Modeling of Cryptocurrency Prices

Part 1. Background

Are you trying to match a curve or are you trying to model a phenomenon?

A recent well-written hackernoon post by Rafael Schultze-Kraft [1] described his effort to match and predict the recent wild swings in cryptocurrency prices. He utilized a multidimensional long short term neural network to predict the price of a cryptocurrency – using Bitcoin as the well-known example. His work is excellent but it stops short of making a prediction of the price of Bitcoin for the next week or next month. It seems that the neural network results are more of a curve matching exercise rather than the derivation of a model of behavior. And, as self-written, "the fundamental flaw with this model is that *for the prediction of a particular day, it is mostly using the value of the previous day*".

Is this a model that you would want to trade on or is it a "price following" model based upon the previous day's closing price? In the author's own words, "However, no matter how accurate the predictions are in terms of the loss error—in practice, the results of single-point prediction models based on *historic price data alone*, as the one showcased here, remain hard to accomplish and are not particularly useful for trading".

If you've derived a curve which matches past performance, then how do you extrapolate the future performance? Do you linearly extrapolate, quadratically extrapolate, or is there some other built-in method for extrapolating into the future built in to the neural networks? Or, inherent in the neural network modeling, are there assumptions built in to the external forcing functions (or external stimuli) that can be re-patterned to predict future prices of a neural network derived model?

Part 2. Model Derivation

On the other hand, why not try to derive a model that matches past performance and can be utilized to extrapolate into the future?

Figure 1 shows the model as a "black box" which takes inputs and produces outputs. Sounds like a neural network, doesn't it? But for a neural network, what are the inputs? More specifically, what are the inputs to be used for future predictions?

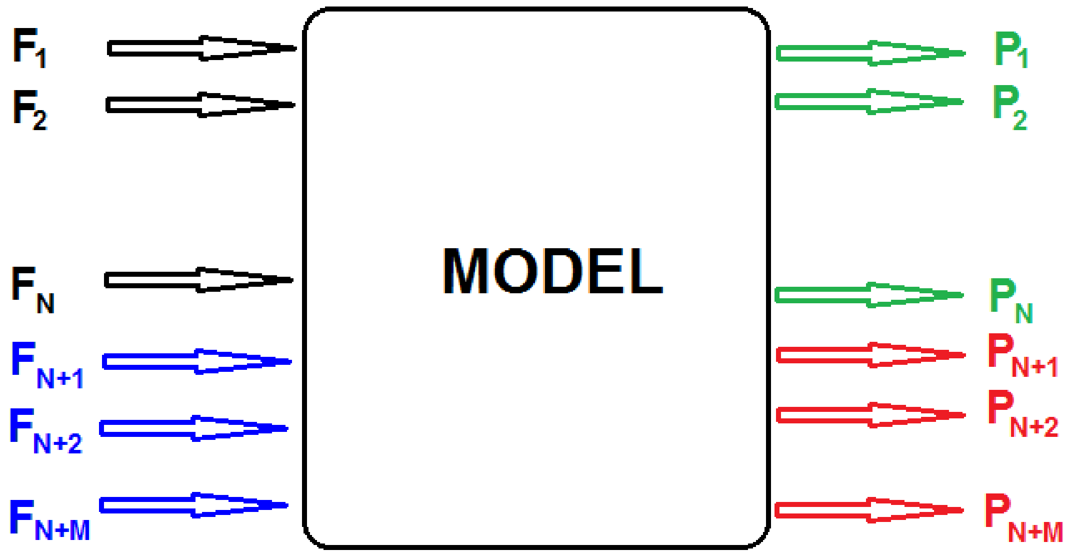

Figure 1. Black Box Model of a Process

This work assumes that the “black box” is a parallel series of second order linear differential equations. For each differential equation, there is a natural frequency associated with it, a damping value for the natural frequency, and a component of force that excites that specific frequency. This “black box” model is analogous to a structural dynamic system, such as a bridge vibration or diving board vibration, where natural frequencies and damping values exist and the resulting response is determined by these natural frequencies, damping values, and the forcing function component for each of these natural frequencies (or modes). We’ve all seen the oscillatory or vibratory response of a system as shown in Figure 2 where different natural frequencies and damping values lead to different responses at each point in time.

