Ethereum Price Forecasting with Machine Learning

An Application of Time Series Regression Models and Neural Networks

Prepared by: Brian McGuckin

Thinkful Data Science Program: Final Capstone

1. Introduction

A Brief Primer on Cryptocurrencies

- Cryptocurrencies are digital, decentralized mediums of exchange
  - Strong cryptography is used to secure and verify transactions
  - Decentralization is achieved through the use of a distributed ledger (Blockchain)
- Cryptocurrencies offer several advantages over the traditional banking system:
  - Fraud is reduced as cryptocurrencies cannot be counterfeited & transactions cannot be reversed
  - Eliminates the need for third party involvement, providing for:
    - Faster settlements
    - Lower transaction fees
    - Accessibility for unbanked populations
  - No exchange rates, duties, or intermediaries to deal with when dealing across borders
  - Privacy: personal information is not necessary to complete transactions
- However, greatest impact of these benefits is contingent on adoption

The Problem: Cryptocurrency Price Volatility

- Volatile markets make cryptocurrencies unreliable stores of value
  - Uncertainty stifles adoption
  - Investing in volatile markets can be risky but rewarding
- Predicting price movements aids speculation
  - Price stabilization promotes adoption and use
  - Potential profits increase investor participation, mitigating manipulation
  - Trading in cryptocurrency markets is big business – top banks are investing heavily in cryptocurrency and blockchain technologies
- This project focuses on predicting Ethereum (Ether, ETH) prices
  - Ethereum: a distributed computing platform and operating system characterized by smart contracts
  - Ether (ETH): cryptocurrency component of Ethereum, it’s blockchain is generated by Ethereum and can be used to compensate participant mining nodes for their computations
  - While Ether is technically the cryptocurrency for the Ethereum platform, the term Ethereum is commonly used to reference the platform and the cryptocurrency alike

Data Access & Cleaning

- Primary time series: Ethereum (ETH) daily close price
  - Daily frequency used to standardize the coin prices in the larger industry context
  - Cryptocurrency trading prices are commonly expressed in terms of OHLCV, despite the fact that trading is not limited to conventional exchanges and financial service providers
  - This allows for reliable comparison to other cryptocurrencies as well as more traditional financial instruments traded during standard hours
- Data access: web APIs were used to get current price information
  - Cryptocurrency data was sourced from Crypto Compare (cryptocompare.com)
  - Other exogenous economic data was sourced from the Economic Reserve Economic Database, commonly referred to as FRED (fred.stlouisfed.org)
- Cleaning/preprocessing:
  - Dates converted to datetime objects, then used as datetime index
  - Indices not traded/calculated for weekends/holidays: forward fill previous value until trade activity resumed
  - Depending on the date range being analyzed, some cryptocurrencies were missing data since they came to exist later on, and were simply dropped if this was the case

Experiment Overview

- First, models forecasted ETH price
  - Forecasting method: rolling window, one step ahead out of sample forecast
  - Models and Python implementations used:
    - Auto Regressive Integrated Moving Average (ARIMA): Statsmodels
    - Long Short-Term Memory Recurrent Neural Network (LSTM RNN): Tensorflow/Keras
  - Evaluation metric: RMSE (Root Mean Squared Error)
    - Absolute measure of error (for a given set of data)
    - Thus, various models can be compared to one another
- Then, models were provided with some exogenous variables to try and improve forecast error
  - Forecasts were done using best performing ARIMA and LSTM specifications
  - Attempt to further minimize RMSE over using just the time series to make forecasts
  - Additionally, this can provide some insight into exogenous drivers of ETH price

