BitPred: A Crypto Price Predictor

Presented by

Kanishka Pandey
Roll no. – E18CSE083
Batch – EB03
Semester 7

A web application utilizing a trio of SARIMAX, FB Prophet, and custom LSTM networks to predict future trends in Bitcoin prices with more currencies on the way.

Submitted
to

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
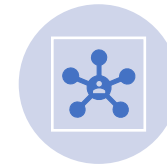## BENNETT UNIVERSITY, GREATER NOIDA, 201310, UTTAR PRADESH, INDIA

OUTLINE

Introduction & Background

Project Problem Statement

Social Importance and Relevance of the Project

Business Context

Proposed Solution

Experiment and Implementation

Results Analysis and Validation

Conclusion and Future Work

References

Introduction & Background

This project draws on the knowledge I have accumulated in three years of working on machine and deep learning technologies.

The goal here was to learn about the principles and working of blockchain technology.

Aiming to capitalize on the cryptocurrency trend.

It primarily uses convolutional and recurrent neural network models to implement long short-term memory cells and gated recurrent units.

Project Problem Statement

I realized that none of the popular crypto exchanges have robust analysis systems that could help a new user to gain insights into market trends and past prices.

Users are left to incur losses as they slowly learn to adapt to the different patterns the crypto market works on as compared to traditional stock prices.

This is the main problem my project aims to solve; provide all users with a service/application framework they can easily access to gain insight into crypto market data.

BENNETT UNIVERSITY
TIMES OF INDIA GROUP

User Design Canvas

Problems

- User is not aware of the current state of the market
- Cannot predict trends that govern the cryptocurrency prices
- Does not have information on factors that control the price ranges at different times of year

Motives

- Investors need past price values to make educated guesses
- Analysis of sharp rises and drops must be performed to reduce volatility
- Users looking for easy-to-use GUI elements and not technical data

Fears

- Investing based on abstract models is risky
- Crypto prices rarely follow ideal time series projections
- Confidence intervals look good on paper but may lead to economic loss

Application

- BitPred aims to be a one-stop solution to all the problems faced by both new and veteran investors while buying assets on the cryptocurrency market. It will help analyze past prices and make accurate predictions to reduce risk factors affecting portfolios.

Users

- Hobby Investors (educating themselves on recent trends)
- Crypto Architects and Financial Consultants (use it to inform their clients)
- Fintech Startups and Neobanks (Train trading bots and offer investment opportunities)

Unique Value Proposition

- Provide all users with an interactive UI that compiles data and results from multiple sources to provide accurate analyse and predictions.

Competitive Advantages

- Best possible results returned after comparing data from 3 models
- Easy-to-understand Python workflow and Flask app with intuitive GUI
- Entire development server ready to deploy using cloud providers

Alternatives

- Bitvision - a command line predictor and trading bot
- Kriptomist - fundamental analysis instead of past trends to predict prices
- Consult directly with crypto developers and experts to form own opinion
- Use own data models without UI to make technical predictions

Solutions

- Clean app UI to avoid confusion during use
- Use standard Flask and CustomJS commands to increase portability across platforms
- Test both popular general models like fbprophet and custom RNNs to get accurate predictions
- Fully interactive plots via Bokeh and Plotly
- Intuitive UI elements via button clicks, drop-down menus and full gregorian calendar support
- Multilayered, Multicolored graphs with tab selection for all three models

Social Importance and Relevance of the Problem

- The current crypto market is one of great interest worldwide, being advertised as a highly disruptive technology poised to alter the way we conduct daily transactions.

- This project aims to be highly accessible, such that it could be used both by early investors in crypto as well as new users that are just now jumping on the bandwagon.

- Potential users of this application include:
  - Hobby Investors, Blockchain Architects, Financial Consultants, Fintech Startups and 'Neobanks'.
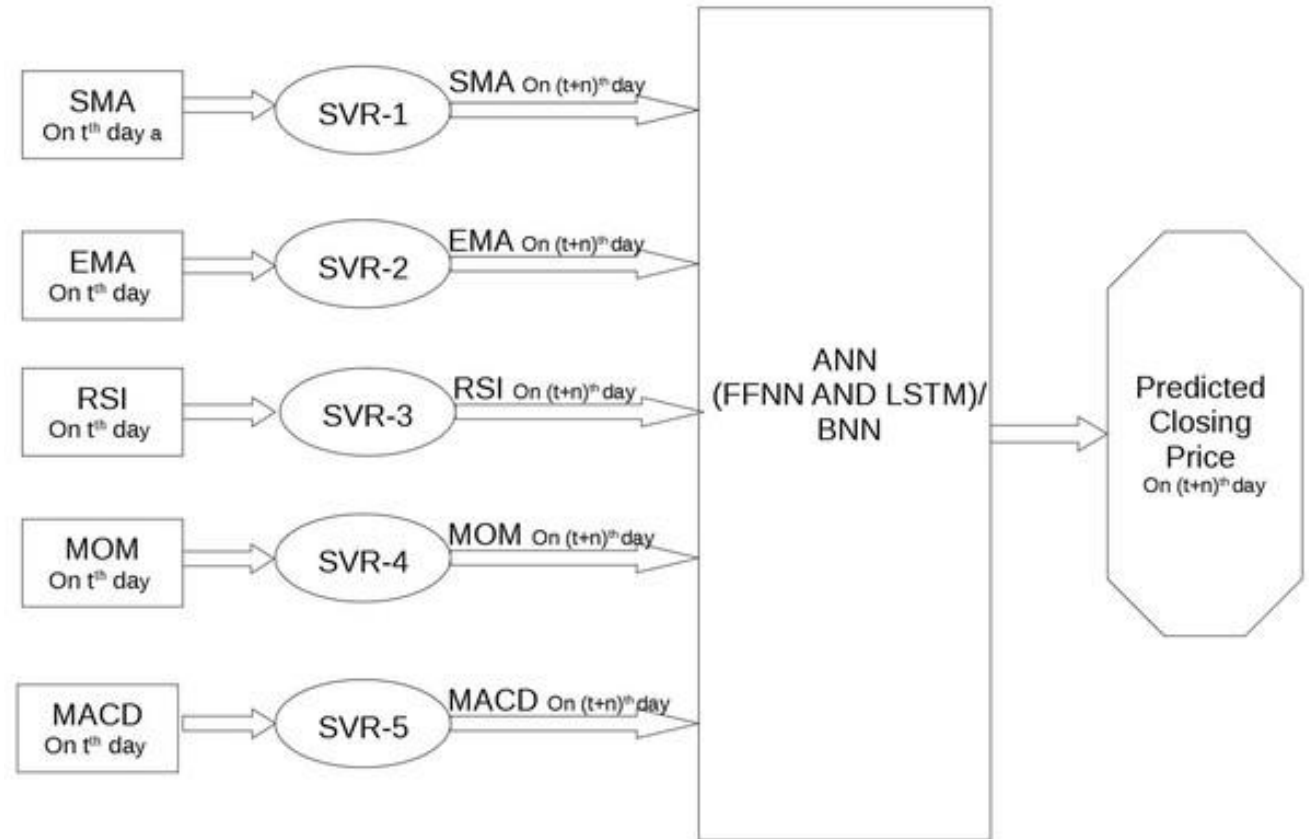
Business Context

Application Domain - The project falls under the decentralized finance domain of applications

Stakeholders - The major stakeholders in this project may include the end-user, the developer and middle-men like startups and consultants

Drivers - Crypto prediction packages are still a relatively new application of popular ML techniques