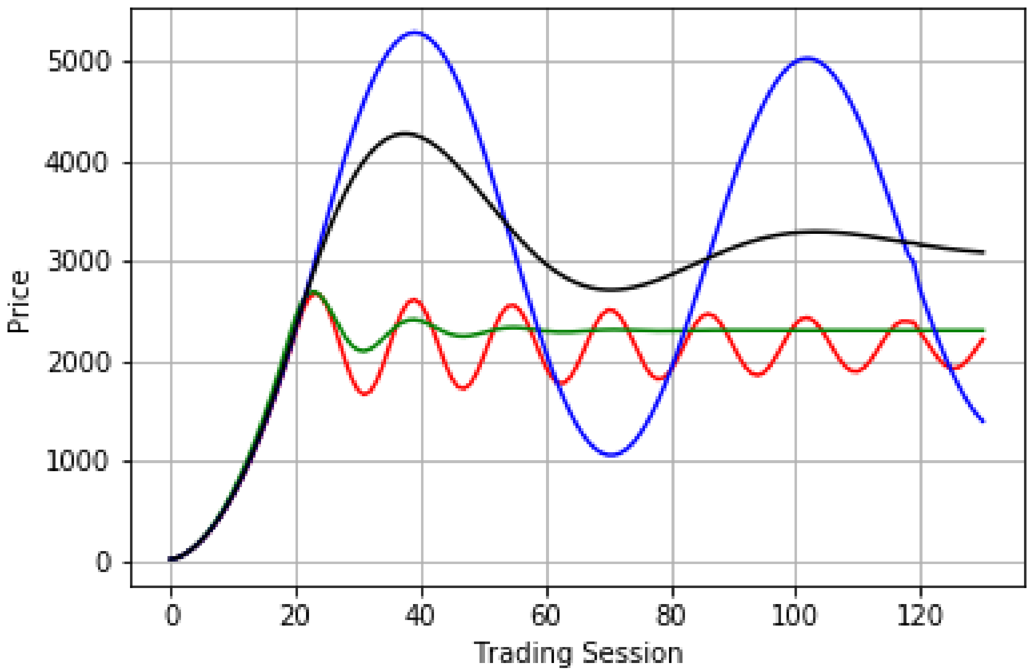

Figure 2. Oscillatory Response of Second Order Systems

In reality, physical phenomenon are modeled as a summation of many modes of oscillatory response. And that is what we are going to do with the price of cryptocurrency as a function of time. As a side note, this method has been used by the author with good success in predicting stock index values for the future though cryptocurrency prices seem to change much more rapidly than stock index prices.

Starting with the past performance price of a cryptocurrency, we randomly generate a set of natural frequencies and damping ratios for each of these natural frequencies (or natural modes). We also assume a set of forcing function values derived from a price quasi-acceleration method. And then we let an algorithm derive better natural frequencies, damping ratios, and forcing function values to match the past performance curve.

For the work described herein, we used two different algorithms to improve the model so that it best matched the past performance. In the first case, we used a genetic algorithm to get into the ballpark and then we switched to a gradient-based method to obtain the final results. In the second case, we used a pure gradient-based method to obtain the final results right from the beginning.

For stock market index prices the hybrid genetic/gradient based method seemed to work best. For cryptocurrency prices the purely gradient based method seemed to work best. We still need to investigate what is the basis of this discrepancy in best methods to obtain a satisfactory solution.

After a sufficient number of iterations the purely gradient-based method obtained a derived model that matched the past performance of Bitcoin closing price. This match is shown in Figure 3. To show that this is not Bitcoin specific, Figure 4 shows the derived model results for Ethereum cryptocurrency. As a benchmark, three different cryptocurrency price models were derived over less than 24 hours of computing time using a an older Pentium Core 2 Duo CPU running at 2.93 GHz on a 32 bit operating system..