Forecasting Terminology

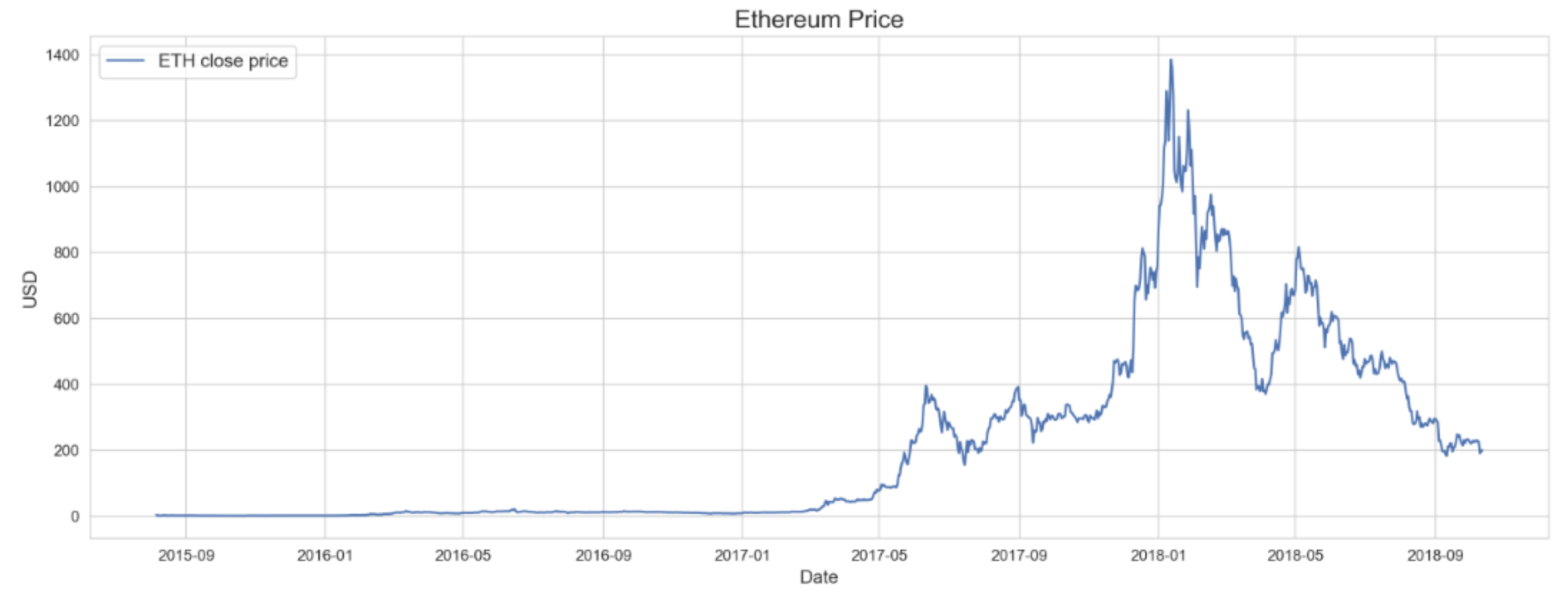
- In-sample: window-length subset of data used to train a model
- Out-of-sample: a model's forecast(s), evaluated against the observed target values
- Expanding (recursive) window: training data includes all previous values up to the next forecast
  - For a one step ahead forecast, window starts at beginning of series and expands by 1 observation per iteration
  - All available time series history is used to make forecasts
- Rolling window: training data is a fixed window length subset of data leading up to the next value to be forecasted
  - For each iteration of a one step ahead forecast, the window drops the oldest observation and includes the observed value for the forecast that was just made
  - Using only recent time series history allows outlier values to fade & can provide a more accurate picture of the target variable's current behavior

Models

- ARIMA (Auto Regressive Integrated Moving Average)
  - Extension of OLS linear regression
  - Denoted ARIMA(*p*, *d*, *q*):
    - *p* = order of autoregressive component (AR)
    - *d* = degree of differencing (I)
    - *q* = order of moving average component (MA)
- LSTM RNN (Long Short-Term Memory Recurrent Neural Network)
  - LSTMs were developed as a solution to the vanishing gradient problem
    - Due to how NN weights are updated by gradient based learning methods, gradients can effectively ‘vanish’ resulting in a NN unable to update its weights (and therefore unable to train itself)
  - LSTM unit: composed of a cell, an input gate, forget gate, and output gate
    - Input gate: controls the extent to which a new value flows into the cell
    - Forget gate: controls the extent to which a value remains in the cell
    - Output gate: controls the extent to which the cell’s value is used to compute the output
    - There are connections (some recurrent) into and out of the LSTM gates
    - Gate behavior is determined by connection weights, which are learned/updated during training