Constraints - The current iteration of the project is limited by the performance of time series prediction models
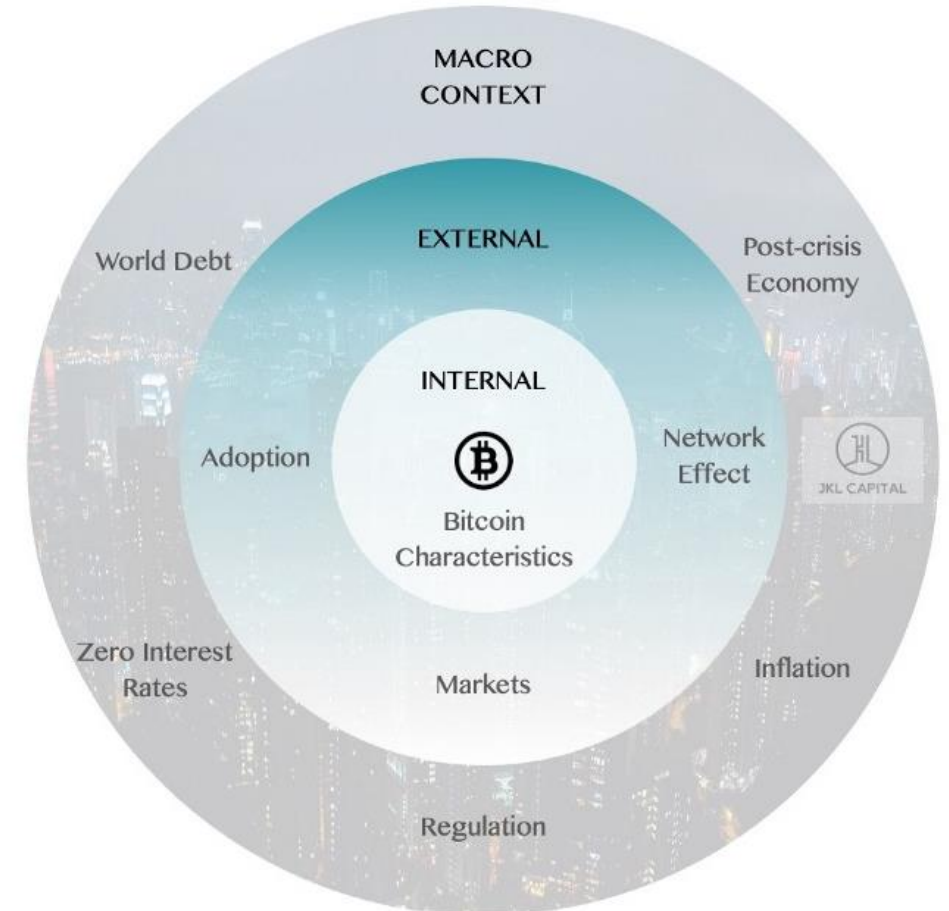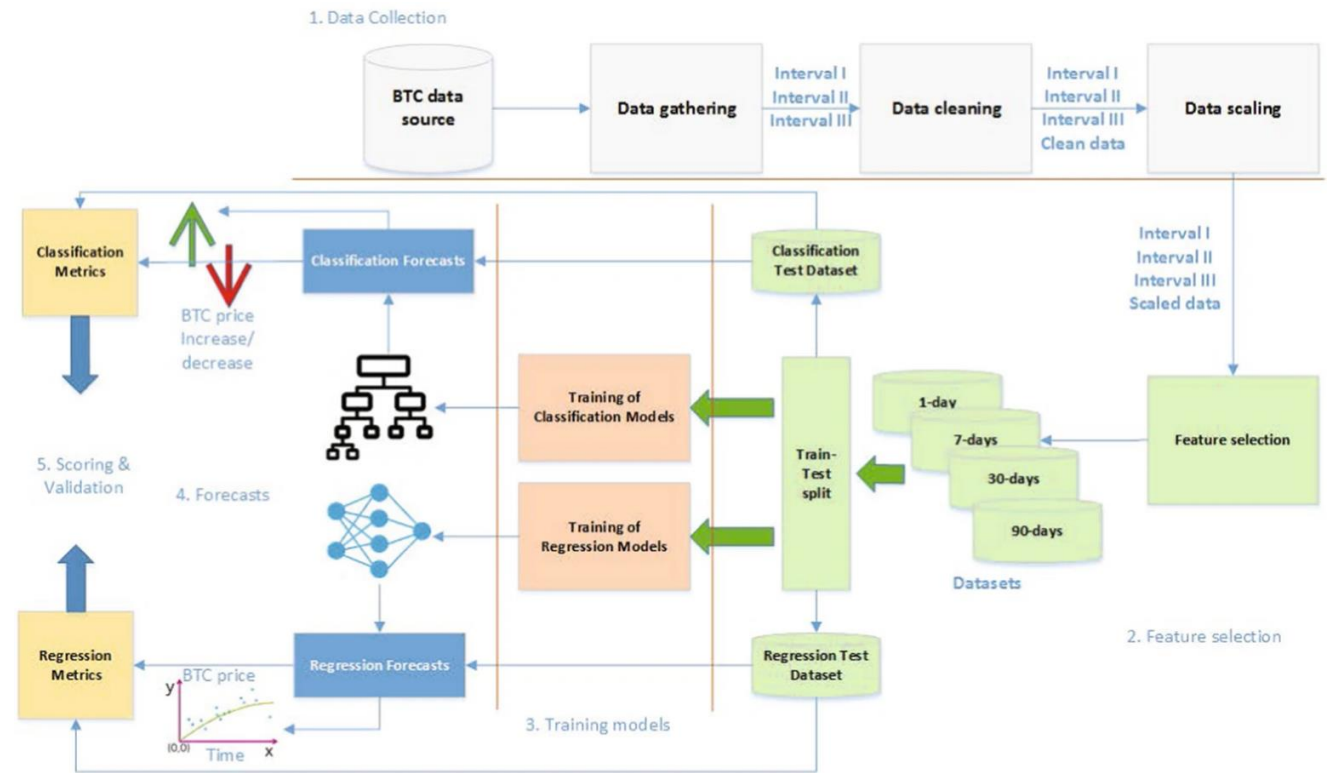
Proposed Solution

- The goal is to create an interactive UI.

- The application utilizes a python flask server.

- Asks for user time period preferences and computes prediction in real-time.

- It shows results returned by three models: SARIMAX, Facebook Prophet, and my custom LSTM network.

