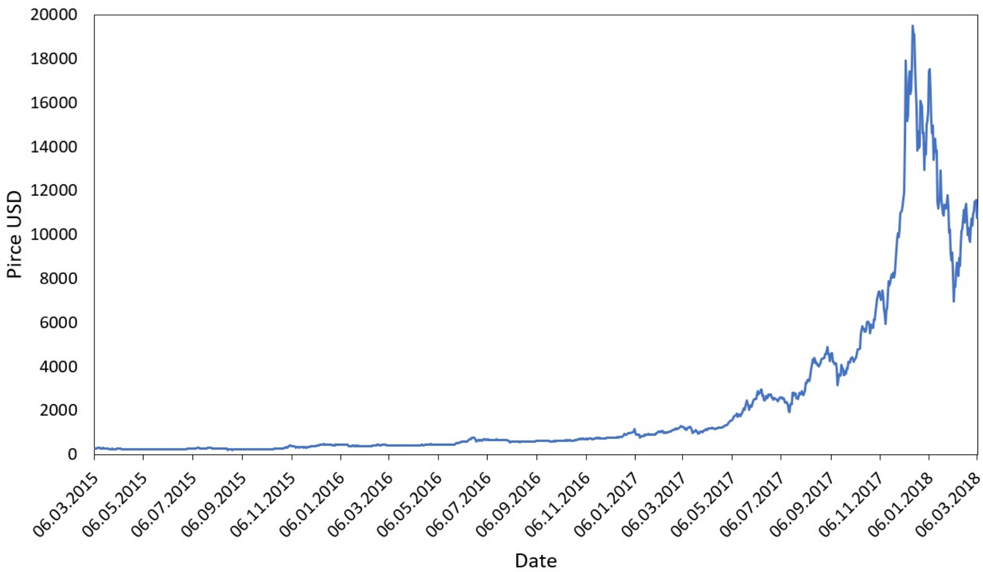

Figure 1. Bitcoin exchange rate against the USD during the investigated time-frame.

This background section will introduce literature that deals with predicting Bitcoin and other financial time-series behaviour. The presented literature has been broadly categorized into research looking at public awareness features for time-series prediction and research looking solely at economic features for prediction.

Financial time-series trends and public awareness

Kristoufek (2013) explains that Bitcoin prices cannot be adequately described by standard economic models as Bitcoin lacks the macroeconomic fundamentals of fiat currencies. As such, Kristoufek found that the sentiment of investors' plays a key role in Bitcoin's price. In a 2015 paper Kristoufek further investigated the main drivers of the price of Bitcoin outlining the main role that Bitcoin trading volume and public interest in Bitcoin have on it's price. Specifically, Kristoufek discussed how "increasing interest in the currency [Bitcoin], connected with a simple way of actually investing in it, leads to increasing demand and thus increasing prices" (Kristoufek, 2015, p.10). The author goes on to explain a long-term interconnectivity trend between Bitcoin prices and interest in Bitcoin (as measured by search frequency for Bitcoin terms in Google and Wikipedia). Garcia et al. (2014) also looked at the relationship between Bitcoin price and public awareness. Garcia et al. also used search frequency for Bitcoin terms in Google and Wikipedia as measures of public awareness. The authors complemented these features with features engineered from the volume of Bitcoin related Tweets and Facebook shares. Ultimately Garcia et al. found evidence of a strong relationship between public interest in Bitcoin and it's price. The authors also noted that Bitcoin price decline trends tend to be preceded by a peak in public awareness.

Polasik, Piotrowska, Wisniewski, Kotkowski and Lightfoot conducted a 2015 empirical inquiry into Bitcoin price fluctuations and their implications for e-commerce. In their study of Bitcoin price, the authors measure public awareness in Bitcoin both by monthly percentage increase in Bitcoin related Google searches and via monthly percentage increase in English language news articles mentioning Bitcoin. Polasik et al. also implemented tone analysis on the Bitcoin mentioning articles and used this as an additional feature in their inquiry. To supplement these public awareness features, Polasik et al. also investigated Bitcoin transaction volume as well as further macroeconomic factors proven to affect traditional stock markets. Polasik et al. present plots displaying Bitcoin price, transaction volume, newspaper mentions and google searches for each month between July 2010 and March 2014. The results are visually compelling, suggesting a clear relationship between the price and the various factors. Polasik et al. ultimately found that key drivers of Bitcoin's value are both public awareness (whereby the tone of Newspaper articles mentioning Bitcoin is significant) and transaction volume. It is important to note however that this work was only conducted on a monthly rather than daily basis.

Kim et al. (2016) attempted to predict fluctuations in cryptocurrency prices by looking at user comments and replies on cryptocurrency forums for Bitcoin, Ethereum and Ripple cryptocurrencies. Kim et al. found that the sentiment of user comments affected price fluctuations of the three cryptocurrencies in different ways. User comments with a positive tone affected fluctuations in Bitcoin prices significantly. Kim et al.'s model did not predict specific price movements (ie. rise or fall) but simply only movement (a fluctuation) associated to forum activity that occurred several days prior.

There appears to be a strong, yet complex, relationship between public awareness of Bitcoin and Bitcoin's price (this can be observed in the exploratory data analysis of this investigation by comparing Figure 1 with Figures 4 and 5). Such relationships are not however limited to cryptocurrencies, in fact; it has long been established that new information (ie. news) also affects stock prices (Fama, Fisher, Jensen and Roll, 1969). Chan (2003) explains that although there is no consensus regarding the speed with which stock prices react to news; stocks subject to public news react and move very differently to similarly performing stocks which are not subject to any news coverage. In a famous 2011 study, Bollen, Mao and Zheng found that by simply looking at public awareness and sentiment (as measured from Twitter data) it was possible to improve prediction accuracy for the Dow Jows Industrial Average over time. More recently, Guan-Ru Wu, Chieh-Tse Hou and Lin (2018) were able to demonstrate improved financial performance of a Taiwan stock market predictive model when the model was trained on a combination of public awareness features and macroeconomic features, compared to only macroeconomic features. The authors engineered public awareness features by conducting text mining of economic news relating to the Taiwan stock market.

One of the classic examples of the relationship between financial time-series and public awareness is provided by Borges, Goldstein, Ortmann and Gigerenzer (1999) who looked at the performance of portfolios of stocks put together based purely on how recognized the stocks were by people on the streets of Chicago and Munich. Over a 6 month period, a portfolio of highly recognized stocks was found to perform much better than a second portfolio of stocks largely unrecognized by the people asked on city streets. Furthermore, the portfolio put together based on the stocks well recognized by the survey participants (who were largely ignorant of the financial world) performed better than a portfolio selected at random and portfolios selected by fund managers. In summary, public awareness of financial indexes can in many cases explain and help predict the time-series behaviour of such indexes. Such insight, is not only informative but can prove to be financially profitable.

Financial time-series trends and economic features

In 2014 MIT researchers, Shah and Zhang, built a profitable Bitcoin price prediction model using Bayesian Regression. Rather than look at public awareness features, Shah and Zhang instead built their model from a dataset of 6 months' worth of short interval trading data taken from a Chinese trading exchange. The researchers put together a simple trading strategy, updated at 10 second intervals, that consisted of three outputs; either buy, sell or do nothing, based on the model's output.

Jang and Lee (2018) also looked at predicting Bitcoin activity. The authors tried to predict actual next day Bitcoin prices with Bayesian Neural Networks (BNN) using historical Bitcoin economic data and Bitcoin technical data. The authors present results whereby a BNN model returns a significantly lower test error than other benchmark models (linear regression and SVR) for Bitcoin price predictions. The results described by Jang and Lee suggest not only better performance of BNN compared to benchmark models, but also a generally very strong performance of BNN's for the price prediction task based on economic and technical data. To manage the wild price fluctuations of Bitcoin over the past 8 years (Figure 1), the authors normalized Bitcoin prices by taking the log(26) values of daily closing prices. Jang and Lee present a RMSE of 0.0039 for predicting the next day Log(26) Bitcoin price and state that their BNN model "succeeded in relatively accurate direction prediction" (p.5436). The authors did not attempt to order the predictions of their BNN model in terms of model confidence. In a similar study, Ticknor (2013) proposed an ANN with Bayesian regularization that supposedly accurately predicted actual next day stock prices. Ticknor's model took only features engineered from historical economic data over a, rather short, 700 day period. Ticknor evaluates his models in terms of test error, specifically mean absolute percentage error and presents results similar to the ARIMA benchmark model at approximately 1% to 1.5%. Ticknor however does not provide any indication of the trend prediction accuracy of the ANN model. As with the Bitcoin price prediction of Jang and Lee, predicting prices with a low test error can be said to be rather meaningless if one cannot also prove a reasonable trend accuracy. Jang and Lee, and Ticknor present graphs overlaying actual prices with predicted prices. Such graphs are misleading as they give the impression of an accurate price prediction. On closer inspection, it appears that the graphs suggest that the authors' models actually output the current day's price plus some random noise as the next day's price. This thesis will demonstrate this anomaly in the results section and demonstrate how, if an ANN can find no signal in the input data, it will revert to predicting the current day's price so as to minimize test error.

Foreign exchange (FOREX) markets are the largest financial markets in the world by volume. Predicting the relationship between two currencies is recognized as an incredibly difficult task due to the wide range of different social, political and financial aspects that can affect currency values (Ni and Yin, 2009). Kamruzzaman and Sarker describe predicting FOREX trends as "one of the most challenging applications of modern time series forecasting" (Kamruzzaman & Sarker, 2003, p.1). Despite this, Kamruzzaman and Sarker demonstrate that an ANN can predict currency pair next day trends better than a benchmark ARIMA model. The authors trained their model only on economic data and evaluated their model via both a test error measure and a directional accuracy score. Their back-propagation ANN model demonstrates a MAE of 0.0036 and a directional accuracy of approximately 80%.