2. Exploratory Data Analysis

Ethereum Time Series

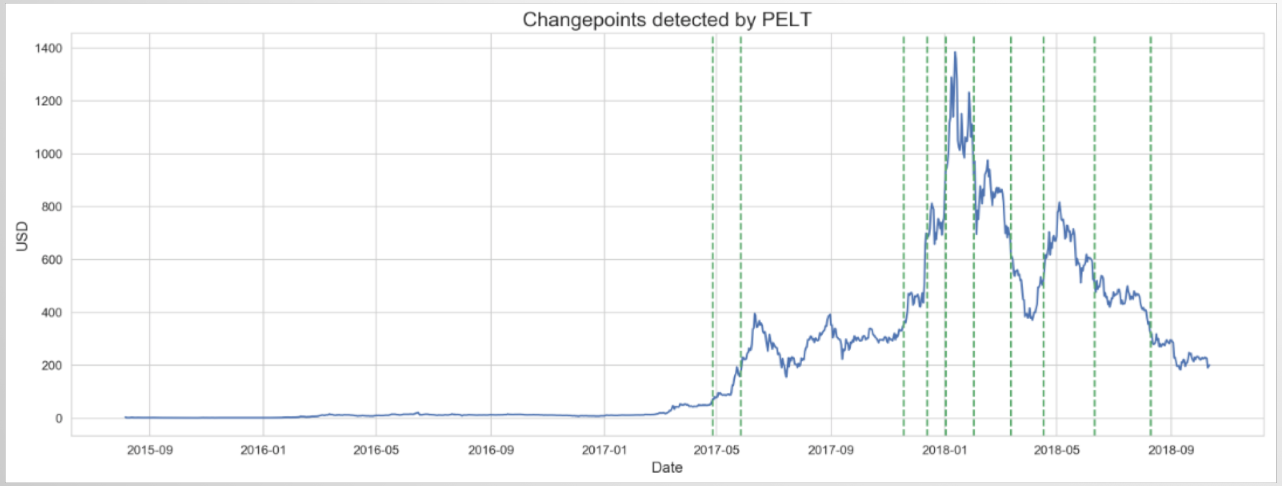


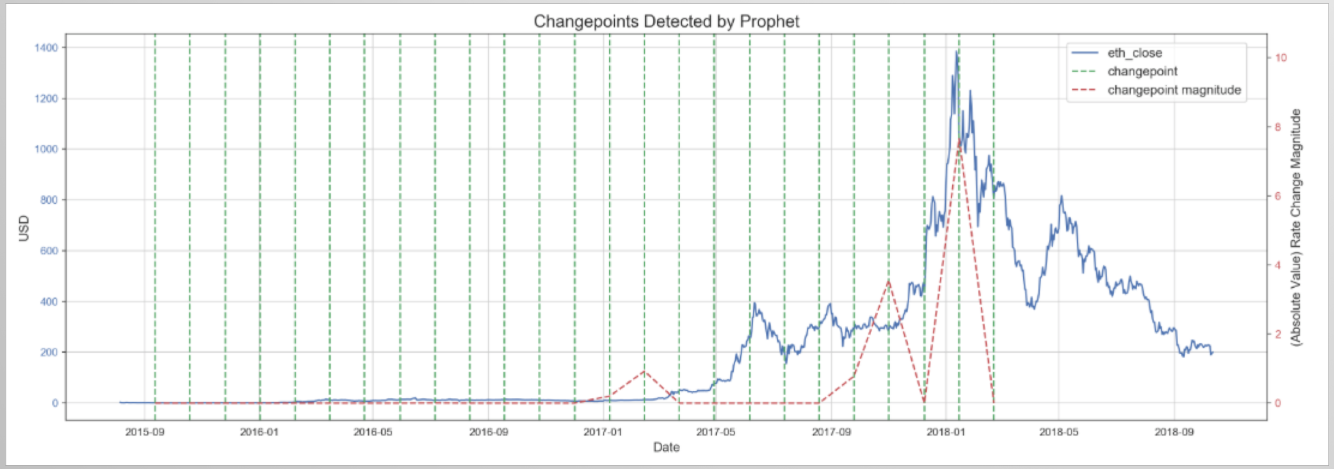

```
Augmented Dickey-Fuller Test Results:
adf: -1.828426625686884
pval: 0.3664427588368545
critical vals {'1%': -3.4360991979671205, '5%': -2.864078635828568, '10%': -2.568121635118498}
```

- Interpreting ETH price visualization and ADF results:
  - Series is not stationary; displays trend
    - Additionally, multiple structural breaks are present
    - Structural break (or regime change)
  - No seasonal component

- Stationary process: process for which a shift in time does not cause a change in distribution
- ADF (Augmented Dickey-Fuller) Test:
  - Null hypothesis test for the presence of a unit root in a time series
  - p-value: not significant, cannot reject the null hypothesis that a unit root is present
- Structural Break (or regime change):
  - An unexpected shift in a time series
  - Structural instability can lead to large forecasting errors & overall unreliable models