- The application dataflow and frontend sequence diagrams are shown
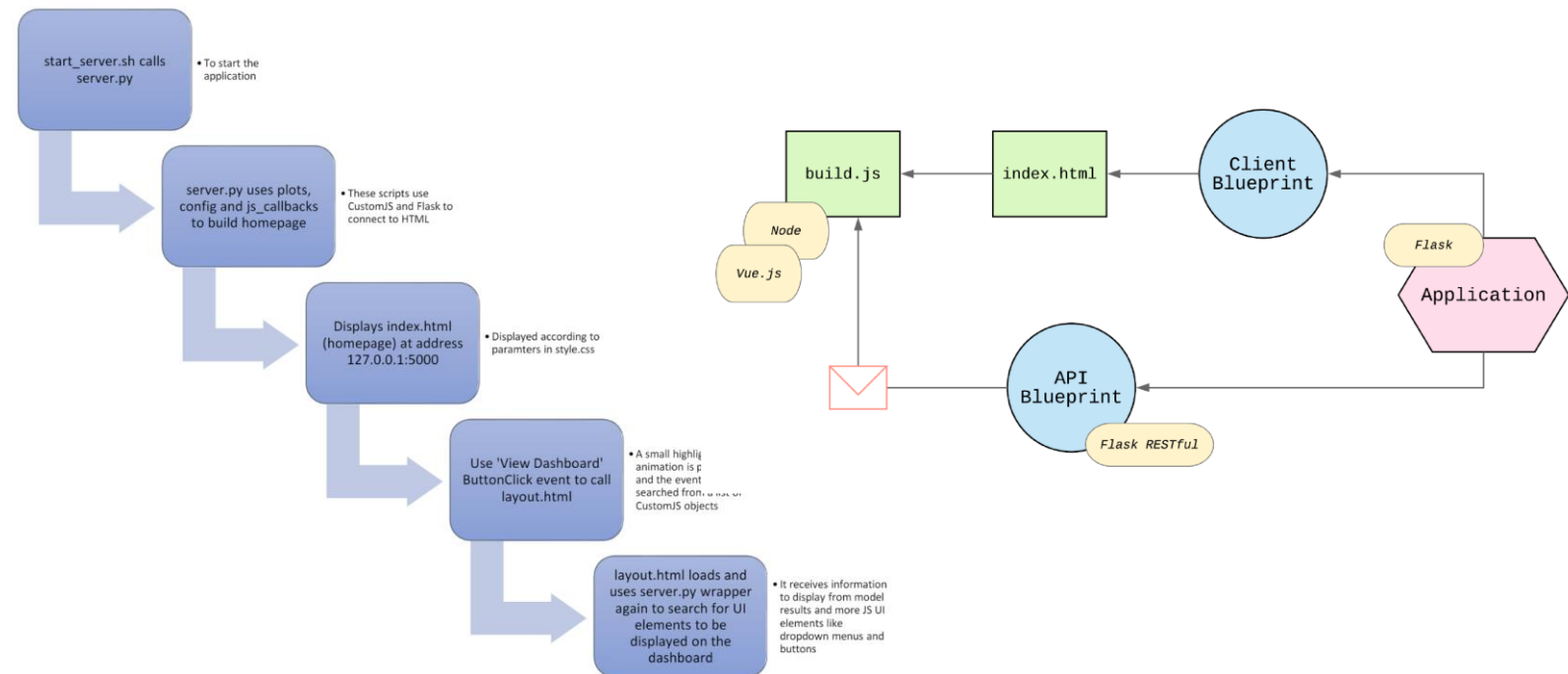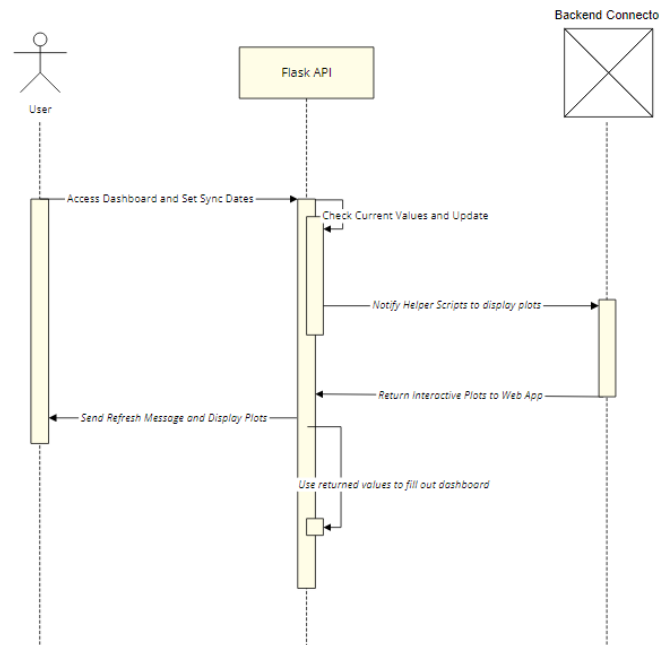
Proposed Solution Contd.

- To retain the advantage of using past information, my custom model uses long short-term networks that are well-suited to any time-series data.

- Different CNN models were also used for additional analysis. Different iterations and training epochs tested.

- To avoid the dying RELU problem, a combination of tanh and leaky RELU activations used.
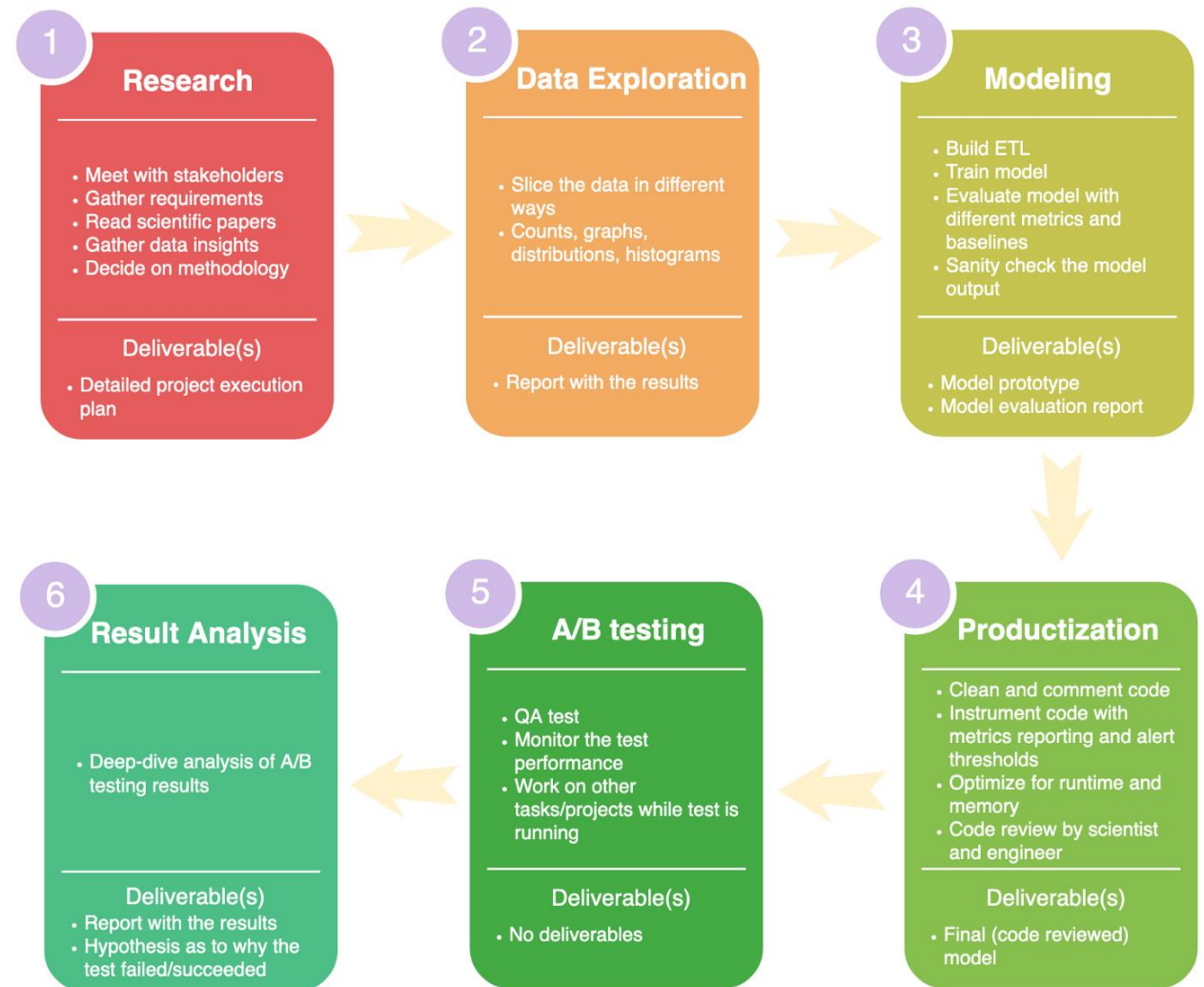
Proposed Solution Framework
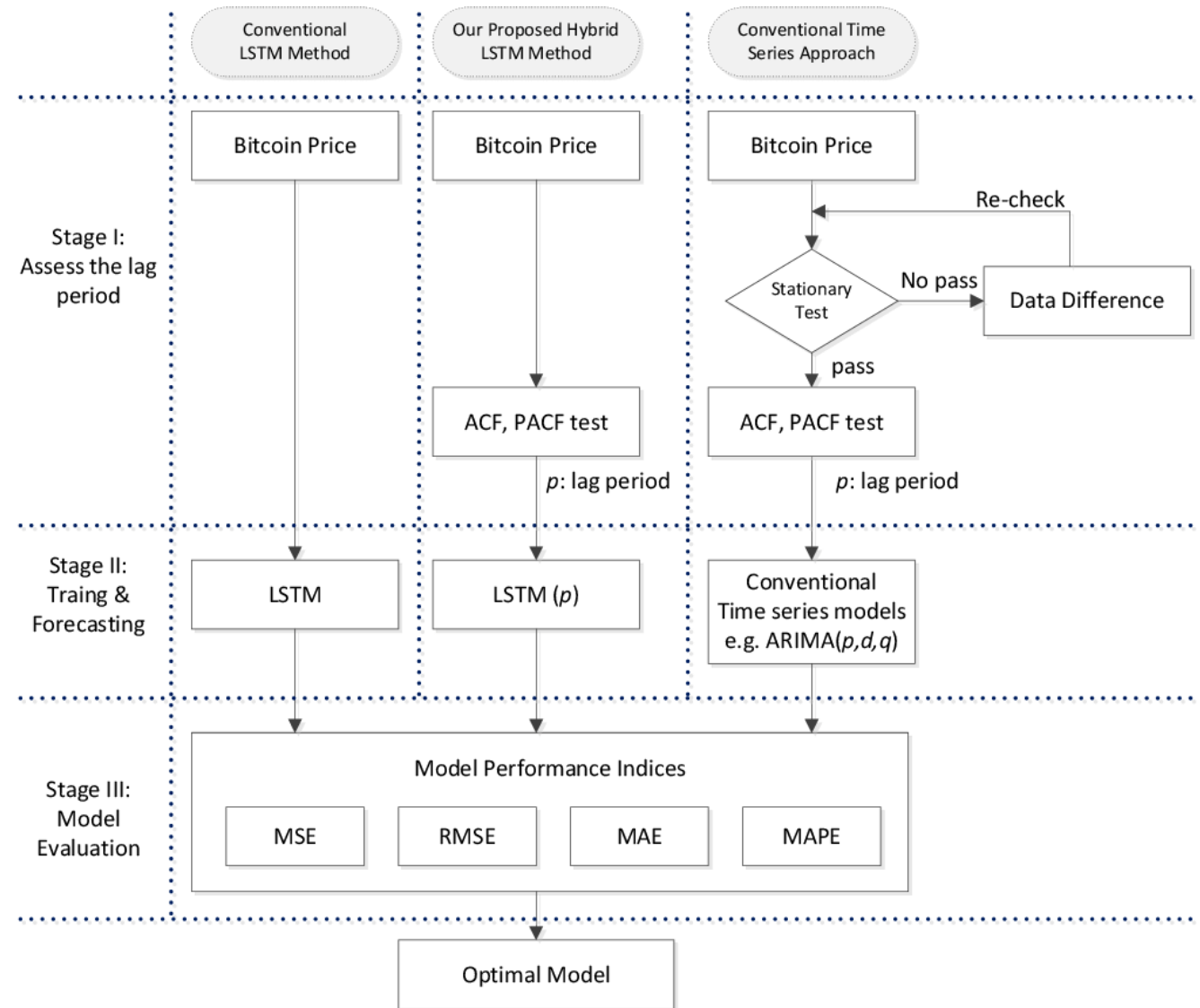
work for bitcoin price prediction