Bitcoin and FOREX prediction using economic measures is a reasonably active area of research. Over the years however, the most active area of research for predicting financial time-series has been predicting stock market trends. The Efficient Market Hypothesis describes stock markets that follow a random walk manner and are, as such, impossible to predict. However, many researchers believe that markets are not fully efficient and that past economic and technical data together with Machine Learning approaches can provide profitable insight about future price movements. Kimoto, Asakawa, Yoda and Takeoka showed in 1990 that ANNs learning from a series of economic features can make profitable predictions on the Tokyo Stock Exchange. Patel, Shah, Thakkar and Kotecha (2015) looked at a wide variety of machine learning techniques, including fusing two different techniques together, to predict two stock indexes over different time intervals. A big data approach was taken whereby 10 years of historical economic data was taken for the two indices being investigated. From daily low, high and opening prices, ten technical indicator features were created. The output of the model was the stock's index either 1-10, 15 or 30 days in the future. Ultimately Patel et al. concluded that it was easier to predict stock indices a day or two into the future than indices further in the future. Out of numerous techniques used to build their models, a SVR method used in a cascade with an ANN was the best predictor.

K. Kim (2003) demonstrated that support vector machines can be used to predict next day stock price index movement with an accuracy of up to 57% on test data. In this particular study, a SVM outperformed an ANN (with a single hidden layer) for the same task. As input features, the author chose 12 technical indicators calculated from stock index opening, closing, high and low data. K.Kim's 2003 paper has inspired numerous similar investigations predicting next day stock index movements, comparing SVMs and ANNs for this task. In a 2008 report, Senol and Ozturan used an ANN to predict daily stock trends for a series of stock indexes reporting an average accuracy of around 77%. The authors used a single hidden layer ANN architecture with 11 neurons and 5 technical indicator features calculated from historical economic data. Kara, Boyacioglu and Baykan followed up this work a few years later (2011) with an investigation predicting the next day stock price index of the Istanbul stock exchange. The authors collected 10 years worth of daily economic data for the Istanbul stock exchange and chose 10 technical indicators as input features calculated from the data. Kara et al. then carried out an extensive grid search for the best ANN and SVM hyperparameters. The ANN architecture was kept similar to that of the ANN used by K. Kim, a single hidden layer with a low number of neurons, Ultimately the authors reported a trend prediction accuracy of 75% for their ANN model and 71% for their SVM model. Similarly, Qiu and Song (2016) also found than an ANN model, optimized using genetic algorithms, was the best for predicting stock index trends of the Nikkei 225 index. The authors found that an ANN, inputted with 10 technical indicator features, could predict daily stock index trends with an accuracy of 81%. The more recent work of Kara et al. (2011) and Qiu and Song (2016) suggests that currently ANNs are better at performing such trend predictions than SVMs. This seems logical, as computer processing power has increased exponentially since K. Kim's 2003 publication. Such increased processing power means that ANNs can nowadays handle much more data and process complex hyperparameter grid searches much faster than in the past.

It should be noted that Pyo, Lee, Cha and Jang (2017) were more critical of the ability of ANNs trained only on economic data to accurately predict stock market trends. Pyo et al. attempted to recreate some of the results found by other researchers, including those of Kara et al. The authors used a similar approach, ten technical indicator features, extracted from historical data, and an ANN with a single hidden layer. Instead of the Istanbul stock exchange, the authors investigated the Korean composite stock index 200. Ultimately the authors were not able to get results anywhere close to the accuracy of other papers. They reported 50.23% trend prediction accuracy for daily trend movements of the Korean stock index with an ANN.

Artificial Neural Networks

ANNs, also known as Multi-Layer Perceptrons, are the "quintessential deep learning model" (Goodfellow et al, 2016, p.164). ANN's consist of an input layer with a node for each input feature, a single or multiple hidden layer(s) each with a specified number of neurons, and an output layer (Figure 2). Importantly, all nodes within an ANN are linked together and each of these connections is given a weight, w. Given input, x, an ANN's goal is to map x to a target value, y. ANN's achieve this goal by learning the best values of weights, w, to solve the function: $y = f(x, w)$ for given x and y pairs. Due to the large number of connections and weights that can be optimized, ANNs can learn complex non-linear relationships between input features and make accurate predictions for complicated tasks.

The term *Network* in Artificial Neural Networks arises from the fact that each layer in an ANN can be given a different function. These layers put together thus form a network. Specifically, for a three layered ANN: $f(x) = f^1(f^2(f^3(x)))$ (Goodfellow et al, 2016, p.164). The depth of an ANN describes how many layers it consists of.

The learning procedure proceeds via the ANN attempting to minimize a specified loss function as it outputs a point estimate $y\prime$ value for a corresponding input, x and checks this $y\prime$ value against the true y value. When predicting numerical values such as prices, RMSE would be a possible loss function by which an ANN learns. For binary prediction problems, a binary cross entropy loss function would be appropriate. Minimizing loss functions is achieved by optimizing weights and biases via backpropagation.

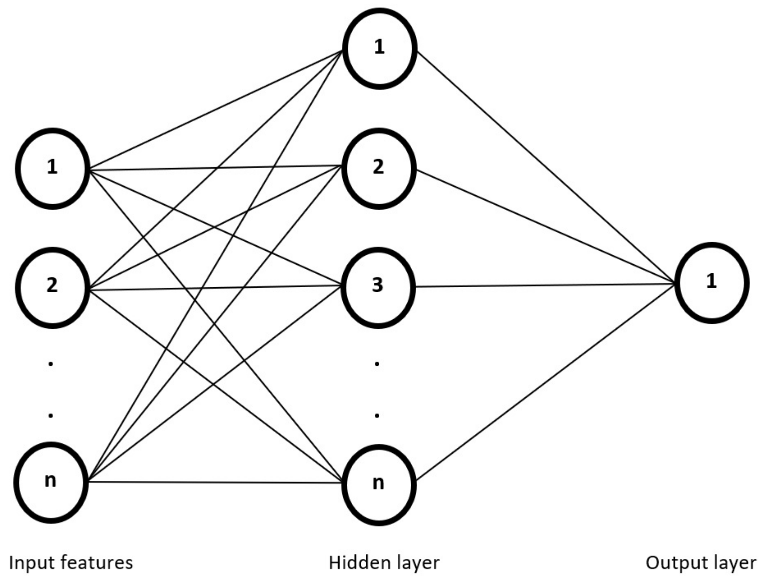

Figure 2. an Artificial Neural Network

ANN without Dropout; all neurons are connected.

Bayesian Neural Networks

ANNs make point estimates for $y\prime$ values that do not convey any uncertainty information. ANN outputs can be evaluated by various measures but these measures tell us nothing of the confidence of a model's predictions. Having an understanding about the confidence of predictions can be very valuable. Such knowledge allows for a better and more robust assessment of models. For a binary classification ANN which returns an accuracy score similar to the majority class, understanding the confidence of the model's predictions allows one to discern whether the model outputs useful information or is simply mimicking the majority class. Furthermore, a binary classification BNN model that predicts financial trends with only 50% accuracy yet provides a confidence measure for each prediction can in theory be just as profitable for a fund manager as an ANN with a much greater accuracy score but no uncertainty information.

In a detailed introductory blog on BNNs, Gal (2015) explains how Bayesian techniques can be allied to deep learning approaches but that such approaches prove to be computationally very demanding. In a 2016 paper, Gal and Ghahramani introduce using Dropout in neural networks as a Bayesian approximation. The authors demonstrate that by ensuring a Dropout layer is used during the prediction phase of a neural network, uncertainty information approximate to a Bayesian approach can be ascertained (Figure 3).

Srivastava, Hinton, Krizhevsky, Sutskever and Salakhutdinov first introduced Dropout in a 2014 paper. The author's explain how, due to the number of connections in ANNs, such networks are often prone to overfitting (Geman, Bienenstock, & Doursat, 1992). Dropout is suggested as a new regularization approach that outperforms other methods of regularization. Dropout occurs during the training phase of a network whereby a specified percentage of neurons and their connections (selected at random) are told to 'drop out'. The constant dropping out of various connections during the training phase prevents certain connections from overfitting the data, improving generalization performance.

Programming a neural network to also perform Dropout during a prediction task is the Bayesian approximation described by Gal and Ghahramani in 2016. Not only is this method more computationally efficient than running a normal bayesian neural network, it means that uncertainty information can be obtained from an ANN with practically no changes to it's structure (Gal, 2015). Typically, a BNN will be run more than 30 times, returning a range of predictions. The average prediction for each instance can then be calculated as well as the standard deviation of all predictions. This standard deviation value is the measure of prediction confidence.

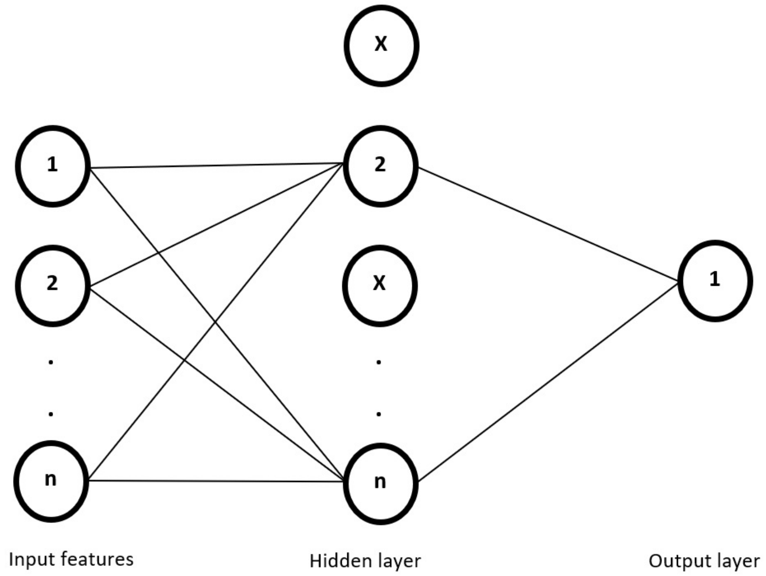

Figure 3. Artificial Neural Network with Dropout

For a Bayesian approximation network, the Dropout is active in both the training and testing phase of the model.