Structural Breaks: Changepoint Detection Algorithms

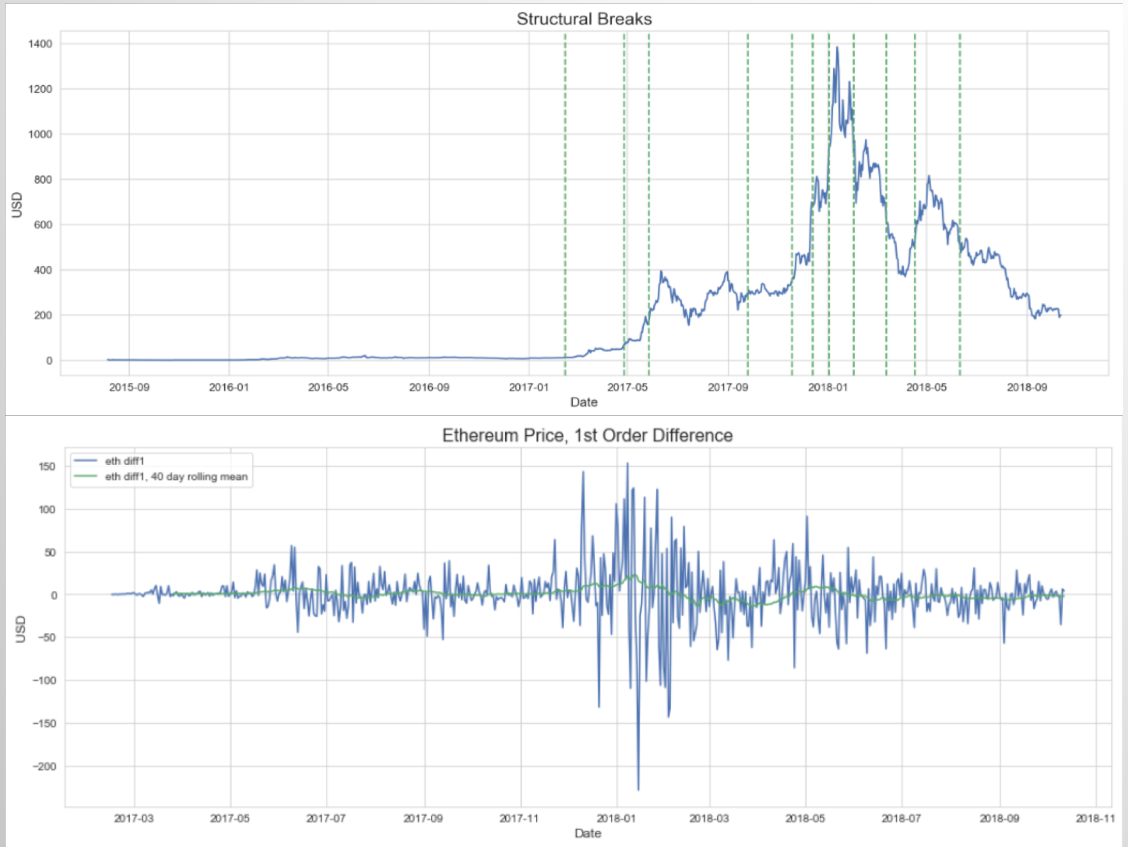




- PELT: Pruned Exact Linear Time
  - Arrives at changepoints by splitting series into segments and minimizing cost in relation to the series taken as a whole
  - Python implementation by the Ruptures package

- Prophet: time series forecasting by Facebook
  - Contains built in changepoint detection
  - Specifies potential changepoints and then chooses which to use by applying a sparse prior on the magnitudes of rate changes