Timeline

- The first month was utilized in exploring possible project topics and shortlisting the ones which were relevant and had existing examples to build my proposal.

- After submitting the proposal, I started data collection from various sources before formatting them in a common style and storing them as fresh databases.

- Once the data was ready, the first half of October was used to train the first set of useful CNN models followed by further processing the data to make it accessible in multiple formats for different models.

- November was used to create and train a GRU-GAN hybrid model that enhances the insights obtained from the LSTM network.

- Then I worked on the mandatory documentation all the while doing hyperparameter tuning based on test loss results.

1 **Research**
- Meet with stakeholders
- Gather requirements
- Read scientific papers
- Gather data insights
- Decide on methodology

Deliverable(s)
- Detailed project execution plan

2 **Data Exploration**
- Slice the data in different ways
- Counts, graphs, distributions, histograms

Deliverable(s)
- Report with the results

3 **Modeling**
- Build ETL
- Train model
- Evaluate model with different metrics and baselines
- Sanity check the model output

Deliverable(s)
- Model prototype
- Model evaluation report

6 **Result Analysis**
- Deep-dive analysis of A/B testing results

Deliverable(s)
- Report with the results
- Hypothesis as to why the test failed/succeeded

5 **A/B testing**
- QA test
- Monitor the test performance
- Work on other tasks/projects while test is running

Deliverable(s)
- No deliverables

4 **Productization**
- Clean and comment code
- Instrument code with metrics reporting and alert thresholds
- Optimize for runtime and memory
- Code review by scientist and engineer

Deliverable(s)
- Final (code reviewed) model

Deliverables

All the necessary scripts, a single 'prediction' file, and a requirements document that allows the project to be run on any system with a single command call.

An easy-to-understand set of notebooks with proper documentation and code commenting that can be run on multiple platforms with minimal additional configuration.

Include graphical analyses wherever possible to ensure rapid delivery of important insights to all users.

Deliver relevant cryptocurrency prices to all users.

After considering multiple factors and trends, provide a reasonably accurate prediction of future changes in market prices.

Learn and apply more efficient functions that can be used in modern frameworks to accelerate learning and push the model towards real-time performance on all types of hardware.

Experiment and Hardware Implementation

As this project is a predictive model most of the requirements are freely available software resources. These include Python (2.7 or higher), Pandas, Tensorflow, Keras, Numpy, Scipy, Sklearn (for data handling and model creation), and h5py (for data storage).

- For the initial setup, clean data from Kaggle was needed. Later, public exchange platforms like Poloniex and Yahoo Finance were tapped.

- All the collation and retrieval of raw data was handled on PCI Express Gen. 4 NVMe SSDs installed on local machines to maximize the I/O bandwidth.

- Final compilation before saving was done on Colab to accelerate learning via its tensor cores that optimize all tensorflow matrix operations.

