Cryptocurrency Price Predictor

Benjamin Carpenter, Shuning Jin, Jacob Pauly, Tristan Larsin
With help from Josh Zhao

February 23, 2018

1 Objectives

Our goal is to leverage machine learning to predict future prices in the cryptocurrency market. If we are successful it would empower businesses and individuals to make better financial investments. It would also assist in understanding the probabilities of risk and reward for specific investments.

2 Dataset

Our dataset is comprised of two separate pieces. The first is the historical price of Bitcoin from January 2012 to January 2018 [1]. The second is a set of 1200 top performing cryptocurrencies (sans Bitcoin) [2]. The dataset includes daily records of each cryptocurrency during a time span of 1384 days from September 11, 2013 to June 26, 2017, with total 567,769 observations. On average, each cryptocurrency accounts for 470 records. It is characterized by 8 attributes: timestamp, opening price, daily high, daily low, closing cost, daily volume, weighted price and symbol.

3 Algorithms

Time series forecasting is an important field in statistics. Classical models include ARIMA, GARCH, etc. In the past decade, machine learning has gained popularity in TSA studies, and especially Support Vector Machine (SVM) is widely applied. Currently, deep learning models stand out with their aptitude for sequential data, such as Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM).

Autoregressive Integrated Moving Average Model (ARIMA)

ARIMA model is a generalization of ARMA model, with the form:

$$\phi(B)(1-B)^d X_t = \delta + \theta(B)w_t \qquad for\ t = 1, 2, \ldots$$

where $\delta = \mu(1 - \sum_{i=1}^{p} \phi_i)$, w_t is white noise with independent and identical distributions, ϕ and θ are vectors of unknown fixed regression coefficients, and B is back-shifting operator defined as $B^d x_t = x_{t-d}$.

(i) Model Specification

The ARIMA(p,d,q) model consists of three parts:

Autoregression of order p: AR assumes current value x_t is correlated with p past values, where p denotes the number of time lags. The autoregressive process models the trend of x_t.

Moving average of order q: MA assumes current value x_t is correlated with q recent white noises. The moving average smoother is to remove white noise in data and helps discover underlying long-term trend.

Differencing of order d: time series are usually modeled by a stationary component and a nonstationary component. Performing differencing is to remove trend and produce stationarity. First difference is defined as $\nabla x_t = x_t - x_{t-1}$.

(ii) Model Fitting

Given p,d,q order for ARIMA model, we shall firstly do d-order differencing and reduce it to ARMA(p,q). Secondly, we focus on estimating parameters ϕ and θ in ARMA model. There are two typical techniques for such estimation: Yule-Walker estimation and maximum likehood estimation.

Generalized Additive Model (GAM)

The additive model is a nonparametric regression method. It has the form

$$Y = \alpha + \sum_{j=1}^{p} f_j(X_j) + \epsilon$$

Where Y is the outcome, X_j terms are predictors; f_j terms are unspecified smooth (nonparametric) functions, α is the constant intercept and ϵ is the error term. The goal is to estimate the unknown smooth functions fj's. Fitting AM can be done by backfitting algorithm with an iterative procedure.

Support Vector Machines (SVM)

SVM constructs a hyperplane or set of hyperplanes in a high-dimensional space, which can be used for classification, regression. It allows flexible mapping of high dimensional features to capture non-linear relationships with regularization to avoid over-fitting.

Justification

For the algorithm, we decide to employ ARIMA model in lieu of GAM. Since GAM is used less frequently in time series analysis, we do not find sufficient resources to help us fulfill the implementation. On the contrast, ARIMA is a classical and well-studied statistical model in this domain. Its statistical principles and algorithms are introduced in details by textbooks. We mainly consult the book Time Series Analysis and Its Applications [3]. Considering our deficient knowledge in time series analysis, we perceive it more reasonable and educational to investigate a simpler model like ARIMA first. Later on, we can further explore other more complex models like GARCH or GAM.

Evaluation Metrics

(i) Goodness of Fit

Akaike's Information Criterion (AIC)

$$AIC = log\hat{\sigma}_k^2 + \frac{n+2k}{n}$$

AIC, Bias Corrected (AICc)

$$AICc = log\hat{\sigma}_k^2 + \frac{n+k}{n-k-2}$$

Bayesian Information Criterion (BIC)

$$BIC = log\hat{\sigma}_k^2 + \frac{klogn}{n}$$

Mean Square Error (MSE)

$$MSE = \frac{SSE(k)}{n-k}$$

where $\hat{\sigma}_k^2$ is the maximum likelihood estimator of variance, SSE is residual sum of squares, k is the number of parameters, and n is the number of observations.