Structural Breaks, Window Size, & Stationarity

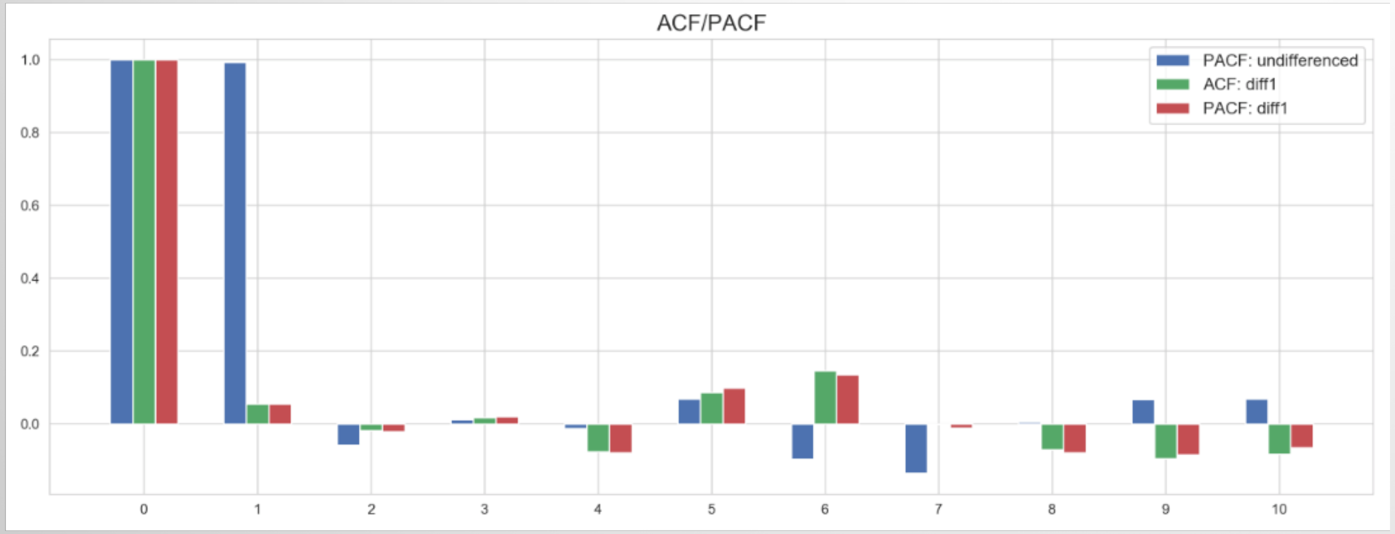


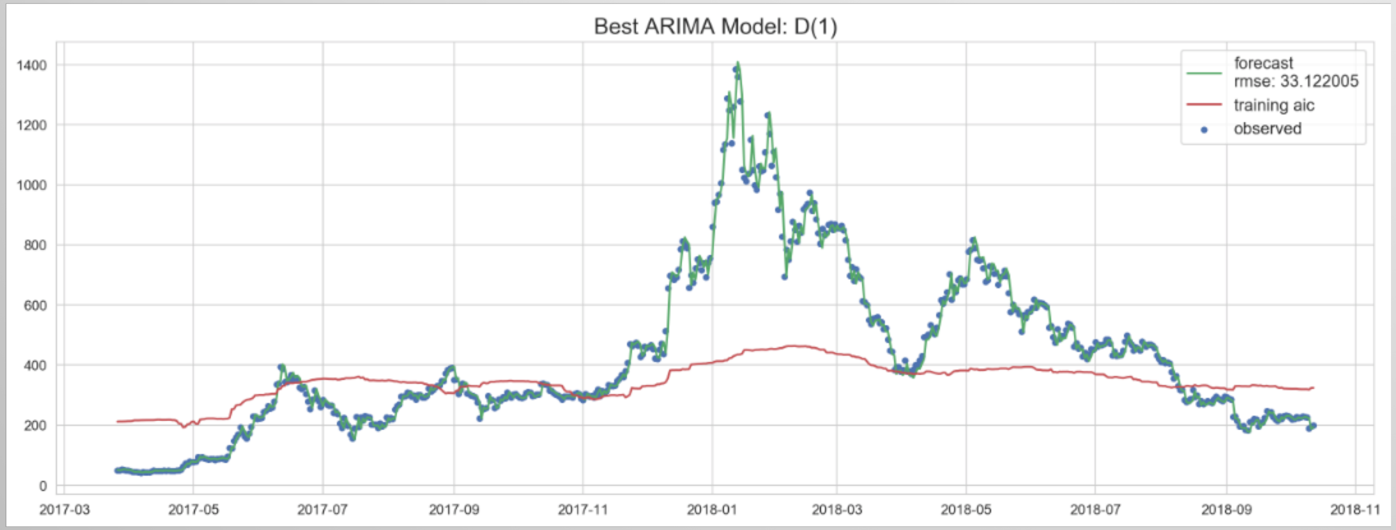

```
1st order differenced ADF test results

price:
adf -3.7284
pval 0.003731342278485783
critical vals {'1%': -3.4415777369651717, '5%': -2.866493255736561, '10%': -2.569407951640003}

rolling mean:
adf -3.9308
pval 0.0018194655296436873
critical vals {'1%': -3.442383534871275, '5%': -2.8668480382580386, '10%': -2.569597004924258}
```

- Structural breaks used for analysis:
  - Determined significant Prophet changepoints as ones with delta magnitudes greater than the mean of all deltas
  - Combined with changepoints detected by PELT, excluding changepoints close enough to likely be duplicates
- Window size selection:
  - Calculated the mean & median regime length, excluding initial regime (beginning of series)
  - Mean impacted by outlier regime lengths
  - Set window size to median regime length
- Stationarity:
  - Stationarity achieved using first degree differenced time series
  - ADF test results indicate both series and rolling mean (set to window size) are stationary

3. Time Series Forecasting

ARIMA Forecasts





- Determined ARIMA order using autocorrelation and partial autocorrelation functions
  - D = 1
  - Perhaps a slight AR/MA component to series, included in ARIMA orders to test

- ARIMA Ethereum price forecasts & RMSE scores using (p,d,q) orders:

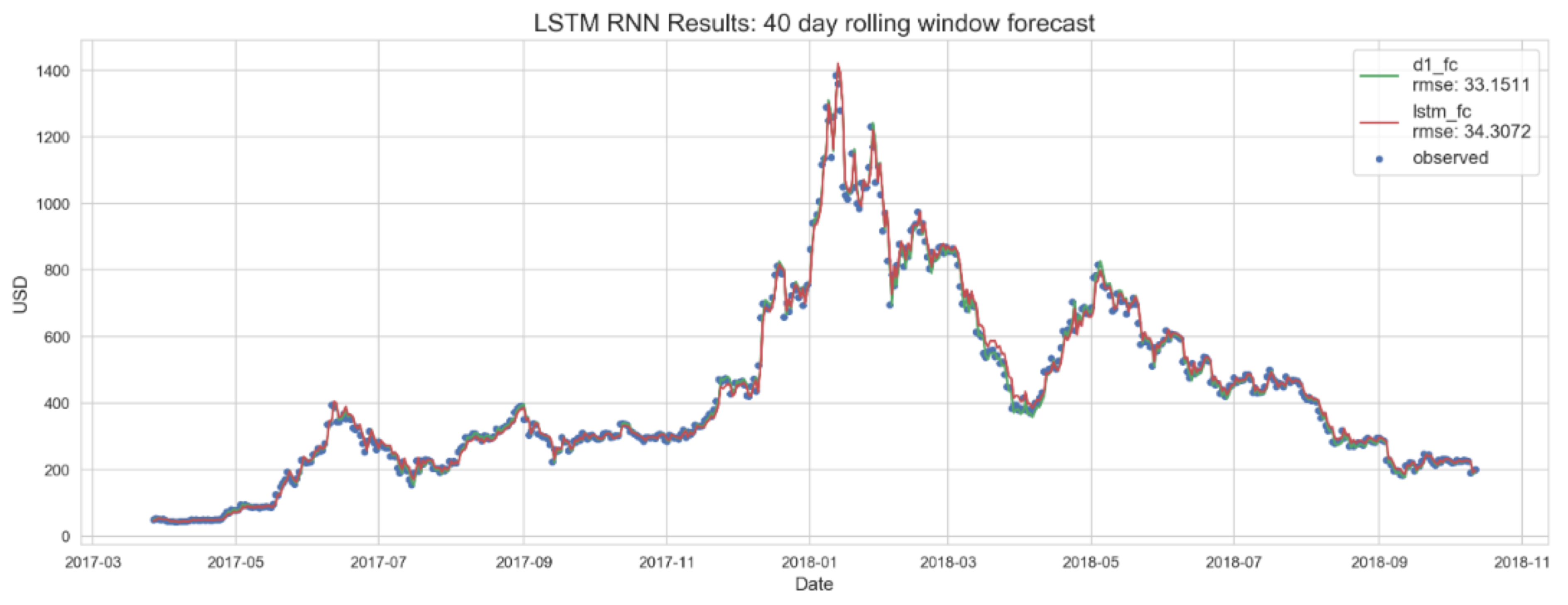
ARIMA(p, d, q)	RMSE
ARIMA(0, 1, 0)	33.122005
ARIMA(1, 1, 0)	33.853124
ARIMA(0, 1, 1)	33.983999