- Finally, the project can now be ported to any system that has access to past data to gain insights.
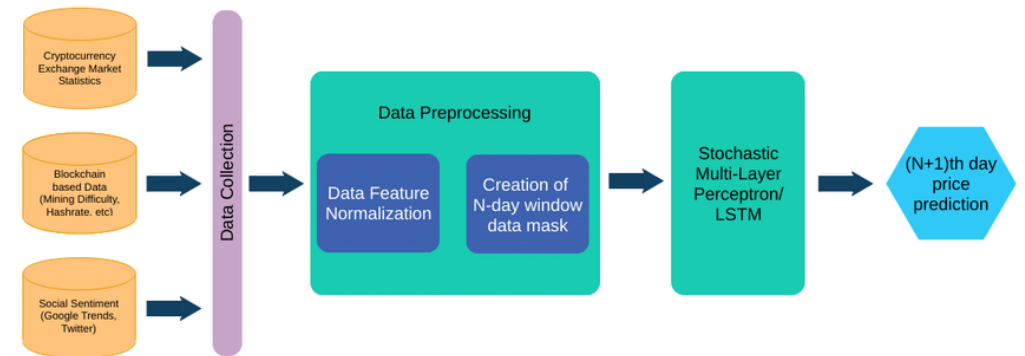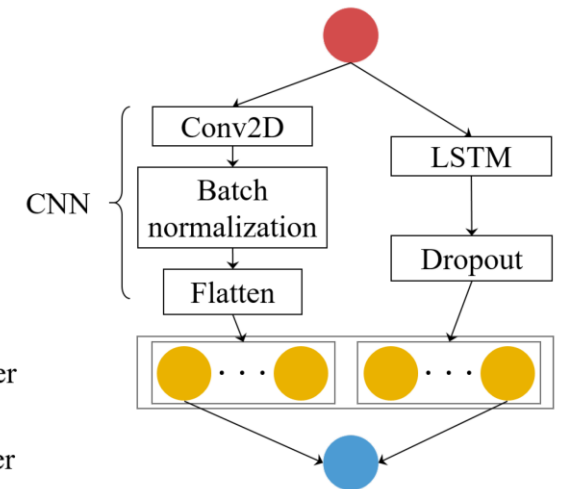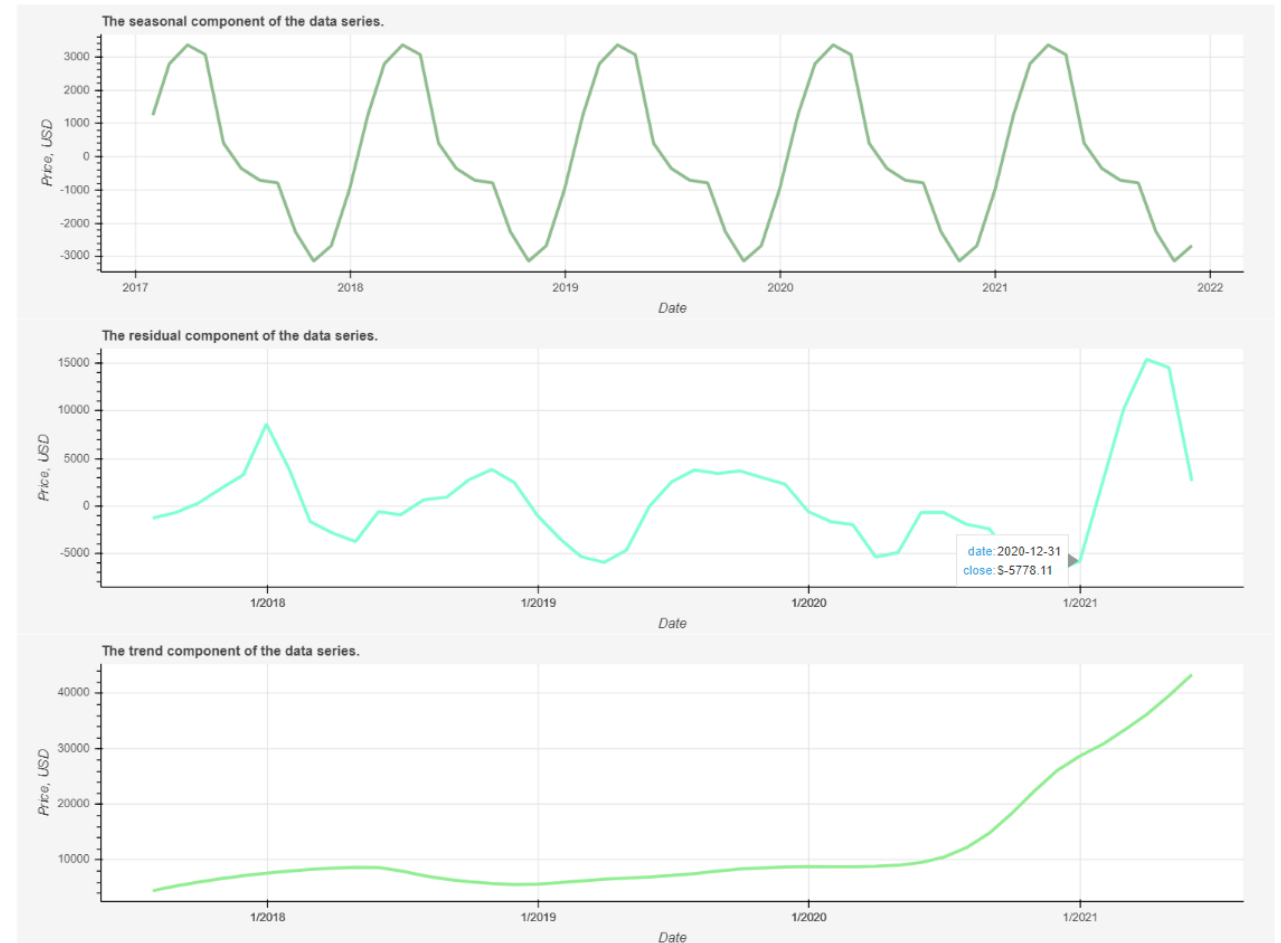
Dataset Modelling

BitPred automatically analyses prices between any time period that the user selects and separates the data trend into 3 components:

- Trend component — changes along with the data in the long term.

- Seasonal — This shows if the data follows patterns which are affected by seasonal factors.

- Residual — In short, everything left after isolating the above two components are recorded here.

The demo dataset was populated using the yfinance library which was then value-checked against the public datasets maintained by crypto exchanges like Poloniex before saving the .csv file.

The seasonal component of the data series.

The residual component of the data series.

date: 2020-12-31
close: $-5778.11

The trend component of the data series.