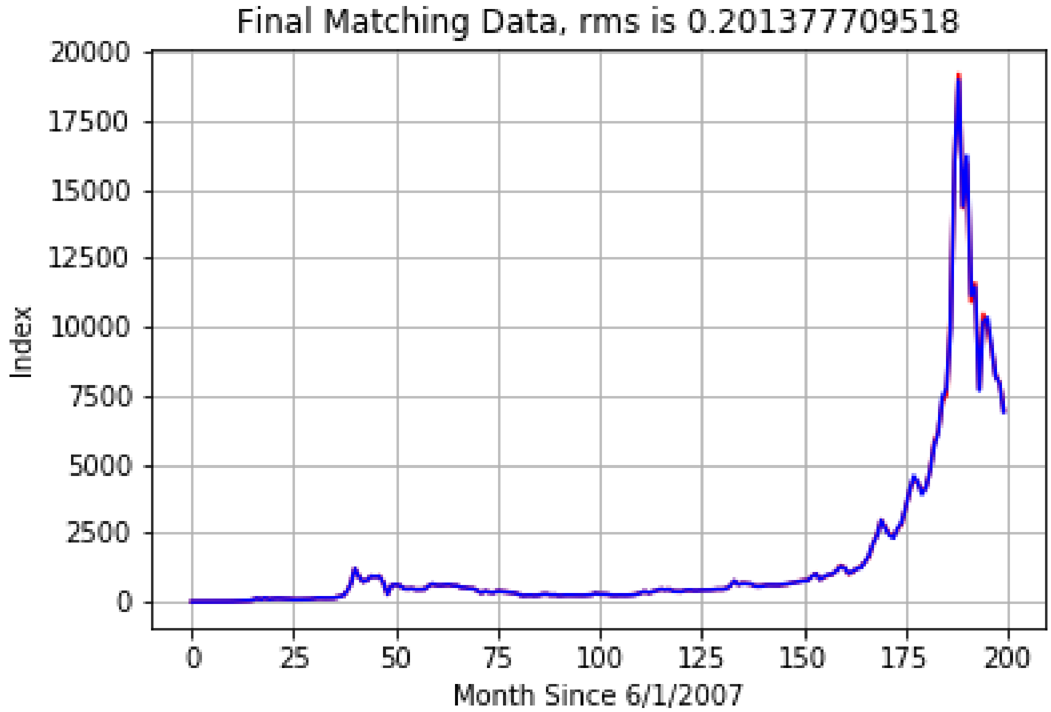

Figure 3. Bitcoin Closing Price Actual and Model-Based Derivation

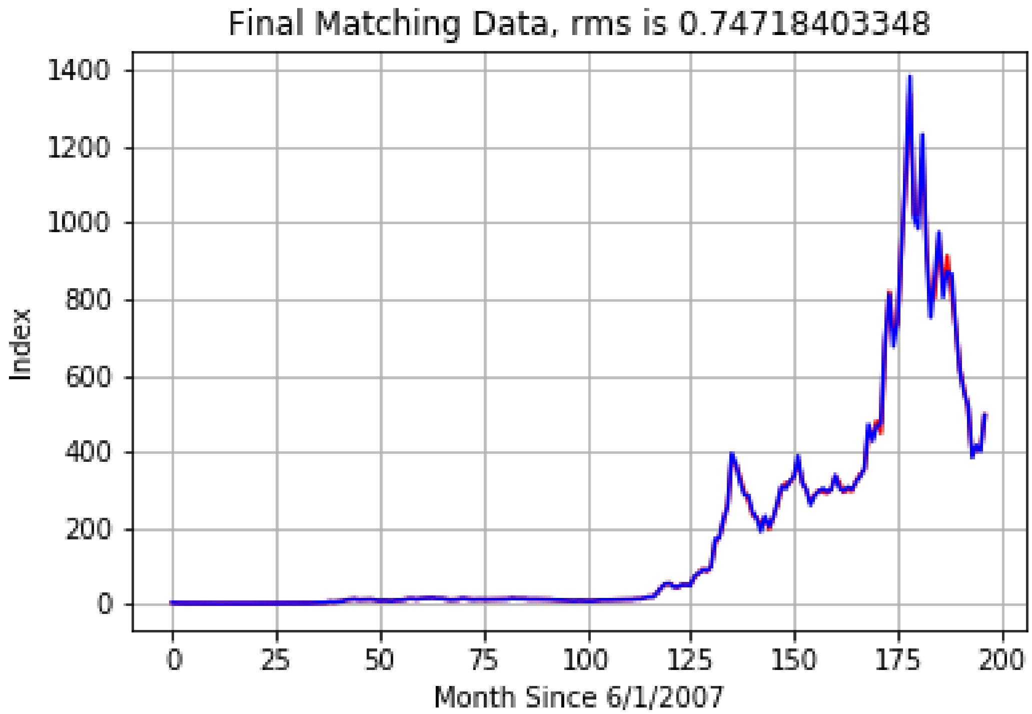

Figure 4. Ethereum Closing Price Actual and Model-Based Derivation

Thus we have a derived a post-predictive model of the closing price of Bitcoin and Ethereum based upon a model rather than a pure curve matching. Now we have to try and utilize the models to predict the future prices of Bitcoin and Ethereum.

Part 3. Predicting the Future

Thus far we've derived models that take external stimuli or forcing functions and yield matching prices to actual cryptocurrency closing prices. These models were based upon linear second order oscillators with a number of natural frequencies and damping values.

Now to predict future prices, we simply need to run the model for additional trading sessions with appropriate external stimuli or forcing functions. There's the catch! What are the values of these external stimuli for each upcoming trading session?

In reality, the external stimuli for the stock market, for example, could be based upon future earning, expectations of moves in the Federal Reserve's discount rate, geopolitical events such as another North Korea missile test, and a whole host of other events or even rumors. And for the stock market there is enough history that we know the timing of many of these events, such as Federal Reserve meetings, and can use future stimuli based upon past stimuli for each month.

This may not be true for cryptocurrency. We don't have enough past history to draw from. And, in fact, as you saw in the Bitcoin closing price, only the last year or so of trading saw wild swings worth predicting or trying to make a profit off.

For reference sake, here is the closing price of Bitcoin and the matched second order system model. The curves pretty much pay on top of one another.

