A REPORT ON

SOLIGENCE PREDICTION SYSTEM

BY

HENRY BUKOLA ADEOYE
STUDENT NUMBER – Q16433378

MODULE TITLE: APPLIED AI IN BUSINESS (COM 724)

TUTORS : PENGFIE XUE, BACHA REHMAN, DRISHTY SOBNATH

COURSE NAME: ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

SOLENT UNIVERSITY, SOUTHHAMPTON, ENGLAND, UNITED KINDGOM.

AUGUST, 2023.

Table of Contents

Table of Figures

Introduction

Solent Intelligence (SOLiGence) is a prominent multinational organization in finance, excelling in stocks, shares, savings, and investments. With a vast online platform serving millions, it's poised to transform trading strategies, particularly in the crypto coin realm.

The Intelligent Coin Trading (IST) platform, initiated by SOLiGence, employs predictive analytics and machine learning to enhance cryptocurrency trading. By accurately predicting crypto prices across different timeframes, the IST platform aims to enable strategic trading for potential profits. Rigorous validation using real-world data reinforces its reliability, making it a trustworthy tool for making informed investment decisions.

Machine learning is a computational technique enabling computers to enhance task performance through data-driven learning. By analysing historical data patterns, machine learning algorithms make predictions and decisions without explicit programming. Applied to cryptocurrency, it deciphers intricate market-price relationships. " **In just the last five or 10 years, machine learning has become a critical way, arguably the most important way, most parts of AI are done," said MIT Sloan professor** Thomas W. Malone, the founding director of the MIT Center for Collective Intelligence.

Cryptocurrency markets are intricate, influenced by a myriad of factors including market sentiment, macroeconomic indicators, regulatory changes, technological developments, and even social media buzz. Traditional models struggle to grasp this complexity, driving the adoption of machine learning for insight and navigation.

Consequently, the objective of this report is to enhance predictive capabilities through the utilization of deep learning algorithms. These algorithms aim to uncover latent patterns within the data, amalgamate them, and yield significantly more precise predictions, Presenting an exploratory data analysis for BTC, ETH, and LTC cryptocurrencies. I've utilized a combination of LSTM (Long Short-Term Memory), Random Forest Model, and ARIMA (AutoRegressive Integrated Moving Average) models for predicting cryptocurrency price movements. These models have been strategically selected to leverage their respective strengths in handling time-series data and capturing complex patterns within the volatile cryptocurrency market. The integration of these models aims to enhance the accuracy and robustness of predictions, allowing for a comprehensive evaluation of potential trading opportunities using evaluation metrics such as MSE, MAE, R2 score, and MAPE.

Literature Review

Machine learning, a subset of artificial intelligence, has emerged as a powerful tool to analyse, understand, and predict cryptocurrency price movements and market trends.

In predictive analysis, recent advancements in machine learning have spurred interest in deep learning for Cryptocurrencies' price prediction. Qiu et al. use wavelet analysis to forecast the trajectory of Bitcoin prices over a quarter, employing Bitcoin's historical price time series. Researchers like Huang et al. explore technical indicators and big data to improve Bitcoin return prediction accuracy. These efforts showcase the potential of combining data and analysis for more precise predictions.

Previous research highlights the advantages of machine learning over traditional forecasting models. These benefits include producing results closely aligned with actual outcomes and improving precision. Hitam et al.'s study in 2022 utilized Support Vector Machines (SVMs) for cryptocurrency prediction, showcasing promising results for BTC and other currencies like ETH, LTC, XRP, and Stellar. Their comparative analysis of SVM, ANN, and DL techniques revealed SVM's remarkable accuracy in prediction.

Modern research incorporates deep learning methods for cryptocurrency price forecasting. Ji et al. explored advanced deep neural networks, including LSTM, DNNs, and hybrids, for predicting Bitcoin prices. Their study showed LSTM's slight accuracy advantage in regression, while DNNs excelled in predicting price movements.

Hence, the core focus of this report centres on predicting Cryptocurrencies price using LSTM, ARIMA and Random Forest prediction model which my success contribution is focused on achieving utmost accuracy. These models are designed to anticipate forthcoming Cryptocurrency prices by harnessing historical cryptocurrency price data. The models have been enhanced and meticulously trained to exhibit exceptional accuracy and precision, resulting in reduced MAPE, MSE, and MAE values for the purpose of predicting future prices.

Problem Definition

Unavailability of a system by SOLigence that can provide users with reliable predictions, enable them to make informed investment decisions, visualize key performance indexes and trends for various cryptocurrencies.

Aims and Objectives

Aim: The primary aim of this project is to develop an Intelligent Coin Trading (IST) platform that leverages machine learning techniques to predict cryptocurrency price movements and provide users with actionable insights for intelligent trading decisions.

Objectives:

1. **Model Integration and Development:**
2. **Graphical User Interface (GUI) Creation:**
3. **Real-Time Data Handling:**
4. **Scenario-Based Decision Support:**
5. **Machine Learning Model Evaluation:**
6. **Validation and Confidence Assessment:**
7. **Profit and Loss Analysis:**
8. **Market Trends and Correlations:**
9. **What-If Analysis:**
    - Allow users to explore “What-If” scenarios by altering parameters such as purchase price, quantity, and time intervals.
    - Enable users to gauge potential outcomes under different circumstances.