Results Analysis and Validation

THE INITIAL MODEL PREDICTIONS SEEM TO BE QUITE PROMISING AND ERROR VALUES SEEM TO FALL UNDER EXPECTED PARAMETERS.
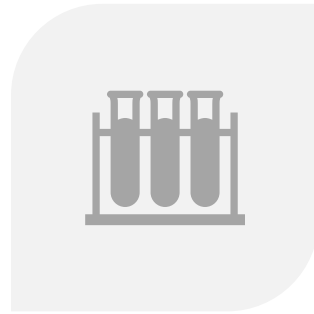
LSTMS WITH DIFFERENT NUMBER OF FEATURES WERE TESTED AND MULTIPLE VALIDATION PARAMETERS WERE RECORDED WITH DIFFERENT TIME PERIODS.
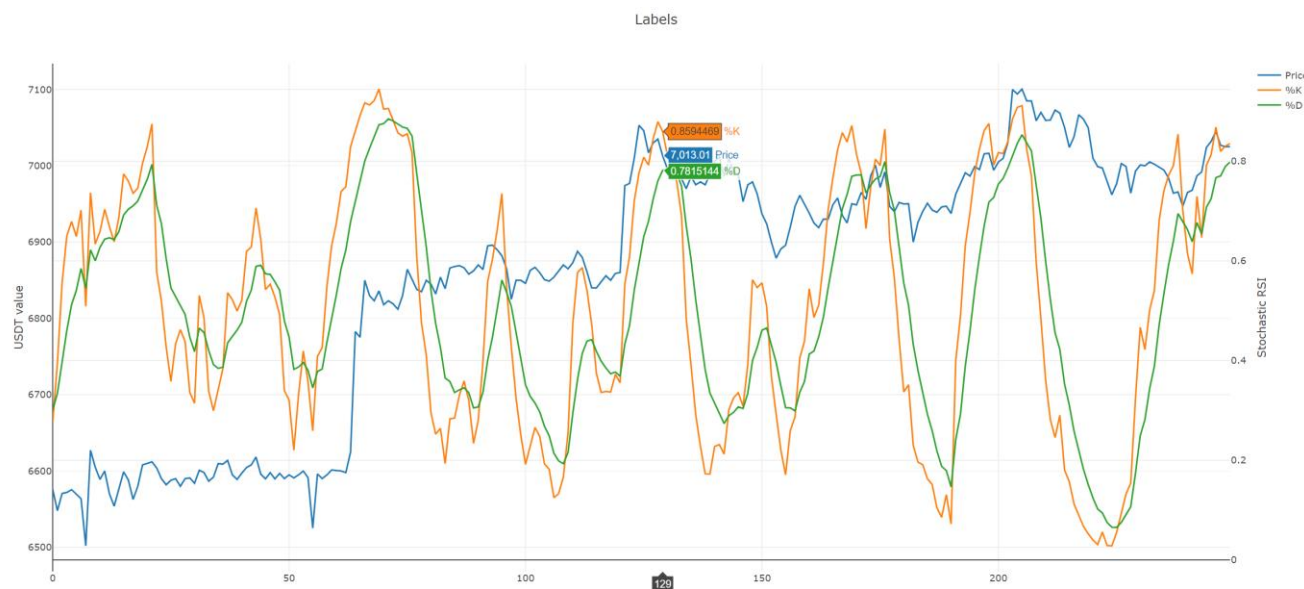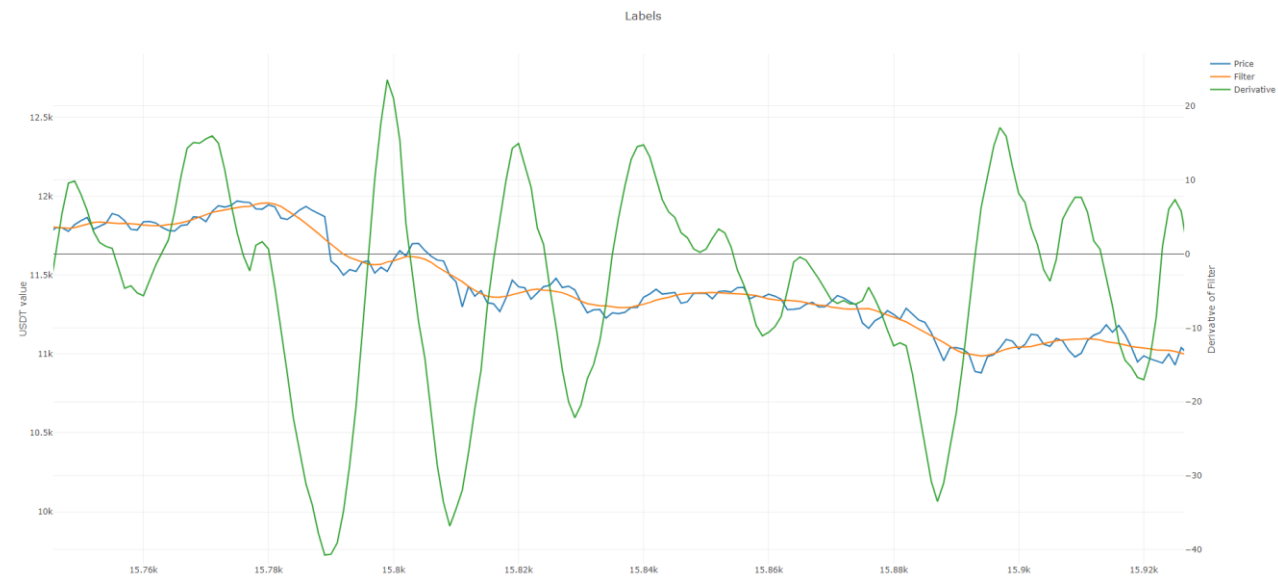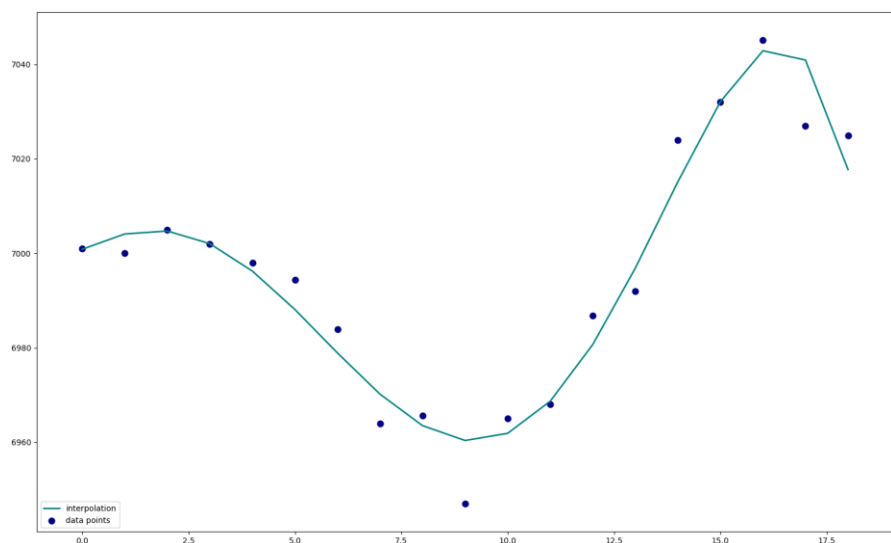
UNDERLYING TRENDS WERE VISUALIZED AND STORED FOR LATER SCRUTINY.

THE CONFIDENCE INTERVALS DRAWN OVER PREDICTED PRICES HELPED QUANTIFY THE PRECISION OF THE RESULTS.

Numerical Analysis Tests

yhat	y_actual
32535.854622	31649.605469
32695.118767	34316.386719
32798.128363	34269.523438
32959.805743	33114.359375
33192.303949	33537.175781

	ds	true	pred
0	2015-08-26	225.830994	708.260620
1	2015-08-27	224.768997	708.061646
2	2015-08-28	231.395996	709.098572
3	2015-08-29	229.779999	709.938599
4	2015-08-30	228.761002	710.415527
...	...	...	...
1983	2021-02-02	35510.289062	35189.097656
1984	2021-02-03	37472.089844	36679.636719
1985	2021-02-04	36926.066406	37522.578125
1986	2021-02-05	37508.750000	38152.109375
1987	2021-02-06	0.000000	38445.300781

1988 rows × 3 columns

| weekly_upper	yearly	yearly_lower	yearly_upper
-23.684358	3955.599988	3955.599988	3955.599988
9.186569	3994.069756	3994.069756	3994.069756
-28.512113	4046.854585	4046.854585	4046.854585
-21.115814	4113.212216	4113.212216	4113.212216
44.645616	4192.025542	4192.025542	4192.025542

|	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_upper	weekly	weekly_lower
1483	2021-01-28	28603.938992	29960.998925	34696.392143	28603.938992	28603.938992	3931.915630	3931.915630	3931.915630	-23.684358	-23.684358
1484	2021-01-29	28691.862442	30383.992891	34966.055908	28691.862442	28691.862442	4003.256325	4003.256325	4003.256325	9.186569	9.186569
1485	2021-01-30	28779.785892	30693.672720	35330.917618	28779.785892	28779.785892	4018.342472	4018.342472	4018.342472	-28.512113	-28.512113
1486	2021-01-31	28867.709341	30572.166154	35356.342750	28867.709341	28867.709341	4092.096402	4092.096402	4092.096402	-21.115814	-21.115814
1487	2021-02-01	28955.632791	30884.251275	35624.817958	28955.632791	28955.632791	4236.671158	4236.671158	4236.671158	44.645616	44.645616