Approach

This background section has described how many researchers have had success explaining and predicting financial time-series using only economic measures (Kamruzzaman & Sarker, 2003; Kimoto et al., 1990; Patel et al., 2015; Shah & Zhang, 2014). It has also been explained how public awareness features can be used to either complement economic features in predicting financial time-series (Guan-Ru Wu, Chieh-Tse Hou and Lin, 2018), or be used in isolation to predict and explain financial time-series trends (Bollen, Mao and Zheng, 2011; Borges, Goldstein, Ortmann and Gigerenzer, 1999; Garcia et al., 2014; Kim et al., 2016). Based on this intuition and the knowledge that public awareness is a driver of Bitcoin price (Kristoufek, 2013, 2015) it was decided that both economic and public awareness features should be compiled for the Bitcoin price trend prediction ANN and Bayesian approximation models. These economic and public awareness features will be used throughout this investigation. Eight economic features will be extracted from historical Bitcoin economic data. The three public awareness features used are; a measure of daily Google search frequency for the term ’Bitcoin’, a measure of daily international news articles with the term ’Bitcoin’ in the title, and a mean sentiment tone measure for all the articles on a specific day.

This background section has been critical of how some researchers (Jang & Lee, 2018; Ticknor, 2013) chose to evaluate the performance of their financial time-series prediction models. This thesis sets out to demonstrate such pitfalls and explain what additional evaluation criterion are required to prove the robustness of results. To this end an ANN trained on economic and public awareness features, tasked with predicting next day actual Bitcoin prices (continuous variables) is created. It will be shown that networks predicting next day Bitcoin prices can achieve low RMSE and MAPE scores, yet fail to predict trend accuracy any better than random guessing.

This thesis will investigate whether an ANN can predict next day Bitcoin trends with an accuracy score better than the majority class (for a specified test set). An ANN was selected for this task due to their proven ability to solve non-linear financial next day trend prediction tasks (Kara, Boyacioglu and Baykan, 2011; Qiu & Song, 2016;

Senol & Ozturan, 2008). The ANN will be trained on the same set of features as described above. A Logistic Regression model for the same task will also be run to compare performance with the ANN. Ultimately, the accuracy score of the ANN was found to be inferior to that of the majority class over the test set.

To solve the problem of ordering daily predictions in terms of accuracy; a novel approach of using the Dropout Bayesian approximation will be used (Gal & Ghahramani, 2016). In the research done for this thesis no previous study appears to have been conducted with regards to prediction ordering for Bitcoin trend prediction. The Bayesian approximation network approach is used as it's more computationally efficient than a normal BNN and it allows for the same ANN architecture to be used. Thus, with only a few changes to the ANN architecture employed in the previously described task, prediction ordering based on confidence can be achieved. Results will demonstrate that as the confidence of the Bayesian approximation network predictions increases, so too does accuracy. This effect is most pronounced on a Bayesian approximation network trained only on public awareness data. This is both an exciting and unique contribution to Bitcoin price trend research.

The importance of the economic and public awareness features groups will be compared. This will be done by running two Bayesian approximation networks, one trained only on economic features and the other trained only on public awareness features. When comparing the results it was not possible to categorically determine which group of features was most important to the Bayesian approximation model. A custom investment strategy is created (where investments are only made on specific days) and ultimately it was found that a Bayesian approximation model trained only on public awareness features is the most potentially profitable model. This is an interesting and novel finding of this research.

Method

Data

Public awareness data

The GDELT 2.0 database is an open source database monitoring and cataloging global events and news since February 2015. GDELT machine translates a vast amount of all global news coverage into English and updates itself every 15 minutes. Queried via Google Big Query, GDELT can provide a list of news stories containing certain keywords on certain days as well as an average sentiment value for such news stories (Gdelt, 2015a).

Google Big Query was used to access the GDELT 2.0 Database. An SQL query was run on the GDELT 2.0 Events database, searching for all news headlines containing the keyword ‘Bitcoin’ between February 20 2015 and March 06 2018 (UTC time) in the ‘SOURCEURL’ field. A starting date of February 20 2015 was chosen as this is when the GDELT 2.0 Database went live. Although the GDELT 1.0 Database covers events since 1970, for this project it was decided to work with data from GDELT 2.0 as GDELT 2.0 contains news headlines machine translated from 65 different languages (Gdelt, 2015a) and therefore reflects global public awareness in Bitcoin rather than western-centric awareness only.

A measure for public interest in Bitcoin was calculated by taking the total number of news headlines containing the relevant keyword for each day. The GDELT 2.0 Events database also contains an average tone value for each event/ news story. This value is described by the GDELT Event Database Data Format Codebook V2.0 (2015b) as reflecting the average tone of a new article. An average tone of 0 reflects a neutral event with negative tones >= -10 reflecting a slightly negative event and positive tones <= 10 reflecting a slightly positive event. The measure is given as a rudimentary tonal assessment of events. The mean tone of all Bitcoin related news articles on a given day was calculated as a feature reflecting public sentiment.

The daily frequency of ’Bitcoin’ keyword searches on Google made up an additional public awareness feature. Data for this feature was obtained from Google Trends (https://trends.google.com/). Google Trends returns a graph and optional .CSV file detailing a normalized global search frequency value for a specific search term during a specified time frame. Unfortunately, Google Trends only provides daily search frequency values for time frames up to 90 days. For longer time frames, only weekly search frequency values can be obtained. As search frequency values are automatically normalized by Google Trends within the specified time frame (to a value between 0-100), it was not possible to simply paste together daily search frequencies for the full time frame in 90 day blocks. Instead, daily search frequency values between February 20 2015 and March 06 2018 were collected in 90 day blocks. Weekly search frequency values were also collected between this time frame, importantly the weekly search frequencies were relevantly normalized to the time frame under investigation. The daily search frequency values were then normalized using the weekly search frequency values according to a method provided by Johansson (2014). This method provides daily search frequency values relevantly normalized for the specific February 20 2015 to March 06 2018 time frame, rather than the strict 90 day time times provided by Google.

Economic data

Cryptocurrencies, such as Bitcoin, have no unified international price (Mamoria, 2018). To build a model that generalizes well to global cryptocurrency prices, not just to a specific country or exchange market, cryptocurrency economic data was taken from the Coin Market Cap website (www.coinmarketcap.com). Coin Market Cap provides historical transaction data for cryptocurrencies against the USD. Coin Market Cap explains in it's FAQ section (https://coinmarketcap.com/faq/) that cryptocurrency prices are calculated by finding the mean of all prices across all listed global cryptocurrency markets. The data is thus indicative of Bitcoin prices globally.

Data extracted from this site included the daily opening, closing, high and low prices for the time frame under investigation. Coin market cap provides open/close prices at UTC time. An overview of all features is presented in Table 2.

Technical indicators

As described in the previous section, numerous researchers have had success predicting financial time series using technical indicators as input features. Technical indicators are features derived from performing mathematical functions on historic price data (Murphy, 1999). Technical indicators play an important role in finance industries and are at the heart of many an investment strategy. Eight technical indicators were selected for this project based on their use and success in similar prediction challenges. The technical indicator formulas are provided in Table 1. In addition to 8 technical indicators, the current day's closing will also be used as an economic feature.

Table 1

Technical Indicators

Technical Indicator	Days (n)	Formula
Relative Strength Index	14	$100 - \frac{100}{1+(\sum_{i=0}^{n-1} Up_{t-i}/n)/(\sum_{i=0}^{n-1} Dw_{t-i}/n)}$
Williams %R	14	$\frac{H_n - C_t}{H_n - L_n} \times 100$
Stochastic %K	14	$\frac{C_t - LL_{t-n}}{HH_{t-n} - LL_{t-n}} \times 100$
Stochastic %D	4	$\frac{\sum_{i=0}^{n-1} Stochastic\%K_{t-i}}{n}$
Slow %D	4	$\frac{\sum_{i=0}^{n-1} Stochastic\%D_{t-i}}{n}$
Momentum	4	$C_t - C_{t-n}$
Rate of Change	12	$\frac{C_t}{C_{t-n}} \times 100$
AD Oscillator	-	$\frac{H_t - C_{t-1}}{H_t - L_t}$

relative-strength index 'Up' refers to an upward trend. 'Dw' to a downward trend. C is closing price on a specific day, t. L is lowest price on day, t. H is highest price on day, t. LL is lowest low price in a range of days. HH is highest high price in a range of days

Table 2

Features

Feature	Feature Type
Volume of news articles	Public awareness
Average news tones	Public awareness
Google searches	Public awareness
Closing price	Economic
Relative strength index	Economic
Williams %R	Economic
Stochastic %K	Economic
Stochastic %D	Economic
Slow %D	Economic
Momentum	Economic
Rate of Change	Economic
AD Oscillator	Economic

Missing data

Despite inspecting the database several times via Google Big Query, GDELT 2.0 returns no news headlines containing 'Bitcoin' for the 29 August 2017. Thus, the GDELT summary tool was used to inspect results for this day in more detail (https://api.gdeltproject.org/api/v2/summary/summary). GDELT summary allows users to search for key words found in global news articles for any day or time-period within the last year. It is therefore a simpler, less comprehensive yet more intuitive alternative to using Google Big Query. GDELT summary returns all news stories containing a search term for a specific day or time-period as well as data regarding the average tone for all articles. It was possible to use this service to find the number of 'Bitcoin' related news headlines and the mean average tone of all these news stories for the 29 August 2017. Interestingly, the GDELT team posted a blog (https://blog.gdeltproject.org/tracking-bitcoin-cryptocurrencies-blockchain-using-gdelt-summary/) explaining the functionality of GDELT summary using Bitcoin and cryptocurrencies as an example. Unfortunately, this service is only available for the past year.

Target label

The aim of this work is the accurate prediction of either next day price rises or falls of Bitcoin. The target labels for the datasets are calculated by finding the difference between the current day's closing cryptocurrency price and the next day's closing price. If the difference is positive (i.e a price increase) the target label is 1. If it is negative the target label is 0.

As this thesis also looks at using models to predict a continuous target label in addition to the binary label, an additional target label is created. This target label is the log(10) value of the next day's closing price. The logarithmic value of the closing price is taken to negate the fact that the price of Bitcoin has greatly increased during the investigated time-frame. Using the log(10) value will mitigate some of the wild increases. A similar approach was used by Jang and Lee (2018).

Feature preparation

For all models the features and y target labels are split into an 85:15 training, test split. Sklearn's standard scaler is used to scale the data. An 85:15 train test split ratio was chosen to reflect the fact that the price of Bitcoin increased greatly in 2017 and it is important that this increase is reflected in the training data. This split equates to a training set of 931 days and a test set of 165 days. The training set runs from 6 March 2015 until 21 September 2017. The test set runs from 22 September 2017 until 5 March 2018. Ultimately a larger test set would ensure a more robust model but capturing a good portion of Bitcoin's 2017 volatility in the training data was deemed more important. As this is time series data an n-fold cross validation approach was not used.