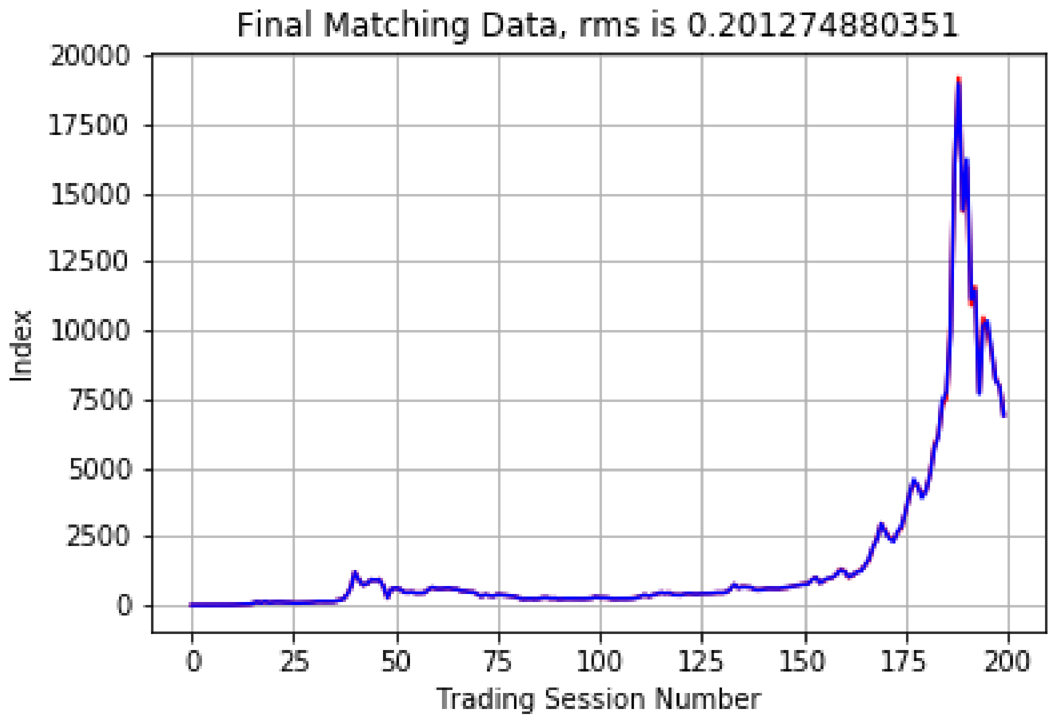

Figure 1. Bitcoin Closing Price Actual and Model-Based Derivation

To generate this model, Figure 2 shows a look at the external stimuli (or forcing function) for each trading that the gradient-based model derivation algorithm converged to.

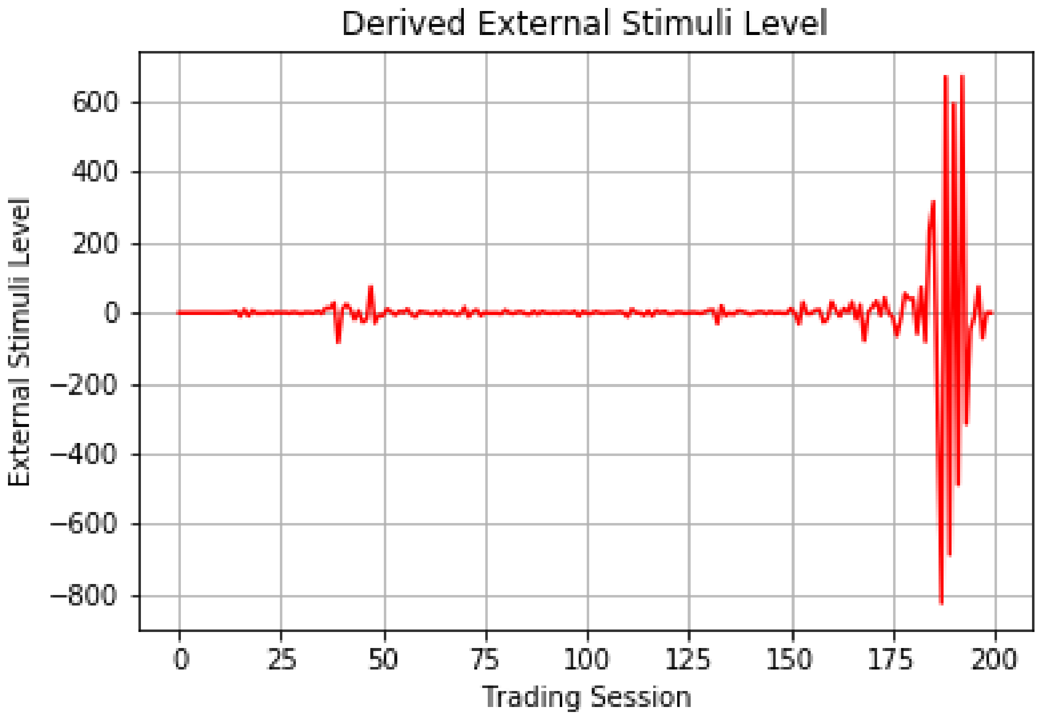

Figure 2. External Stimuli for Each Bitcoin Trading Session

Note that in the early days the external stimuli was small and there were no significant changes in Bitcoin closing prices. This is expected. But then the required external stimuli started oscillating wildly in order to match the Bitcoin closing price.

So now we have to make a rational estimate as to what the future external stimuli will be in order to predict the price of the cryptocurrency.

Part 4. Predicting the Future (For Real?)

Thus far we've derived models that take external stimuli or forcing functions and yield matching prices to actual cryptocurrency closing prices. These models were based upon linear second order oscillators with a number of natural frequencies and damping values.

References

1. "Schultze-Kraft, Rafael, "Don't be fooled – Deceptive Cryptocurrency Price Predictions Using Deep Learning", 16 April 2018, https://hackernoon.com/dont-be-fooled-deceptive-cryptocurrency-price-predictions-using-deep-learning-bf27e4837151.
2. A