Ethereum Tabular results

Ethereum Predictions and Diagnostics
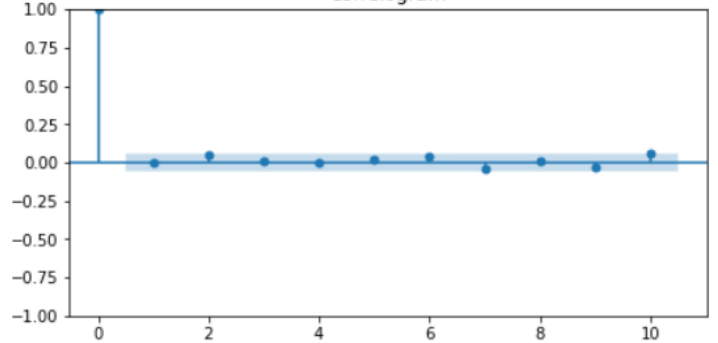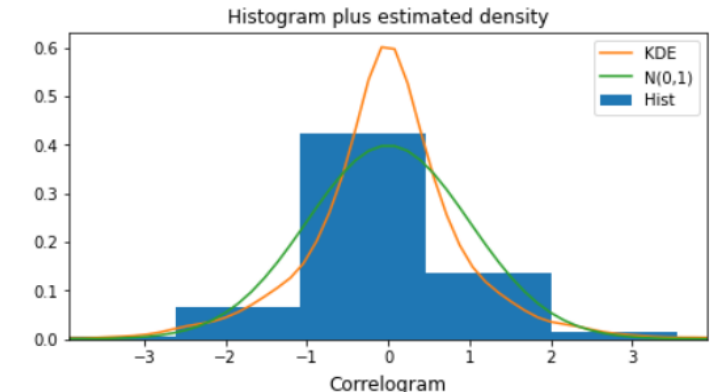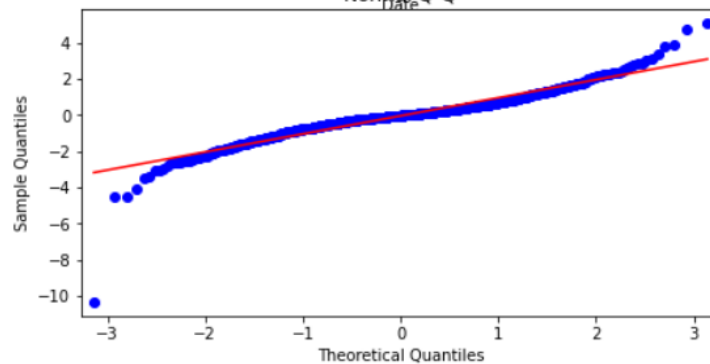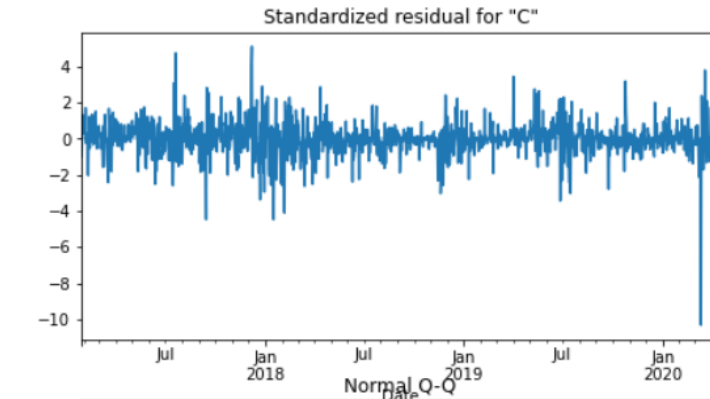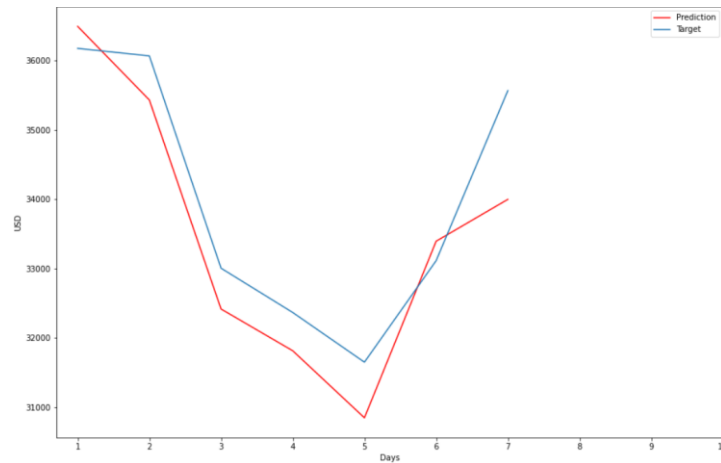
1. Top left: The residual errors seem to fluctuate around a mean of zero and have a uniform variance.\
2. Top Right: The density plot suggest normal distribution with mean zero.\
3. Bottom left: All the dots should fall perfectly in line with the red line. Any significant deviations would imply the distribution is skewed.\
4. ACF plot shows the residual errors are not autocorrelated.\
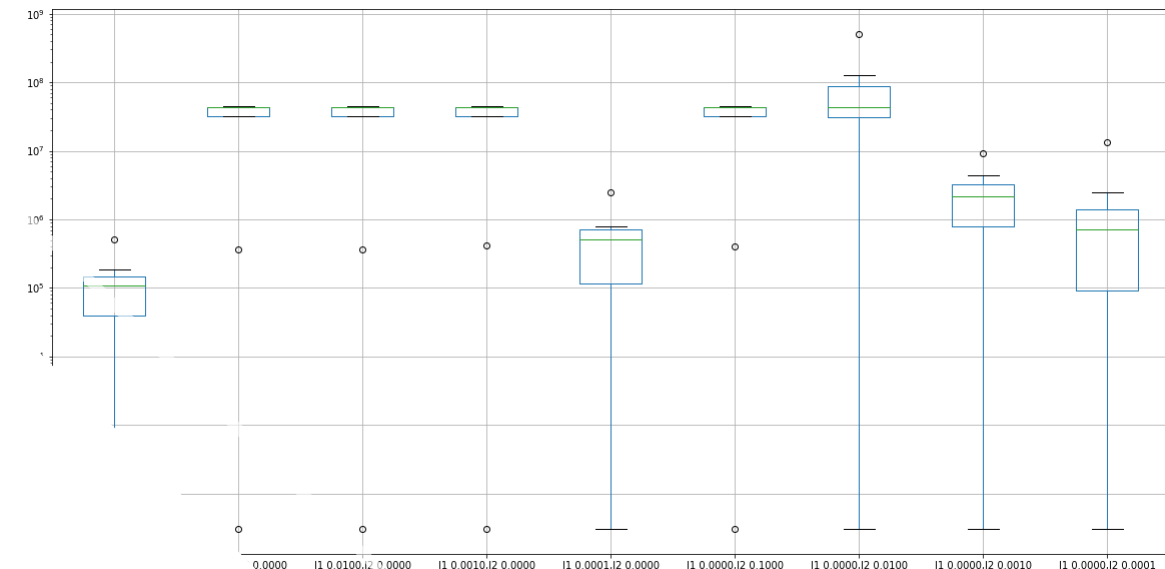
Bitcoin Prediction and Regularization Graphs
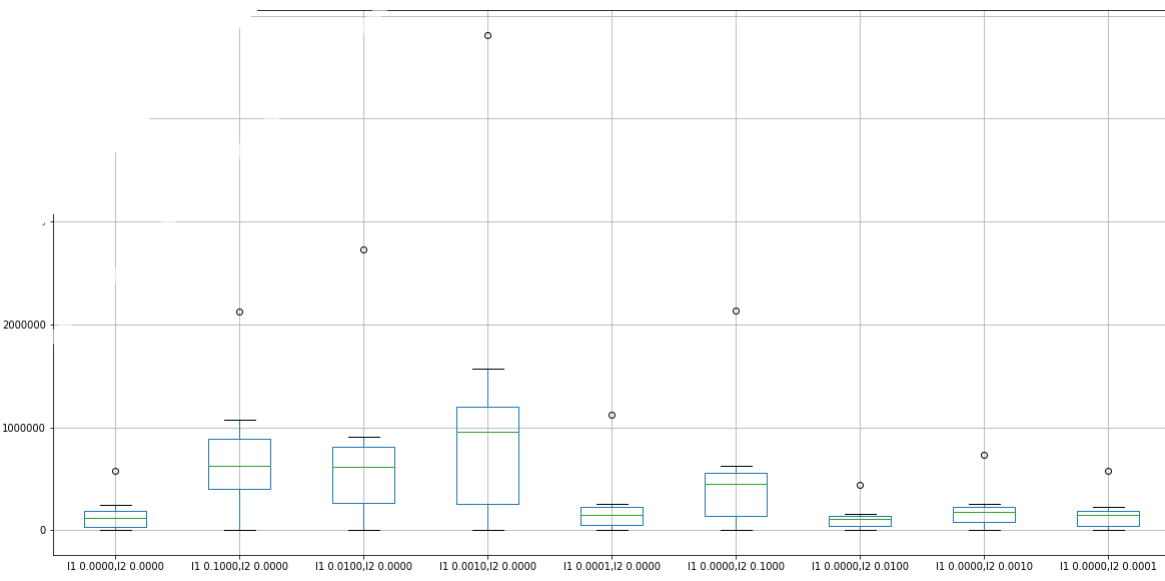
Unscaled RMSE values