ANN price prediction model

This model aims to predict the next day log(10) Bitcoin price and adequate evaluation criterion are to be investigated to understand if RMSE and MAPE alone can adequately describe the model's performance. The architecture of this model is informed by a blog from Heinz (2017) entitled 'A simple deep learning model for stock price prediction using TensorFlow'. This model is made up of 5 layers each with 1024, 512, 256, 128 and 1 neuron(s) respectively. There is 20% dropout after each layer (excluding output layer) and the 'relu' activation function is used for all 5 layers. The model is optimized by the Adam optimizer (lr = 0.001) and run for 100 epochs with a batch size of 1.

No in depth hyperparameter grid search or model optimization was carried out for the ANN price prediction model. This is because this investigation was only tasked with explaining some common pitfalls with next day actual price prediction evaluation. As will be explained in the discussion section; the results of such models are not of much use and therefore what specific architecture is used is not important.

Logistic Regression classification model

This model aims to predict next day Bitcoin price trends better than majority class. A standard Logistic Regression model is trained, without any regularization, on all input features.

ANN classification model

Should the logisitic regression model not be sufficient, this model aims to predict next day Bitcoin price trends better than majority class. An ANN classifier is built using KERAS functional API. The ANN has three layers. An input layer with shape equal to the number of input features. One hidden layer with 28 neurons and 'tanh' activation function. An ouput layer with one neuron and a 'sigmoid' activation function. The ANN is optimized by the 'adam' optimizer with a learning rate of 0.001, minimizes the binary cross entropy loss function, and runs with a batch size of 1 for 100 epochs. There is a 20% dropout layer between the hidden and output layers.

A grid search was carried out for a range of neuron values between 28-500 and for two different hidden layer activation functions ('tanh' or 'relu'). In all cases though the train test accuracy of the network appeared to change little, a simpler architecture (of 28 neurons only) was therefore decided on. A sigmoid function outer layer was selected due to it's relevance for the binary classification task. The output of this model is a value mapped to minus infinity and plus infinity by the output layer's sigmoid function. A '1' label represents a mean prediction of >= 0.5, a '0' < 0.5. The accuracy score is then reported.

Bayesian approximation classification model

The aim of this model is to provide predictions that can be ordered in terms of the model's confidence in each prediction. Using dropout as a Bayesian approximation allows the same ANN model to be used, with only a few modifications. The ANN model already contains a dropout layer. The KERAS functional API includes an option for this dropout layer to be active in both training and testing phases. For the Bayesian approximation classifier model, this option was switched on. The output of the model is the output of the outer layer's sigmoid function (a value between minus infinity and plus infinity). Importantly, the standard deviation values of predictions are the measure of the model's prediction confidence

Feature importance

The importance of the public awareness and economic features will be assessed via a feature ablation approach using the Bayesian approximation network. The same model will be run numerous times, each time with a group of features removed. The results will be reported in terms of the accuracy score of the model without a particular group of features. A drop in accuracy for each removed feature set serves as a measure of the importance of that feature group in the prediction task. Investment performance of the Bayesian approximation predictions will be reported to help quantify feature importance.

Evaluation criteria

Accuracy score for correct prediction of either price rises or falls is the main evaluation criteria of this investigation. Confusion matrices for the classification models are reported in the appendix. The majority trend of the time series will serve as a benchmark.

This project will also investigate RMSE and MAPE and their usefulness as evaluation measures. RMSE and MAPE will only be used for the model attempting to predict the log(10) actual value of the next day's closing price (i.e a continuous variable rather than a binary one). An accuracy score is also calculated for this model.

$$RMSE = \sqrt{\frac{\Sigma_{i=1}^{n}(y\prime_i - y_i)^2}{n}} \tag{1}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n} abs(\frac{y_i - y\prime_i}{y_i}) \tag{2}$$

The standard deviation of daily predictions from the Bayesian approximation model is an evaluation of the uncertainty of the model's mean prediction. These confidence scores can then be listed in order of uncertainty via ordering from lowest to highest standard deviation. An investment returns function was programmed to evaluate these results in financial terms. $1000 worth of Bitcoin is bought on the day of each prediction and then sold the next day. For price fall predictions, the negative value is taken to imitate an arbitrage position. The investment performance is assessed for the first 10 days with the lowest prediction uncertainty, the first 20 days and first 50 days. This evaluation is useful as it evaluates not only whether or not the prediction is correct, but also it evaluates the magnitude of price rises or falls which the model can predict confidently. For example, input features that confidently predict rises and falls associated with relatively large price swings would be very useful from a financial perspective. The investment performance function allows different models to be assessed comparatively. This investment performance is also compared to a totally naive and a totally random performance strategy to quantify how insightful each model is.

Exploratory data analysis

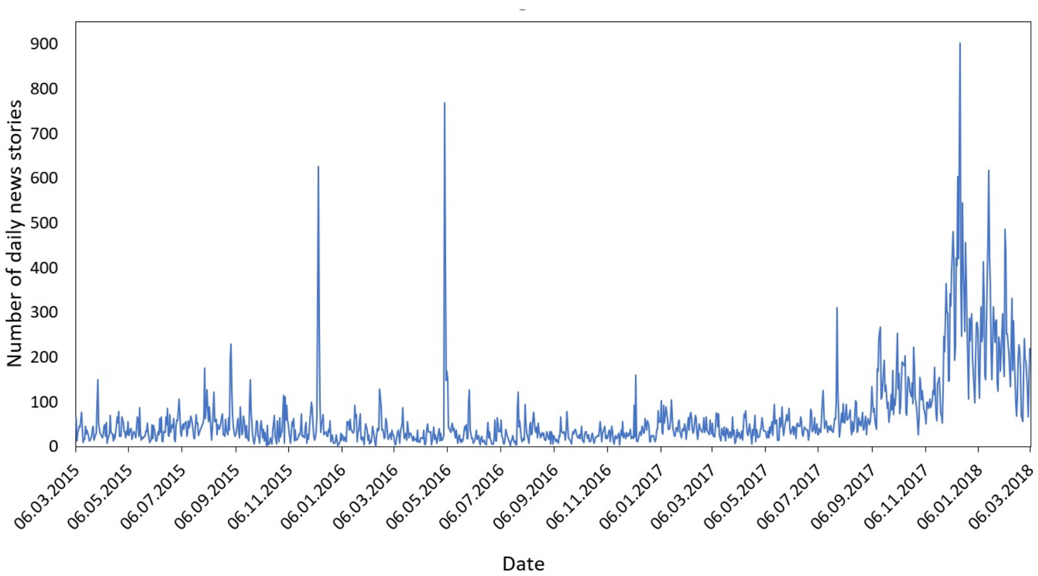

Figure 4. Number of news articles per day with Bitcoin in the title

The unnormalized GDELT data for number of news stories per day seems to consist of a few outliers. Generally speaking however an increase in news stories seems to mirror the increase in Bitcoin price as seen in Figure 1.

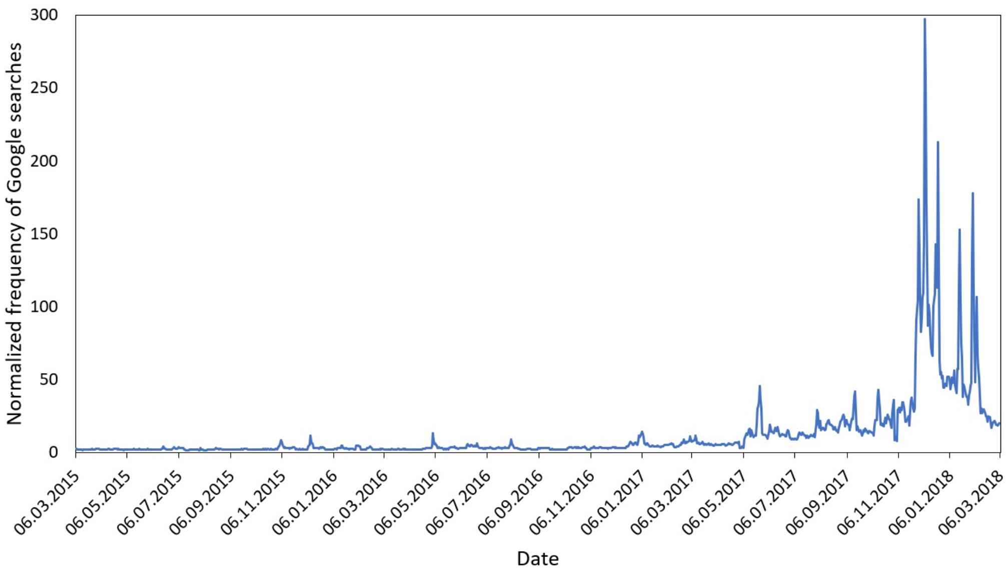

Figure 5. Daily (normalized) frequency of Google searches containing the word 'Bitcoin'

The trend of frequency of Google searches for 'Bitcoin' appears to mirror the trend in Bitcoin price as seen in Figure 1. There is a clear relationship between Bitcoin price and public interest in Bitcoin.

Table 3

Pearson correlation coefficients of features and next day's closing Bitcoin price

Feature	Correlation coefficent
Current day's closing price	0.997
Number of news stories per day	0.743
Average news tone	-0.13
Google searches	0.781
Relative Strength Index	0.026
Williams %R	0.061
Stochastic %K	0.061
Stochastic %D	0.065
Slow %D	0.067
Momentum	0.144
Rate of Change	0.226
AD Oscillator	-0.009

The linear relationship between public awareness features (number of news stories per day and Google searches) and Bitcoin's next day closing price is confirmed by this table. Linear correlations between next day's closing price and the technical indicators is however not obvious. The average news tone features also does not appear to strongly correlate linearly.

Results

ANN price prediction

An ANN price prediction model was created with 12 input features and tasked with predicting next day's log(10) closing price. The aim of this model was to assess whether RMSE and MAPE alone are sufficient to evaluate financial time-series.

Table 4 outlines the results of the ANN price prediction model. The model demonstrates RMSE and MAPE values that, when considered in isolation from the accuracy score and together with Figure 6, appear predict the next day's log(10) price relatively reasonably.