- Best ARIMA model: ARIMA(0,1,0)

LSTM RNN Setup

- While many deeper RNN configurations were tried , a simple single LSTM layer with dense output layer performed best, trained quickest, and was most consistent
  - Stacking LSTM layers and/or adding non LSTM layers (predictably) increased computational demand but did not outperform the single LSTM layer configuration
  - Adding dropout layers only produced inconsistent and unimproved RMSE
- LSTMs compiled & fitted using mean squared error loss function & RMSprop, Adam, Adamax, Adagrad, Adadelta optimizers
  - Excluded optimizers: SGD, Nadam (SGD slow, both returned consistently high RMSE)
- Some light hyperparameter tuning was performed using hyperopt with hyperas wrapper
  - First: with optimizer learning rates set to default, hyperopt tuned LSTM layer’s activation function, output dimension, and layer bias vector hyperparameters
  - Next: the same hyperparameters were tuned along with optimizer learning rates, once each using TPE search and random search algorithms
    - TPE (Tree-Structured Parzen Estimator) search: sequentially constructs models to approximate the performance of hyperparameters based on historical measurements, then chooses new hyperparameters to test based on this model
    - Random search: tests randomly sampled hyperparameter tuples, done uniform at random within specified hyperparameter constraints

LSTM RNN Forecasts



Optimizer	RMSE LR Default	RMSE LR TPE	RMSE LR Rand
RMSprop	34.654522	37.968861	35.384001
Adam	35.480750	34.964693	34.940433
Adamax	34.568555	35.264852	50.691526
Adagrad	35.252480	34.307214	59.761588
Adadelta	39.469572	45.130393	41.507783

- Best performing LSTM:
  - Adagrad, TPE search tuned learning rate (0.821336…)
  - Adagrad default LR = 0.001
- Overall best model:
  - ARIMA D1 (RMSE 33.122005)

4. Exogenous Variables

Exogenous Variable Groups

- Ethereum related
  - OHLCV: Open, High, Low, Close (target), VolumeFrom (in terms of ETH), VolumeTo (in terms of USD)
- Other cryptocurrency prices
  - BTC (Bitcoin): the first decentralized cryptocurrency, bitcoin is the genesis of the current cryptocurrency landscape
  - XRP (Ripple): real-time gross settlement system with the goal to enable "secure, instantly and nearly free global financial transactions of any size with no chargebacks."
  - EOS (EOS.IO): smart contract platform and decentralized operating system intended for the deployment of industrial-scale decentralized applications
  - LTC (Litecoin): technically similar to BTC, except provides faster transaction confirmations (2.5 minutes on average) and uses a memory-hard, scrypt-based mining proof-of-work algorithm to target the regular computers and GPUs most people already have
  - XLM (Stellar): open-source, decentralized protocol for digital currency to fiat currency transfers which allows cross-border transactions between any pair of currencies
  - XMR (Monero): in contrast to most coins, public ledger is obfuscated which allows users to broadcast or send transactions, but no outside observer can tell the source, amount or destination
- Other economic indicators
  - VIXCLS (CBOE Volatility Index): measure of market expectation of near term volatility conveyed by stock index option prices
  - TWEXB (Trade Weighted US Dollar Index: Broad): weighted average of the foreign exchange value of the U.S. dollar against the currencies of a broad group of major U.S. trading partners
  - EFFR (Effective Federal Funds Rate): volume-weighted median of overnight federal funds transactions

Granger Causality

- Granger causality tests: statistical hypothesis tests for determining whether one time series is useful in forecasting another
  - Tests the ability to predict the future values of a time series using prior values of another time series
  - Predictive (or probabilistic) causality: A time series X is said to Granger-cause Y if it can be shown through a series of t-tests and F-tests on lagged values of X (and lagged values of Y), that values of X provide statistically significant information about future values of Y
- The following exogenous variables were determined by Statsmodels’ Granger causality implementation to one-way granger cause ETH price:
  - ETH VolumeFrom:

```
eth_volumefrom
{'ssr_ftest': (4.798422088361457, 0.028868566532146988, 602.0, 1), 'ssr_chi2test': (4.8223344
90795152, 0.028093271759724894, 1), 'lrtest': (4.803217077164845, 0.028406645758713257, 1),
'params_ftest': (4.798422088358041, 0.028868566532203706, 602.0, 1.0)}
```