Root Mean Squared Error

ARIMA: 508.94965228374423

Prophet: 1786.1781454204304

LSTM: 424.92938066897835

Conclusion and Future Work

The UI part works well but can be improved by adding visual cues for loading and computing tasks.

The SARIMAX model has reliable forecasts but does not do well with limited data.

Prophet offers a fast, automated solution but is less accurate as the hyperparameter tuning options are limited.

My custom model tries to incorporate multiple use cases with a small number of parameters.

The entire project is still constantly learning and improving, and I hope to implement a more robust solution with options for predicting multiple financial assets.
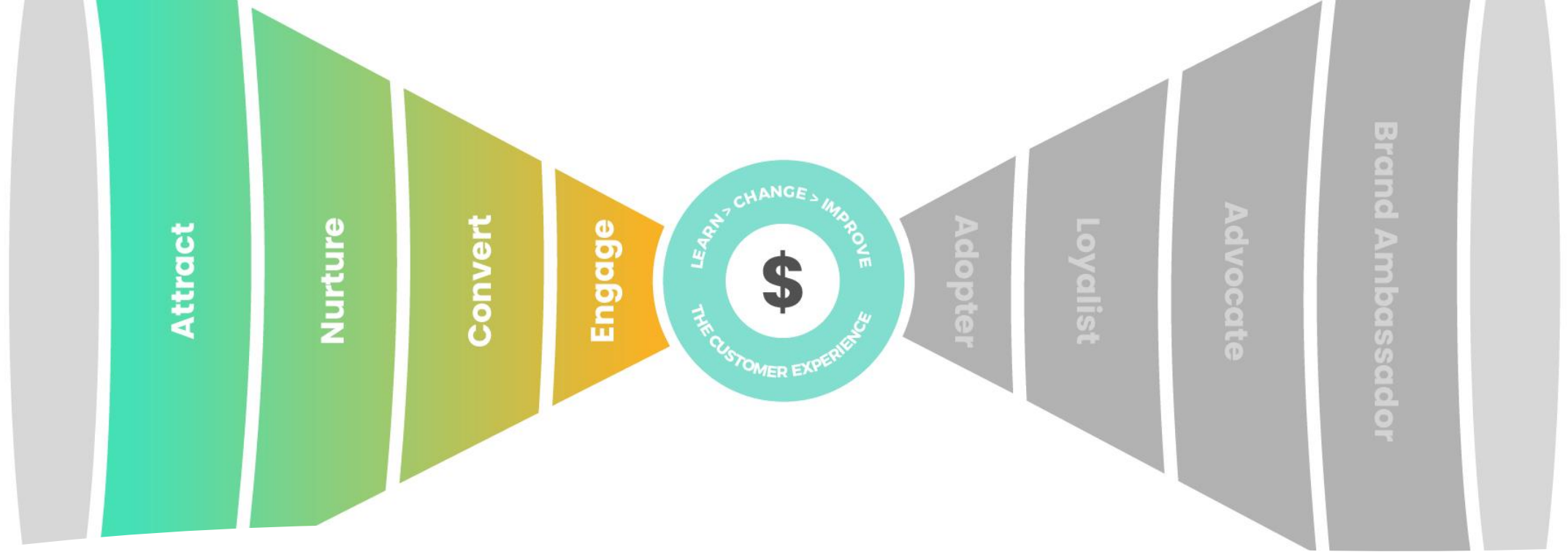
Monetization Opportunity

- The obvious parameter to monetize is the predictions themselves.

- Charging a small fee whenever the predictions made by the app help a user turn a profit.

- Commissions collected in this way will also help customer retention.

- Also, instead of trying to extract exorbitant sums from potential users in anticipation, the revenue model will instead focus on promoting word of mouth advertising.

- Finally, a premium version of the app could be launched.

References

1. J. Rebane, I. Karlsson, P. Papapetrou, and S. Denic, 'Seq2Seq RNNs and ARIMA models for Cryptocurrency Prediction : A Comparative Study', in Proceedings of SIGKDD Workshop on Fintech (SIGKDD Fintech'18), 2018. H. Casanova, et al. Heuristic for scheduling parameters sweep applications in grid environments, in: Proceedings of the 9th Heterogeneous Computing Workshop, HCW, pp. 349–363. 2009.

2. Huang, X., Zhang, W., Huang, Y., Tang, X., Zhang, M., Surbiryala, J., ... & Zhang, J. (2021). LSTM Based Sentiment Analysis for Cryptocurrency Prediction. *arXiv preprint arXiv:2103.14804*. A. Mutz, R. Wolski, Efficient auction-based grid reservation using dynamic programming, in: IEEE/ACM Int'l Conf. on Super Computing, SC 2007.

3. S. Ji, J. Kim, and H. Im, "A Comparative Study of Bitcoin Price Prediction Using Deep Learning," *Mathematics*, vol. 7, no. 10, p. 898, Sep. 2019.

4. M. Rizwan, S. Narejo and M. Javed, "Bitcoin price prediction using Deep Learning Algorithm," 2019 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS), 2019, pp. 1-7, doi: 10.1109/MACS48846.2019.9024772.

5. Patel, M.M., Tanwar, S., Gupta, R. and Kumar, N., 2020. A deep learning-based cryptocurrency price prediction scheme for financial institutions. *Journal of Information Security and Applications*, *55*, p.102583.

6. Predicting Crypto Prices with Deep Learning (https://dashee87.github.io/deep%20learning/python/predicting-cryptocurrency-prices-with-deep-learning/)

7. LSTM regression using TensorFlow (https://mourafiq.com/2016/05/15/predicting-sequences-using-rnn-in-tensorflow.html)

YouTube Video Link and Blog link

- Project Demo Video - https://youtu.be/1mJgOXuY35s

- Blog - https://medium.com/mlearning-ai/my-experience-creating-bitpred-3ab7296e6f12

- GitHub Repository - https://github.com/HyperexD/BitPred

- Reference Repositories:
  - Currency Analyzer (https://github.com/rebeccabernie/CurrencyAnalyser)
  - Kriptomist (https://github.com/flowcoin/kriptomist)
  - BitVision (https://github.com/shobrook/BitVision)
  - An Analysis of the Viability and Volatility of Cryptocurrency (https://github.com/rebeccabernie/CurrencyAnalyser/blob/master/Dissertation.pdf)

Questions Please

Thank You