Table 4

Results of ANN log(10) Bitcoin next day closing price prediction

Evaluation	Result
RMSE	0.1875
MAPE	4.35%
Accuracy	55.5%

Ticknor and Jang and Lee present similar figures as Figure 6. Plotting actual versus predicted prices. Such figures, together with RMSE and MAPE values (which Ticknor and Jang and Lee report as being much lower than those found in this investigation), suggest that the models are capable of predicting price trends.

On closer inspection of Figure 6 however, it can be observed that the prices predicted by the ANN follow the price trends with a lag of approximately one day. It seems that the ANN is simply returning a value similar to the current day's price rather than making an intelligent prediction for the next day's price. The trend prediction accuracy for this model (55.5%) is only slightly better than random guessing and inferior to predicting the majority trend (a price increase, 59.4%). If the trend prediction results are shifted back by one day the trend prediction accuracy increases to 75.5%, further evidence that the model is simply producing values similar to the current day's log(10) closing price.

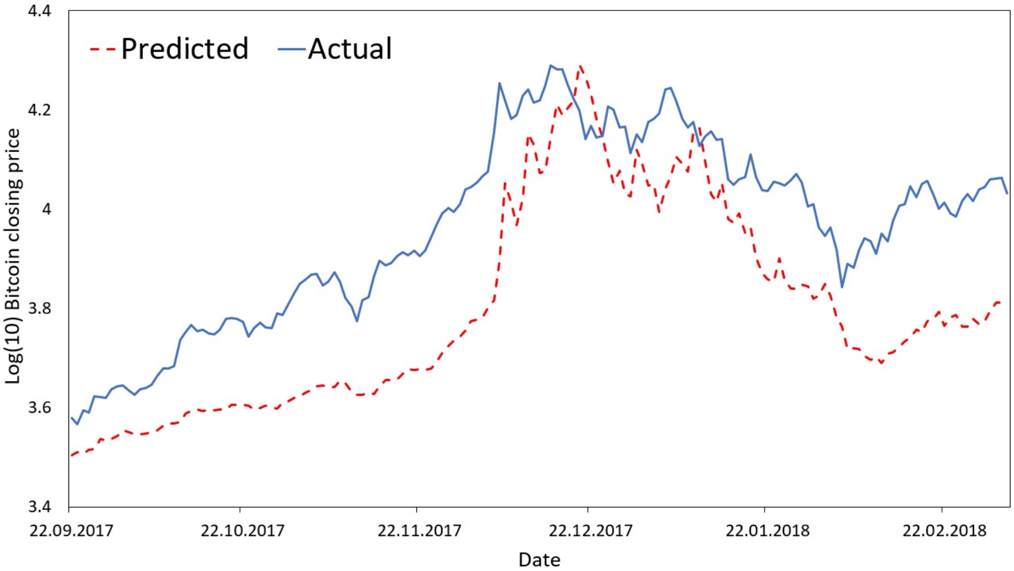

Figure 6. Bitcoin price prediction by ANN for 165 day test set

The ANN model at first glance appears to predict the actual price trend very well. On closer inspection however, it is obvious that there is a lag.

Logistic regression classification model

A logistic regression model, built to predict the daily rise fall trend of next day Bitcoin price, was trained on all 12 features. The accuracy score was poor, only 53.3%.

ANN classification model

An ANN classification model was built for next day Bitcoin price trend binary classification prediction. The aim of this model was to predict Bitcoin price trend with an accuracy score better than the majority class of the test set.

This model was inputted with 3 public awareness features and 8 technical indicators plus the current days closing price (12 features in total). The target label for this model was either a '1' for a prediction that tomorrow's closing price will be above the current day's or a '0' if tomorrow's price is below the current day's. The training set consisted of 931 training examples. The test set for the ANN classification model consisted of 165 daily price movements. 98 movements were rises (ie. '1') and 67 were falls ('0'). The majority trend therefore of the test set is a price rise, 59.4% of the time. The results from the ANN classification model are reported in Table 5. It is clear that the ANN classification model performs poorly, only slightly better than random guessing.

Table 5

Results of ANN Bitcoin daily rise fall trend classification

Evaluation	Result
Accuracy	53.3%
Correct rise predictions	64.3%
Correct fall predictions	37.3%

Bayesian approximation classification model

A Bayesian approximation network is created from the ANN architecture via the Dropout technique (Gal & Ghahramani, 2016). Essentially, this means ensuring that the 20% Dropout layer remains active during the prediction phase of the model. The aim of this model was to see, when next day Bitcoin price trend predictions are ordered in terms of the network's confidence in each prediction; does prediction accuracy increase with prediction confidence?

Results are presented in Table 6. The Bayesian approximation predictions were very similar to that of the ANN. However the mean standard deviation of all the prediction sigmoid function results (for 30 runs of the model) was 0.043; almost double the mean standard deviation of all ANN results (0.023). The minimum standard deviation of the Bayesian approximation results was 0.0249 and the maximum was 0.0668.

Table 6

Results of Bayesian approximation Bitcoin daily rise fall trend classification

Evaluation	Result
Accuracy	52.7%
Accuracy rise predictions	64.3%
Accuracy fall predictions	35.8%

The Bayesian approximation classification results appear no more useful than the ANN classification results. However, if the standard deviation results are looked at in more detail (whereby standard deviation is a proxy for prediction uncertainty), the results become more intriguing. The lower the standard deviation of the predictions, the lower the uncertainty of the prediction. It was found that when looking at predictions with the lowest standard deviations (i.e the model is most certain about these predictions) the trend prediction accuracy score increases. This is illustrated in Figure 7. Essentially, when ordered by uncertainty, the 50 most certain predictions by the model are more accurate than a majority class prediction over the whole test set. (62.0%

compared to 59.4%). Furthermore, the 10 predictions with the lowest uncertainty (standard deviation <= 0.03) have 90% accuracy (8 out of 8 correct rise predictions and 1 out of 2 correct fall predictions). Predictions with a standard deviation <= 0.0283 were correct 5 times out of 5. All of these predictions were rise predictions.

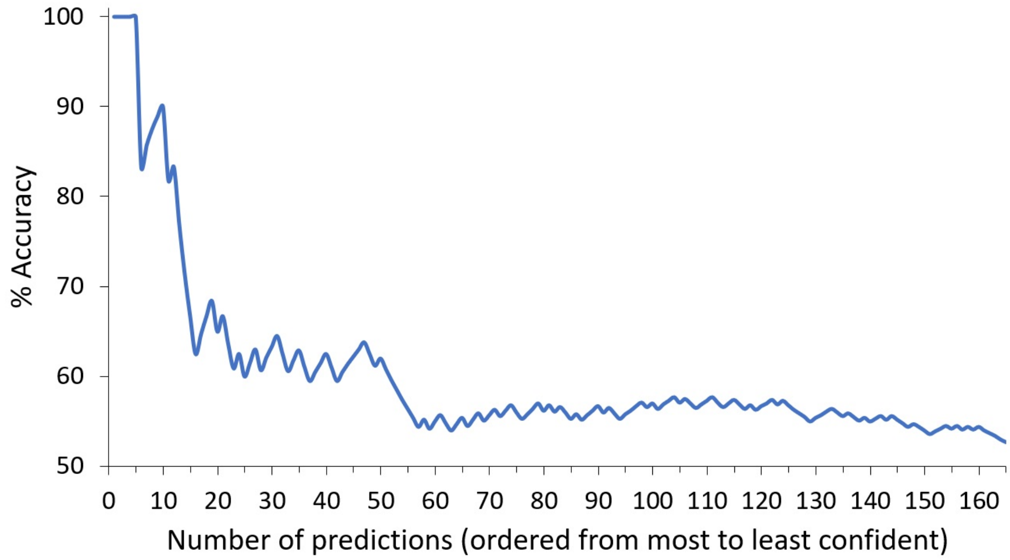

Figure 7. Predictions ordered by confidence for Bayesian approximation network trained on all economic and public awareness features

Feature Importance

The training features of the Bayesian approximation network can be grouped into economic features and public awareness features. Which of these features is most important to the Bitcoin price trend prediction task? Table 7 reviews the results of the feature ablation approach. A Bayesian approximation model fed economic features only scored the highest accuracy, 57.0%. As the accuracy of all three feature ablation models was below the majority class accuracy (59.4%) it was decided to look a little closer at the results by ordering the predictions in terms of the Bayesian approximation models' confidence.

These results are demonstrated in Figure 7 for all features, Figure 8 for economic features only, and Figure 9 for public awareness features only. By selecting the 50 predictions which the Bayesian approximation model is most confident about, we can make a comparison between the models fed with the different features. The model fed with all features scores 62%, the model with economic features only scores 58.3%, and most surprisingly, the model fed with only public awareness features scores 66%.