  - XRP Close (Ripple):

```
xrp_close
{'ssr_ftest': (23.825045546095822, 1.3530106329889756e-06, 602.0, 1), 'ssr_chi2test': (23.943
775008950116, 9.919049425225322e-07, 1), 'lrtest': (23.482111096553126, 1.2608064710533996e-0
6, 1), 'params_ftest': (23.825045546095996, 1.353010632988829e-06, 602.0, 1.0)}
```

  - LTC Close (Litecoin):

```
ltc_close
{'ssr_ftest': (10.791241869535185, 0.001079029384691407, 602.0, 1), 'ssr_chi2test': (10.84501
8822373401, 0.00099061872692401, 1), 'lrtest': (10.748963021378586, 0.0010433776657996791,
1), 'params_ftest': (10.791241869535188, 0.001079029384691407, 602.0, 1.0)}
```

  - XMR Close (Monero):

```
xmr_close
{'ssr_ftest': (8.069375762437181, 0.004654222868574518, 602.0, 1), 'ssr_chi2test': (8.1095885
98462615, 0.0044031718718594145, 1), 'lrtest': (8.055717870614899, 0.004536026615660543, 1),
'params_ftest': (8.069375762437426, 0.004654222868574171, 602.0, 1.0)}
```

Exogenous Variable Forecasts

- Procedure: variables were provided to models as individual features, then as a feature set including all Granger causal variables
- Forecasts were produced by the best performing ARIMA model and best 3 performing LSTM configurations
  - Note: LSTM performance can likely be improved upon with feature set specific tunes, however time did not allow for each LSTM hyperparameter configuration to be re-tuned

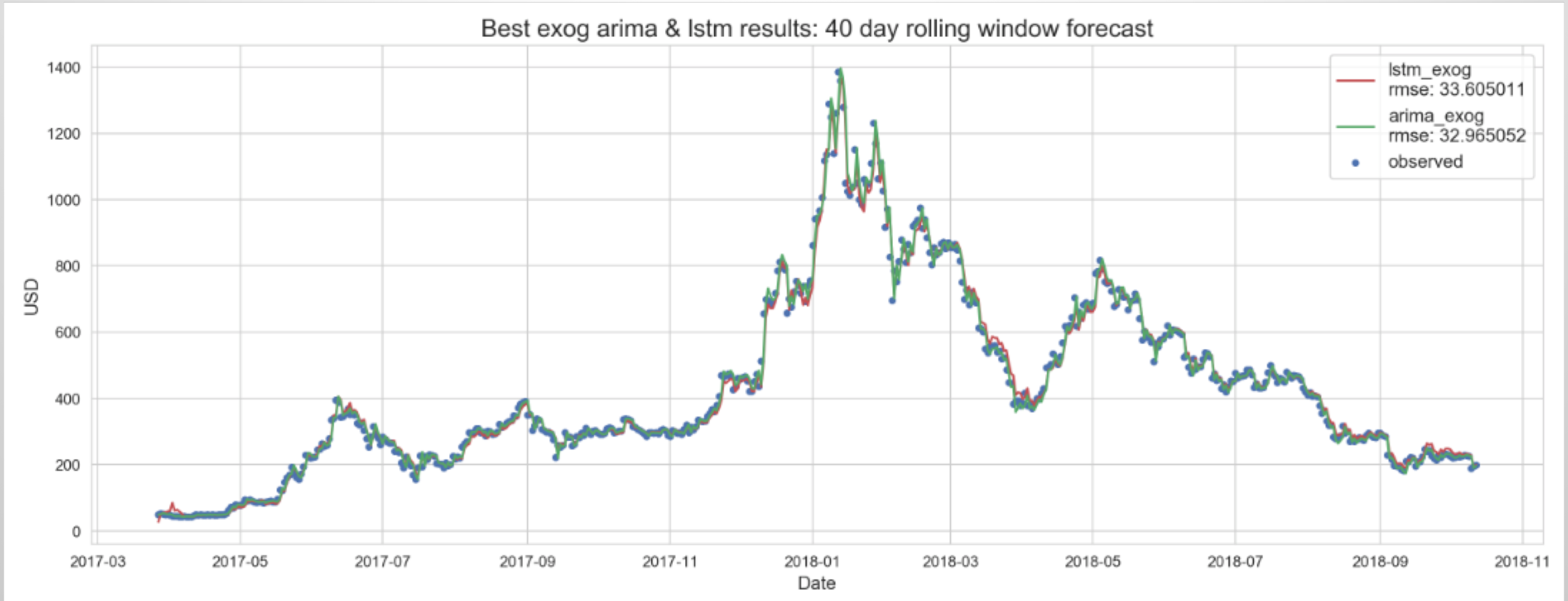
ARIMA (0, 1, 0)	RMSE
Time series only	33.122004
ETH Volume From	32.965052
XRP Close	33.250483
LTC Close	33.127093
XMR Close	33.177369
All GC variables	35.989936

LSTM (Adagrad)	RMSE
Time series only	34.307214
ETH Volume From	36.949516
XRP Close	33.605011
LTC Close	40.952003
XMR Close	38.685143
All GC variables	48.069971