(ii) Prediction Accuracy

Mean Absolute Percentage Error (MAPE)

$$M = \frac{100}{n}\sum_{t=1}^{n}|\frac{A_t - F_t}{A_t}|$$

where A_t is the actual value and F_t is the forecast value.

4 Related Works

The work proposed by Zhang et al. [4] is a hybrid methodology to combine the linear ARIMA and nonlinear ANN models for time series forecasting to improve performance. The empirical results with three real data sets suggest that the hybrid model outperforms its component model, and Canadian lynx data shows 19% decrease in MSE and 8% decrease in MAD.

In a 2014 white paper [5], a team from the University of Manitoba uses structural support vector machines (SSVMs) in an attempt to predict future stock market prices. SSVMs are a specialized form of supervised learning algorithm that can perform linear and non-linear regression to predict similar outcomes from past data [6]. In their results, they found the algorithm was performing with 78% accuracy.

In 2017, Facebook developed Prophet [7], which is a powerful tool for time series forecasting available in Python and R. Prophet is aimed at efficiently producing high quality forecasts with large number of time series data points. They used the generalized additive model (GAM), which is based on time series decomposition. Three main components are used: trend, seasonality, and holiday. GAM has greate flexibility when used non-parametrically.

Autoregressive Integrated Moving Average (ARIMA) was used by Vinay et al. [8] to predict road-traffic volume. ARIMA is used for short-term and long-term historical data and has statistical superiority. There happens to be several variations of ARIMA that are best suited for given problems. In this paper, This research tested all of the variations to compare the different accuracies of his road-traffic volume predictions. Some variations have more overhead than others, but lack accuracy. Based purely on accuracy ARIMA-GARCH outperformed all other variations of ARIMA.

In a 2016 paper [9], a team used a support vector machine (SVM) and kernel functions to forecast the stock market. They separated the stock market changes into two classifications, rising and declining. When rising, the value at $x-1$ is less than that at x. When declining, the value at $x+1$ is less than that at x. The types of SVM kernels used were linear, polynomial,

and radial basis. The most accurate SVM kernel performed with an 88.34% accuracy.

Additional methods used by David Sheehan [10] including the long short term memory (LSTM) model, which is a type of deep learning that can predict cryptocurrency prices. LSTM models are well suited for time series data because loops in neural nets allow for persistent data. In essence, they can remember how time series data has behaved in the past. Sheehan decided to use the same Bitcoin dataset that we have chosen. This dataset includes Bitcoin's details at a one-minute interval from January 2012 through January 2018. This provided him with six years of data. Testing his algorithm from July 1st, 2017 through November 1st, 2017 he received a mean absolute error (MAE) of 0.0392.

In our project, we will apply both SVM and GAM for cryptocurrency price prediction. While SVM has been widely applied for stock price forecasting, GAM is much less surveyed. Thus, we are interested in further investigating GAM and comparing their performance in time series forecasting.

5 Time Table

Date	Action	Notes	Deliverables	Status
2.1	Written Project Proposal	Write out the description of our project, algorithms and previous works.	Project Proposal	Completed
2.6	Preprocessing	Deal with missing values (depends on models), data transformation.	Prepared dataset	Completed
2.12	Structure project & start development	Layout the structure of our project (file wise). Delegate roles for each section of our project. Start working on respective sections. Preliminary data analysis.	File structure and data set uploaded to Git repository.	Completed
2.25	Implement ARIMA algorithm	Testable algorithm.	Testable project	Partially done
3.10	Implement SVM algorithm	Both algorithms are now implemented and ready to be tested	2 testable algorithms	Not started
3.15	Devise Tests	Create test parameters to check accuracy of our algorithms	Runnable tests	Not started
3.22	Test different parameters	Change of the parameters we are using to train the algorithm to test if we achieve a higher accuracy.	Comparison of new parameters to previous	Not started
3.29	Choose best parameters	Choose the best performing parameters.	Best performing parameters	Not started
4.5	Run algorithms on several cryptocurrencies	Run algorithms on several cryptocurrencies to access final accuracies.	Accuracy of algorithms	Not started
4.24	Complete project	Submit final report	Final algorithms and accuracy	Not started

Justification

A previous version of this proposal had an unreasonable completion date for the time series algorithms, including finishing initial implementation before they had been introduced in the course.