Table 7

Results of BNN Bitcoin classification model

Features	Accuracy
Public Awareness and Economic	52.7%
Economic only	57.0%
Public Awareness only	55.8%

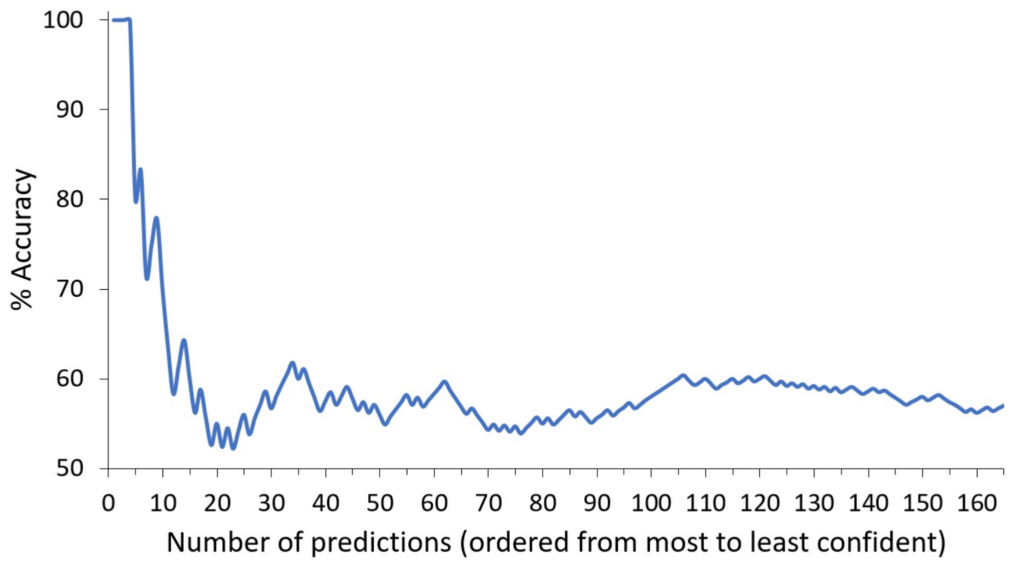

Figure 8. Predictions ordered by confidence for Bayesian approximation network trained only on economic features

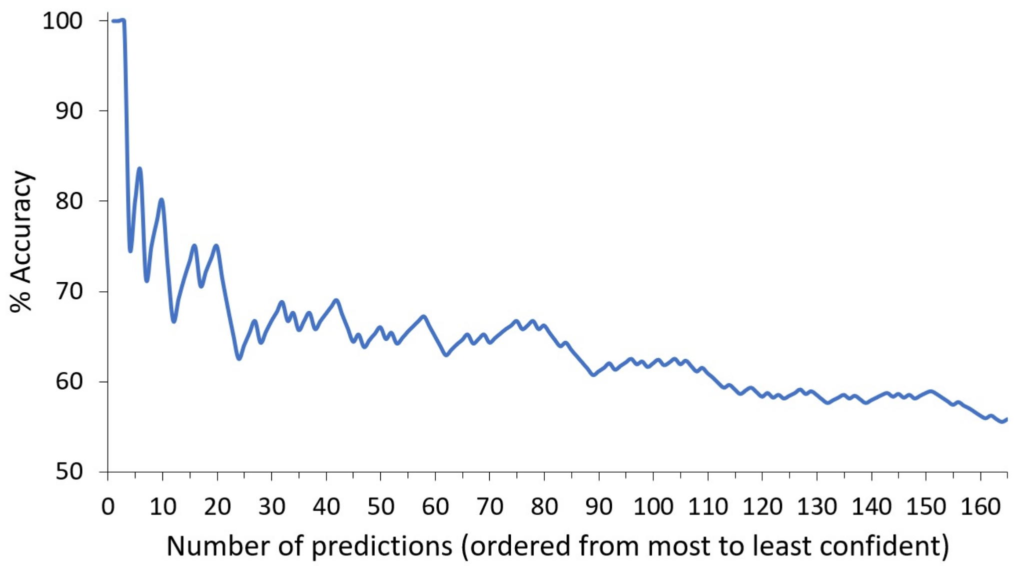

Figure 9. Predictions ordered by confidence for Bayesian approximation network trained only on public awareness features

Investment performance of the different feature groups

To evaluate feature importance more closely the investment performance function described in the Methods section is deployed. As a baseline for this evaluation a simple investment strategy of buying $1000 of Bitcoin for the first 10 days of the test set (and selling the next day) would return a profit of $202. This naive approach returns $423 for the first 20 days and $598 profit for the first 50 days. Table 8 shows the results of investment returns for the Bayesian approximation model with the different input features.

To better understand whether the Bayesian approximation models are offering genuine insight with regards to the 50 most confident predictions; an additional investment function was programmed. This program selects 50 days at random from the test set, assumes a next day price rise and follows the same investment procedure as before. This process is carried out 100,000 times and the overall profit is averaged. The average profit of this random investment strategy assuming always price rise over the test set is $445.

Table 8

Investment returns for 10, 20 and 50 most confident Bayesian approximation model predictions

Input features	Profit top 10	Profit top 20	Profit top 50
Economic and public awareness	$401	$279	$571
Economic only	$434	$146	$752
Public awareness only	$178	$421	$964

Discussion

ANN price prediction

- *Is evaluating financial time-series prediction with only RMSE and MAPE criterion sufficient?*

Ticknor (2013) used a Bayesian regularized artificial neural network to predict next day stock indexes. Ticknor reported a MAPE of 1.05% for predicting the Microsoft index and a MAPE of 1.33% for the Goldman Sachs index. Jang and Lee (2018) used a BNN to predict the next day's log(26) closing price of Bitcoin. The authors reported a MAPE of 0.0138% and a RMSE of 0.0039. This investigation reported much higher results (RMSE 0.1875 and MAPE 4.35%).

The ANN price prediction model was not able to achieve RMSE and MAPE scores similar to Ticknor (2013) or Jang and Lee (2018). It is clear from Figure 6 that the ANN price prediction model is consistently under predicting the log(10) Bitcoin price. This ANN model was inputted with public awareness and economic features whereas Jang and Lee fed their model features related to technical aspects of Bitcoin. Jang and Lee's model was run on data up to only August 2017 and thus did not experience the volatility in the latter part of last year.

However, obtaining low RMSE and MAPE scores was not the aim of this model. Rather, these results aim to highlight the trouble with evaluating financial time series models using only RMSE or MAPE in isolation. The RMSE and MAPE scores for the ANN price prediction model together with Figure 6 suggest a model, although not as accurate as those presented in the literature, that is learning and making informed price predictions. At first glance, Figure 6 suggests model predictions that follow the actual time series trend quite well. Indeed this impression would be even more pronounced with reduced RMSE and MAPE scores. However, by presenting an accuracy score (Table 4) of 55.5% it is possible to make a more robust critical evaluation of the model. If a model cannot predict the rise and fall trend of a financial time series meaningfully better than random guessing or majority trend (price rise, 59.4%) then it is obvious that despite reasonable RMSE and MAPE scores; the deep learning model is not learning anything. Armed with this intuition, when examining financial time series plots such as Figure 6, it becomes obvious that such graphs are misleading. The model actually appears to simply be returning a value close to the current day's log(10) price rather than making any intelligent prediction.

Financial time series models that claim to be able to accurately predict continuous variables should not report RMSE or MAPE values in isolation but also a measure of trend prediction accuracy. A model that claims to be able to predict a stock price or Bitcoin price to a high degree of accuracy ought to be able to also accurately predict a, comparatively simple, price rise or fall trend with a high degree of accuracy as well. From a hedge fund or financial manager's perspective, knowing whether tomorrow's prices will increase or decrease is more important than a ball park price estimate that is uncertain regarding trend.

The contribution of these particular results is that RMSE and MAPE alone are not sufficient to evaluate financial time series as low RMSE and MAPE values can be obtained by an ANN price prediction model returning the current day's price rather than a prediction for tomorrow's price.

ANN classification model

- *Can an ANN predict Bitcoin daily price trends more accurately than guessing majority class?*

As a simple logistic regression model could not be found to predict Bitcoin trends move accurately than majority class over the test set (53.3% compared to majority class of 59.4%), a more complex ANN model was utilized. Kara, Boyacioglu and Baykan (2011) demonstrated 75% accuracy predicting daily rise fall trends of the Istanbul stock exchange with 10 technical indicator features, 10 years of historic data and an ANN model. Qiu and Song (2016) could predict the Nikkei 225 index's daily rise fall trend with 81% accuracy using 10 technical indicator features and an ANN optimized by a genetic algorithm. Kamruzzaman and Sarker (2003) could predict rise fall trends of various FOREX currency pairs with an overall accuracy of approximately 80% using only moving average technical indicators and an ANN model.

It was not possible to build an ANN classification model that accurately predicted Bitcoin trends better than the majority class (59.4%). Generally speaking, the results of the ANN classification model (accuracy of 53.3%) are disappointing and identical to the results of the logistic regression model. The model's accuracy does not match that of related literature outlined above. This may be in part due to the nature of Bitcoin price and it's extreme volatility. It should be noted that other researchers used several years worth of data to build their models. This model was trained on only 931 days. The extreme rise and volatility of Bitcoin prices throughout 2017 is perhaps also a factor which makes prediction difficult as there is no past Bitcoin trend similar to 2017 from which the model can learn. Alternatively, this result can be seen to lend credence to the work of Pyo, Lee, Cha and Jang (2017) who also reported difficulty in obtaining accuracy scores similar to past work for daily trend predictions of financial time-series.

Bayesian approximation classification model

- *When next day Bitcoin price trend predictions are ordered in terms of the network's confidence in each prediction; does prediction accuracy increase with prediction confidence?*

The Bayesian approximation classification model scores similarly to the ANN classification model over the entire test set, i.e poorly. However, when predictions are ordered with regards to the model's confidence of predictions (i.e sorted via standard deviation), the 50 predictions with highest confidence (lowest uncertainty) are more accurate than majority class (62.0% versus 59.4%). If this model was to be used on a daily basis and a prediction is given with a standard deviation of <= 0.0358 (i.e a standard deviation in the lowest 50 of the predictions) then this model will predict the direction more accurately than simply guessing majority class of the whole test set. The ten predictions that the model tells us it is most certain about are 90% accurate.

The results of the ANN and Bayesian approximation network over the entire test set are not particularly inspiring and give the impression that such deep learning approaches, learning from economic and public awareness features, are almost useless in terms of trend prediction. However, by being able to order the Bayesian approximation predictions in terms of the model's confidence and seeing a decrease in prediction accuracy as prediction confidence decreases demonstrates that the model is indeed learning something from the data. These results are very interesting; there is some sort of signal in the data which the Bayesian approximation model architecture can pick up on (on certain days) and be able to confidently predict the next day's price trend.

These results contribute to the idea that even for poorly performing classifiers, ordering financial time series predictions in terms of prediction confidence can be useful. From a financial manager's or investment banker's perspective if an accurate prediction classifier cannot be built, why not still build a Bayesian approximation model and assess predictions based on their confidence? The Bayesian approximation model in this investigation performs poorly over the 165 day test set. However, predictions with a high confidence are demonstrated to be more accurate than less confident predictions.

Therefore, although not every prediction is useful, predictions with high confidence can be used to make informed investments. Such an investment strategy could be to only make investments when predictions are returned with a standard deviation value below a certain threshold. This is a novel contribution from this thesis as such an approach is not covered in the presented literature.