LSTM (Adamax)	RMSE
Time series only	34.307214
ETH Volume From	35.975974
XRP Close	34.791382
LTC Close	44.305915
XMR Close	38.583072
All GC variables	59.335610

LSTM (RMSprop)	RMSE
Time series only	34.307214
ETH Volume From	37.091043
XRP Close	38.910223
LTC Close	47.219267
XMR Close	41.075051
All GC variables	97.369460

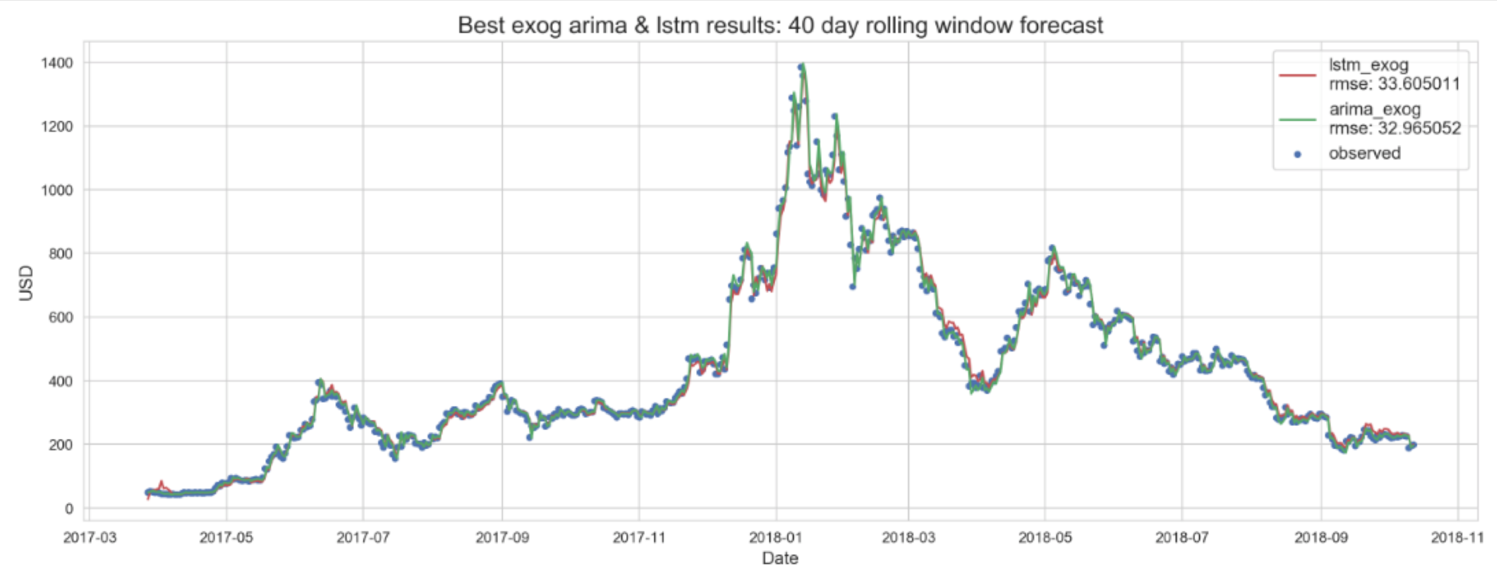
5.3 Exogenous Forecast Results



- Exogenous data was able to improve both ARIMA & LSTM models
  - However, the data they benefitted from was not the same
  - ARIMA: ETH Volume From (Volume expressed in ETH)
  - LSTM: XRP (Ripple) close prices (strongest Granger causality)
  - LSTM improved upon itself, but still had higher RMSE the time series only ARIMA (0,1,0) model

5. Overall Results and Next Steps

Best Models



- Lowest RMSE: ARIMA (0, 1, 0) with ETH Volume From exogenous data, RMSE 32.965052
  - Previous days' ETH trade volume contains information about future price movements
  - ARIMA was unable to improve using the strongest Granger-causal exogenous driver
  - Exogenous data was not analyzed to a great extent: existence of trends/non-stationarity may negatively impact ARIMA's ability to use this data
- Best LSTM: Adagrad (TPE tuned) with XRP (Ripple) close price, RMSE 33.605011
  - LSTM was able to use XRP data to explain variance in ETH prices
    - Did not improve using ETH trade volume as the ARIMA model did
    - ARIMA w/ XRP resulted in higher RMSE (33.250483) than original time series model, but this score is lower than the TPE Adagrad LSTM model's RMSE
  - Caveat: these LSTM tunes are likely suboptimal, as networks were tuned only using past ETH price data

Next Steps

- Perform more extensive LSTM hyperparameter tuning
- Expand the use exogenous variables
  - Crypto market news/headlines
  - Include more exogenous coin data
  - Use different measures of economic activity and uncertainty
  - Perform more intensive dependent variable analysis and feature engineering on exogenous variables as necessary
- Experiment with forecasting techniques:
  - Combination recursive & rolling window forecasting techniques
  - Investigate more formal methods for window size selection

Thank you!

Slides and Python notebook available at:
github.com/brianmcguckin/thinkful_final_capstone