Progress

(i) Preliminary Analysis

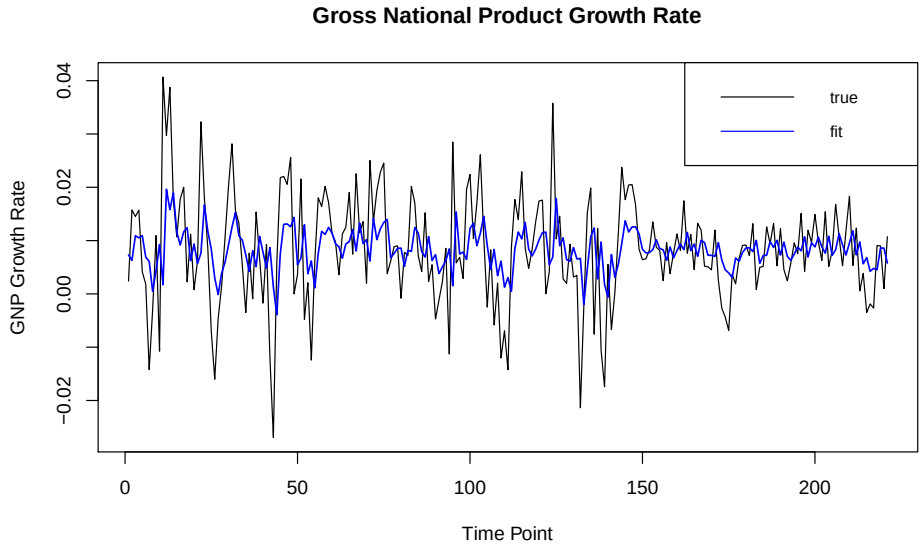
We did a rough statistical analysis of our dataset for model checking. First, we plotted the closing price versus time points to learn the trend, and we decided to truncate the early points as they exhibit an apparent departure from the recent trend. Additionally, we did the first differencing to eliminate the linear trend and check stationarity.

(ii) ARIMA Implementation

We have been devoted to implementing ARIMA model. So far, we have implemented Autoregressive Model in R to estimate parameters, and developed evaluation metrics for goodness of fit. We are still proceeding Moving Average part. We plan to use numerical approximation via iterations to fit additional coefficients, and typically we are focusing on Newton–Raphson and Yule–Walker algorithms.

(iii) Test Result

We tested the model on the GNP dataset from 'astsa' package in R, which includes Quarterly U.S. Gross National Product from 1947(1) to 2002(3). With autoregressive order of 1, the fitted model gives performance: AIC -8.29, AICc -8.28, BIC -9.26, and MSE 9.11×10^{-5}.



References

[1] *https://www.kaggle.com/mczielinski/bitcoin-historical-data*

[2] *https://www.kaggle.com/akababa/cryptocurrencies*

[3] *Shumway RH, Stoffer DS. Time Series Analysis and Its Applications. New York: Springer. 2000.*

[4] *Zhang GP. Times series forecasting using a hybrid ARIMA and neural network model. Neurocomputing 50:159–75 2003.*

[5] *Carson Kai-Sang Leung, Richard Kyle MacKinnon, and Yang Wang. A machine learning approach for stock price prediction. In Proceedings of the 18th International Database Engineering & Applications Symposium (IDEAS '14), Ana Maria Almeida, Jorge Bernardino, and Elsa Ferreira Gomes (Eds.). ACM, New York, NY, USA, 274-277. 2014.*
DOI: https://doi.org/10.1145/2628194.2628211

[6] *Tsochantaridis, Ioannis, et al. Support Vector Machine Learning for Interdependent and Structured Output Spaces. Twenty-First International Conference on Machine Learning - ICML '04, 2004.*
DOI:10.1145/1015330.1015341.

[7] *Taylor SJ, Letham B. Forecasting at scale. PeerJ Preprints 5:e3190v2 2017.*
DOI: https://doi.org/10.7287/peerj.preprints.3190v2

[8] *Vinay B. Gavirangaswamy, Gagan Gupta, Ajay Gupta, and Rajeev Agrawal. Assessment of ARIMA-based prediction techniques for road-traffic volume. In Proceedings of the Fifth International Conference on Management of Emergent Digital EcoSystems (MEDES '13). ACM, New York, NY, USA, 246-251. 2013.*
DOI: http://dx.doi.org/10.1145/2536146.2536176

[9] *Ved Prakash Upadhyay, Subhash Panwar, Ramchander Merugu, and Ravindra Panchariya. Forecasting Stock Market Movements Using Various Kernel Functions in Support Vector Machine. In Proceedings of the International Conference on Advances in Information Communication Technology & Computing (AICTC '16), S. K. Bishnoi, Manoj Kuri, and Vishal Goar (Eds.). ACM, New York, NY, USA, Article 107 , 5 pages. 2016.*
DOI: https://doi.org/10.1145/2979779.2979886

[10] *David Sheehan. Predicting Cryptocurrency Prices With Deep Learning. (November 2017). 2017.*