It should however be noted that although an increasing trend in prediction accuracy is found when results are ordered in terms of increasing confidence (Figure 7); ultimately, only the 10 most confident predictions would probably be of interest from a financial managers perspective. If the accuracy of the tenth to twentieth predictions is looked at, only 3 out of the 10 predictions are correct. Therefore, although the predictions of the Bayesian approximation network can be ordered in terms of increasing accuracy with increasing confidence, the network's confidence and accuracy is only strongly correlated for the 10 or so most confident predictions. In spite of this, the results contribute to the understanding that Bayesian approximation networks have significant advantages over ANNs. The output of the Bayesian approximation network is not a single point estimate but rather a prior probability from which confidence information can be inferred. Furthermore, the results highlight the benefits of using Dropout as a Bayesian approximation. With only an additional line of code it was possible to change an ANN into an approximation of a Bayesian neural network. The ANN was returning useless results yet from the Bayesian approximation network it was quickly possible to glean potentially financial valuable insight from the results once they were ordered in terms of model confidence.

Feature Importance

- *Which group of input features is more important to Bitcoin trend prediction with the Bayesian approximation model; economic or public awareness?*

A feature ablation approach on a Bayesian approximation network was used to evaluate the importance of each group of features to the trend prediction task. The usefulness of using economic features for financial time series prediction has been established (Kamruzzaman & Sarker, 2003; Kara, Boyacioglu and Baykan, 2011; Kimoto et al., 1990; Patel et al., 2015; Qiu & Song, 2016; Senol & Ozturan, 2008; Shah & Zhang, 2014). The background section also establishes a strong theoretical background regarding the link between financial time series trends and public awareness or news features (Bollen, Mao and Zheng, 2011; Borges, Goldstein, Ortmann and Gigerenzer, 1999; Chan, 2003; Fama, Fisher, Jensen and Roll, 1969; Garcia et al., 2014; Guan-Ru Wu, Chieh-Tse Hou and Lin, 2018; Kim et al., 2016) and the link between Bitcoin and public awareness (Kristoufek, 2013, 2015).

The results found that a Bayesian approximation network trained only on economic features was the most accurate (57.0%). The same model trained on only public awareness features was next most accurate (55.8%). Lastly, the model with the full combination of input features was least accurate (52.7%). The model with both the economic and public awareness features was expected to perform the best, as intuitively this model should include the most information for performing the classification task. It however performs the worst. The Bayesian approximation network includes only a single hidden layer of 28 neurons plus 20% Dropout, thus over-fitting is not considered to be a major issue. More likely, the issue with this model is that the model is too wide; perhaps using 12 features to try and pick out an already weak signal causes confusion, which affects accuracy.

To better understand the results, the predictions were ordered in terms of model confidence. The 10, 20 and 50 predictions with the highest confidence were then evaluated in terms of their accuracy and investment performance. The model with all input features scored 90%, 65% and 62% respectively for the 10, 20 and 50 predictions with highest confidence. The model with only economic features scored 70%, 55% and 56%. The model with only public awareness features scored 80%, 75% and 66% respectively. The model trained on public awareness features only seems to be the most sure about the classification task it is performing. Despite being the second best for general accuracy, when ordered in terms of confidence, prediction accuracy decreases the most smoothly of all the models (Figure 9). Even after 110 predictions, prediction accuracy is at 60.9%. The model with only economic features drops below 60% accuracy after only 20 predictions, the model with all features, after 60 predictions. The model with public awareness features appears to be the most sure about what it is trying to do.

Interestingly, the model with public awareness features does make price fall predictions but when ordered by confidence, none of these predictions fall within the top 110 most confident predictions. The other models both include numerous price fall predictions in their top 50 most confident predictions. However, the accuracy of these price fall predictions (in the top 50) are 29% and 25% for the model with all features and only economic features respectively (these results are gleaned from the confusion matrices presented in the appendix section). In summary, it appears that the model trained only on public awareness features acknowledges that it's price fall predictions are inaccurate and therefore assigns these predictions a low confidence.

Comparing Figures 7, 8 and 9 (plots of prediction accuracy for predictions ordered by model confidence for the models with all features, only economic features and only public awareness features respectively) demonstrates that predictions for the model with only public awareness features can best be ordered in terms of model confidence. Instead of only sorting out the 10 or so predictions with high accuracy, the prediction accuracy of the Bayesian approximation model trained only on public awareness data decreases relatively smoothly with prediction confidence. The sudden dip in accuracy seen for the models trained on all features and only economic features after 10 or so predictions (Figure 7 and Figure 8) is not present in Figure 9. It can therefore be said that prediction accuracy and model confidence is most strongly correlated for the model trained only on public awareness data. The correlation appears weakest for the model trained only on economic data (Figure 8).

The results of the investment returns for the 10, 20 and 50 most confident predictions of the three models demonstrate again that the model with public awareness features only seems to be better at attributing confidence to correct predictions than the other models. The profit for this model for the 50 most confident predictions was \$964, compared to \$752 for the economic features only model and \$571 for the model with all features. A completely naive model investing by assuming price rises for the first 50 days of the test set scored \$598. The average profit of the random investment strategy (assuming always price rise over the test set) is \$445. This demonstrates that all three models with the various features provide meaningful predictions (when the 50 most confident predictions are taken) that outperform a random majority class guessing investment strategy. However, the model with public awareness features only is the most profitable with \$964 for 50 investment days.

Both economic and public awareness features have shown to be useful when performing the Bitcoin trend prediction task. Interestingly, the investment results of using the feature groups in isolation were better than using the features together in the same model. Ultimately, stating that one group of features is more important that the other is not entirely possible based on the results of this investigation. The Bayesian approximation model scored better over the entire test set with economic features only, compared to public awareness features. Yet the model trained only with public awareness features was found to be more sure of the classification task it was trying to carry out, and when ordered by confidence, the accuracy was found to decrease the most smoothly with confidence. Table 3 listed the linear correlation coefficients between Bitcoin price and each of the model features, there was a very strong linear correlation between two of the three public awareness features and next day price. Presumably, it is this correlation that is aiding the model, trained only on public awareness data, to make profitable predictions that can be ordered reasonably well with regards to confidence and accuracy.

The most interesting and exciting result from this part of the investigation is that the most profitable investment strategy is found to be that using the prediction model trained only public awareness features (so long as prediction confidence is employed in any investment strategy). This is a relevant contribution as not only does it support literature findings such as Kristoufek (2013, 2015) describing a strong correlation between Bitcoin price and public awareness, but the results show that Bitcoin price movements can, to a certain extent, be predicted by public awareness.

Conclusion

Using RMSE and MAPE in isolation, as in the work of Jang and Lee (2018) and Ticknor (2013), has been demonstrated to not be rigorous enough evaluation criteria for financial prediction models attempting to predict next day prices. Such models should be able to also demonstrate a reasonable trend accuracy and ensure that predictions are not simply a rehash of daily price information. Literature that presents results only with RMSE and MAPE should be looked at critically.

Despite using an approach informed by compelling financial time series prediction literature, this thesis was unable to create an ANN that could outperform majority class Bitcoin next day price trend prediction. Bitcoin price is very volatile and is affected by numerous and wide ranging factors. The prediction challenge is further hampered by the lack of training data; the price of Bitcoin increased 14 fold during 2017 and thus pre 2017 training data fails to capture much of the recently experienced volatility and price rise.

Dropout as a Bayesian approximation is a useful approach that, in this thesis, has been demonstrated to be able to successfully attribute more confidence to more accurate predictions. This is most pronounced in the Bayesian approximation model trained only on public awareness features (Figure 9). This is an exciting result that can provide genuinely useful insight with regards to Bitcoin prediction, albeit not for every day. Furthermore, this successful ordering of predictions based on model confidence is a novel contribution to Bitcoin trend prediction.

Both economic and public awareness features were found to be important in the Bitcoin price trend prediction challenge. It was not possible to categorically conclude which of the feature groups was most useful to the classification task as the economic features proved more accurate over the entire test set, yet the public awareness features enabled the Bayesian approximation model to be more sure about it's predictions. The most profitable model was found to be one based only on public awareness features. This result further contributes to the large body of literature detailing the important relationship between news and public awareness features and financial indexes. Finding that, to an extent, Bitcoin trends can be predicted using only public awareness features was the most surprising, and interesting contribution of this investigation. Furthermore, demonstrating that a profitable investment strategy can be achieved using only public awareness measures is another interesting finding from this report.

Limitations of this work include that despite demonstrating a relationship between accuracy and model confidence, this relationship was not as pronounced as hoped. For the Bayesian approximation models trained on all the features and only economic features, the confidence information essentially only allowed 5 - 10 predictions with high accuracy to be isolated. It would be interesting to look at this relationship for a model with a better generalization performance over an entire test set, to see if it is more significant. Furthermore, the Bayesian approximation approach should be compared to other ordering approaches, such as ordering ANN predictions via the values of the sigmoid outer layer, to better understand it's usefulness.

The 50 most confident predictions of all three Bayesian approximation models (trained on either all features, only economic or only public awareness features) were found to be more profitable than a majority class investment strategy that picked 50 days at random from the test set. However, this approach is not a practical real world application of an investment strategy, as for financial time-series, only a single prediction (i.e tomorrow's) would be available for a trader. Therefore, the investment strategy present would need to be cross validated on new test sets to better understand it's robustness. A real world application of such a strategy would require a standard deviation threshold to be established. If a prediction for tomorrow's trend is made below such a threshold, an investment could be made.

References

Adkisson, J. (2018, February). *Why bitcoin is so volatile.* Retrieved from https://www.forbes.com/sites/jayadkisson/2018/02/09/why-bitcoin-is-so-volatile/#6c887bb539fb (Last accessed 16 May 2018)

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of computational science*, *2*(1), 1–8.

Borges, B., Goldstein, D., Ortmann, A., & Gigerenzer, G. (1999). Can ignorance beat the stock market. In S. Stick (Ed.), *Simple heuristics that make us smart* (p. 59 - 72). Oxford University Press.

Chan, W. (2003). Stock price reaction to news and no-news: drift and reversal after headlines. *Journal of Financial Economics*, *70*(2), 223–260.

Chaparro, F. (2017, December). *Bitcoin went bonkers in 2017 — here's what happened as the cryptocurrency surged more than 1000%.* Retrieved from https://www.businessinsider.de/bitcoin-price-in-2017-review-2017-12?r=US&IR=T (Last accessed 25 June 2018)

Fama, E., Fisher, L., Jensen, M., & Roll, R. (1969). The adjustment of stock prices to new information. *International economic review*, *10*(1), 1–21.

Gal, Y. (2015, July). *[blog]what my deep model doesn't know...* Retrieved from http://mlg.eng.cam.ac.uk/yarin/blog_3d801aa532c1ce.html (Last accessed 17 May 2018)

Gal, Y., & Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning* (pp. 1050–1059).

Garcia, D., Tessone, C., Mavrodiev, P., & Perony, N. (2014). The digital traces of bubbles: feedback cycles between socio-economic signals in the bitcoin economy. *Journal of the Royal Society Interface*, *11*(99), 20140623.

Gdelt. (2015a, February). *Gdelt 2.0: Our global world in realtime.* Retrieved from https://blog.gdeltproject.org/gdelt-2-0-our-global-world-in-realtime/ (Last accessed 11 May 2018)

Gdelt. (2015b, February). *The gdelt event database data format codebook v2.0.* Retrieved from http://data.gdeltproject.org/documentation/GDELT-Event_Codebook-V2.0.pdf (Last accessed 11 May 2018)

Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural computation*, *4*(1), 1–58.

Goodfellow, I., Yoshua, B., & Courville, A. (2016). *Deep learning.* MIT Press. (http://www.deeplearningbook.org)

Guan-Ru-Wu, G., Chieh-Tse-Hou, T., & Lin, J. (2018). Can economic news predict taiwan stock market returns? *Asia Pacific Management Review.*

Hagan, S. (2018, February). *Cyptocurrencies are like ponzi schemes, world bank chief says.* Retrieved from https://www.bloomberg.com/news/articles/2018-02-07/cryptocurrencies-are-like-ponzi-schemes-world-bank-chief-says (Last accessed 22 March 2018)

Heinz, S. (2017, November). *A simple deep learning model for stock price prediction using tensorflow [blog post].* Retrieved from https://medium.com/mlreview/a-simple-deep-learning-model-for-stock-price-prediction-using-tensorflow-30505541d877 (Last accessed 23 May 2018)

Jang, H., & Lee, J. (2018). An empirical study on modeling and prediction of bitcoin prices with bayesian neural networks based on blockchain information. *IEEE Access*, *6*, 5427–5437.

Johansson, E. (2014, December). *Creating daily search volume data from weekly and daily data [blog post].* Retrieved from http://erikjohansson.blogspot.de/2014/12/creating-daily-search-volume-data-from.html (Last accessed 14 April 2018)

Kamruzzaman, J., & Sarker, R. (2003). Forecasting of currency exchange rates using ann: A case study. In *Neural networks and signal processing, 2003. proceedings of the 2003 international conference on* (Vol. 1, pp. 793–797).

Kara, Y., Boyacioglu, M., & Baykan, O. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the istanbul stock exchange. *Expert systems with Applications*, *38*(5), 5311–5319.

Kim, K. (2003). Financial time series forecasting using support vector machines. *Neurocomputing*, *55*(1-2), 307–319.

Kim, Y., Kim, J., Kim, W., Im, J., Kim, T., Kang, S., & Kim, C. (2016). Predicting fluctuations in cryptocurrency transactions based on user comments and replies. *PloS one*, *11*(8), e0161197.

Kimoto, T., Asakawa, K., Morio, Y., & Takeoka, M. (1990). Stock market prediction system with modular neural networks. In *Neural networks, 1990., 1990 ijcnn international joint conference on* (pp. 1–6).

Kristoufek, L. (2013). Bitcoin meets google trends and wikipedia: Quantifying the relationship between phenomena of the internet era. *Scientific Reports*, *3(1).* Retrieved from `doi:10.1038/srep03415`

Kristoufek, L. (2015). What are the main drivers of the bitcoin price? evidence from wavelet coherence analysis. *Plos One*, *3(1).* Retrieved from `doi:10.1371/journal.pone.0123923`

Mamoria, M. (2018, January). *Cryptocurrency doesn't have the same price between countries – here's why [blog post].* Retrieved from `https://hackernoon.com/cryptocurrency-doesnt-have-the-same-price-between-countries-here-s-why-2c0b44f57a14` (Last accessed 21 April 2018)

Martin, W. (2018, January). *The global cryptocurrency market hit a new record high above $700 billion.* Retrieved from `https://www.businessinsider.nl/bitcoin-price-global-cryptocurrency-market-capitalisation-january-3-2018-1/?international=true&r=UK` (Last accessed 21 March 2018)

Murphy, J. (1999). *Technical analysis of the financial markets: A comprehensive guide to trading methods and applications.* Penguin.

Nakamoto, S. (2008). *Bitcoin: A peer-to-peer electronic cash system.* Retrieved from https://bitcoin.org/bitcoin.pdf

Ni, H., & Yin, H. (2009). Exchange rate prediction using hybrid neural networks and trading indicators. *Neurocomputing*, *72*(13-15), 2815–2823.

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock market index using fusion of machine learning techniques. *Expert Systems with Applications*, *42*(4), 2162–2172.

Polasik, M., Piotrowska, A., Wisniewski, T., Kotkowski, R., & Lightfoot, G. (2015). Price fluctuations and the use of bitcoin: An empirical inquiry. *International Journal of Electronic Commerce*, *20*(1), 9–49.

Pyo, S., Lee, J., Cha, M., & Jang, H. (2017). Predictability of machine learning techniques to forecast the trends of market index prices: Hypothesis testing for the korean stock markets. *PloS one*, *12*(11), e0188107.

Qiu, M., & Song, Y. (2016). Predicting the direction of stock market index movement using an optimized artificial neural network model. *PloS one*, *11*(5), e0155133.

Senol, D., & Ozturan, M. (2008). Stock price direction prediction using artificial neural network approach: The case of turkey. *Journal of artificial Intelligence*, *1*(2), 70–77.

Shah, D., & Zhang, K. (2014). Bayesian regression and bitcoin. In *Communication, control, and computing (allerton), 2014 52nd annual allerton conference on* (pp. 409–414).

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929–1958.

Ticknor, J. (2013). A bayesian regularized artificial neural network for stock market forecasting. *Expert Systems with Applications*, *40*(14), 5501–5506.

Appendix

Throughout this project Python 3 was used including the numpy, sklearn, pandas and keras packages. In order to achieve consistent results it should be noted that the same random seeds for numpy and tensorflow were used throughout this investigation.

Data notes:

The various datasets were loaded into a Python programming environment as Panda DataFrames, each with a Date column serving as both Primary and Foreign Key. In all cases, the Date column was then standardized to a Pandas DataFrame format.

Note that on 28 February 2017 and 14 May 2016 there was no reported difference in Bitcoin's closing price compared to the previous day's. The data for these two days were removed from the model. As having only two examples would not be enough to train a third classification (ie. no movement).

The Bitcoin economic data was fed into the functions written to produce Pandas series for each of the technical indicators. All the technical indicator series and the Panda dataframes for public awareness features were then concatenated together using the 'Date' foreign key, creating a unified DataFrame with all the public awareness and economic data. The datasets were joined using an outer join, so that any missing values would be present in the final DataFrame.

Technical indicators such as Relative Strength Index (RSI) require price information for the previous 13 days before an RSI value can be calculated for the fourteenth day. Therefore, the first 14 days of the dataset were removed from the model after calculating all technical indicator features (as no RSI value was available for them). The technical indicators were programmed as python functions and can be found at: **https://github.com/mpphughes/bitcoinprediction**

All reported results are the average result of 30 runs of the model. Specifically, the models are run 30 times and the mean value for each prediction is calculated. From this array of mean predictions an accuracy and/ or RMSE value is reported.

Confusion matrices

Table 9

ANN Bitcoin daily rise fall trend classification confusion matrix

	Predicted Rise	Predicted Fall
Actual Rise	63	35
Actual Fall	42	25

Table 10

Bayesian approximation network Bitcoin daily rise fall trend classification confusion matrix (all features)

	Predicted Rise	Predicted Fall
Actual Rise	63	43
Actual Fall	43	24

Table 11

50 most confident predictions of Bayesian approximation network confusion matrix (all features)

	Predicted Rise	Predicted Fall
Actual Rise	25	4
Actual Fall	15	6

Table 12

Bayesian approximation network classification model with economic features only confusion matrix

	Predicted Rise	Predicted Fall
Actual Rise	66	32
Actual Fall	39	28

Table 13

Bayesian approximation network classification with public awareness features only confusion matrix

	Predicted Rise	Predicted Fall
Actual Rise	80	18
Actual Fall	55	12

Table 14

50 most confident predictions of Bayesian approximation model with economic features only confusion matrix

	Predicted Rise	Predicted Fall
Actual Rise	23	7
Actual Fall	15	5

Table 15

110 most confident predictions of Bayesian approximation model with public awareness features only confusion matrix

	Predicted Rise	Predicted Fall
Actual Rise	67	0
Actual Fall	43	0

Table 16

Bayesian approximation network prediction accuracy for predictions with lowest standard deviation

# of predictions	prediction std	Accuracy
Lowest 5	<= 0.0283	100%
Lowest 10	<= 0.0300	90.0%
Lowest 20	<= 0.0321	65.0%
Lowest 30	<= 0.0340	63.3%
Lowest 40	<= 0.0340	62.5%
Lowest 50	<= 0.0358	62.0%
Lowest 60	<= 0.0395	55.0%

Table 17

Bayesian approximation network classification with Economic features only for predictions with lowest standard deviation

# of predictions	prediction std	Accuracy
Lowest 5	<= 0.0217	80%
Lowest 10	<= 0.0230	70%
Lowest 20	<= 0.0264	55%
Lowest 30	<= 0.0276	56.7%
Lowest 40	<= 0.0289	57.5%
Lowest 50	<= 0.0299	56.0%
Lowest 60	<= 0.0313	58.3%

Table 18

Bayesian approximation network classification with Public Awareness features only for predictions with lowest standard deviation

# of predictions	prediction std	Accuracy
Lowest 5	<= 0.0082	80%
Lowest 10	<= 0.0084	80%
Lowest 20	<= 0.0093	75%
Lowest 30	<= 0.0099	66.7%
Lowest 40	<= 0.0103	67.5%
Lowest 50	<= 0.0111	66.0%
Lowest 60	<= 0.0112	65.0%
Lowest 110	<= 0.0180	60.9%
Lowest 120	<= 0.0187	58.3%