By achieving these objectives, the project aims to produce a robust and user-friendly Intelligent Coin Trading platform that empowers users to make informed decisions in the dynamic cryptocurrency market.

The Success Metrics for the evaluation of each Model averagely on the 3 coins (BTC, ETH and LTC) are;

LSTM achieved an average accuracy of 78% when predicting on new datasets for BTC, ETH, and LTC. In comparison, Random Forest exhibited an average accuracy of 77%, while ARIMA demonstrated an average accuracy of 76% for the same cryptocurrencies

Data Collection

The historical cryptocurrency price data was retrieved the specified cryptocurrencies using the Yahoo Finance API, Coin Gecko and Coin compare then merged and stored in individual CSV files. This data serves as the foundation for further analysis and the development of the Intelligent Coin Trading (IST) platform. The start date as '2017-01-01' and the end date as the current date ('%Y-%m-%d' format). List of ticker symbols corresponding to various cryptocurrencies of interest were defined.

1. Date (Timestamp):
   - This is a critical dimension representing the chronological order of the data. Timestamps indicate when each data point was recorded. The time-series nature of this data makes it suitable for time-dependent analysis and modeling.
2. **High Price:**
   - This column represents the highest price of the cryptocurrency during a specific time interval (e.g., a day or a week). High prices indicate the maximum value the cryptocurrency reached within that period.
3. **Low Price:**
   - The low price column represents the lowest price of the cryptocurrency within the same time interval. Low prices indicate the minimum value the cryptocurrency reached during that period.
4. **Close Price:**
   - The close price column denotes the price of the cryptocurrency at the end of the time interval. It is a crucial value as it indicates the final trading price for that interval.
5. **Volume:**
   - The volume column refers to the total amount of the cryptocurrency traded within the given time interval. Volume can provide insights into the market's activity and liquidity.

Data Dimensions: The dataset's dimensions can be characterized as follows:

- **Temporal Dimension:** The data is organized chronologically based on timestamps. This dimension enables time-series analysis, making it suitable for forecasting future price trends.

- **Feature Dimensions:** The dataset contains four primary numerical features: high, low, close prices and volume. Each of these dimensions contributes to understanding different aspects of cryptocurrency performance and market behaviour.

- **Target Dimension:** since the aim is to predict future cryptocurrency prices, the target dimension is the price itself (i.e. close) for a specific future time interval.

Significance: Understanding the dimensions of the data is essential for pre-processing, model selection, and feature engineering. The temporal aspect guides time-series analysis, while the various features provide insights into price movements, volatility, trading activity, and the impact of corporate actions. This knowledge aids in building the accurate and effective machine learning models for the cryptocurrency price prediction.

Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a crucial initial step in understanding and gaining insights from the dataset. It involves visualizing and summarizing data to uncover patterns, anomalies, relationships, and potential outliers. In the context of cryptocurrency dataset retrieved from coin gecko, coin compare and Yahoo Finance, EDA serves as the foundation for ther subsequent modeling and prediction tasks.

Key Aspects of EDA for Cryptocurrency Data:

1. Data Overview: Begin by loading the data into a DataFrame and taking an initial look at its structure after merging from different coinbase. Display the first few rows to understand the columns and data format.

	Date	Open	High	Low	Close
0	2017-01-01	963.658020	1003.080017	958.698975	998.325012
1	2017-01-02	998.617004	1031.390015	996.702026	1021.750000
2	2017-01-03	1021.599976	1044.079956	1021.599976	1043.839966
3	2017-01-04	1044.400024	1159.420044	1044.400024	1154.729980
4	2017-01-05	1156.729980	1191.099976	910.416992	1013.380005

	Adj Close	Volume
0	998.325012	147775008
1	1021.750000	222184992
2	1043.839966	185168000

Figure 1. Showing structure of Data after merging.

2. **Summary Statistics:** Compute basic summary statistics for each column, including mean, median, standard deviation, minimum, and maximum values. This provides an overview of the data's central tendencies and variability.

	Open	High	Low	Close	Adj Close	Volume
count	2403.000000	2403.000000	2403.000000	2403.000000	2403.000000	2.403000e+03
mean	18359.758263	18805.842617	17871.806557	18369.711879	18369.711879	2.228881e+10
std	16208.134012	16613.181621	15740.611478	16202.032499	16202.032499	1.956553e+10
min	775.177979	823.307007	755.755981	777.757019	777.757019	6.085170e+07
25%	6602.525146	6746.965088	6494.954834	6604.863281	6604.863281	5.994930e+09
50%	10519.278320	10793.507812	10229.628906	10530.732422	10530.732422	1.936904e+10
75%	28164.223633	28576.186523	27573.795898	28176.900391	28176.900391	3.241112e+10
max	67549.734375	68789.625000	66382.062500	67566.828125	67566.828125	3.509679e+11

Figure 2. Showing Summary Statistics for each column.

3. **Time-Series Visualization:** Since the data includes timestamped information, visualizing the cryptocurrency price trends over time Is needed using line chart as shown in the Figure below.

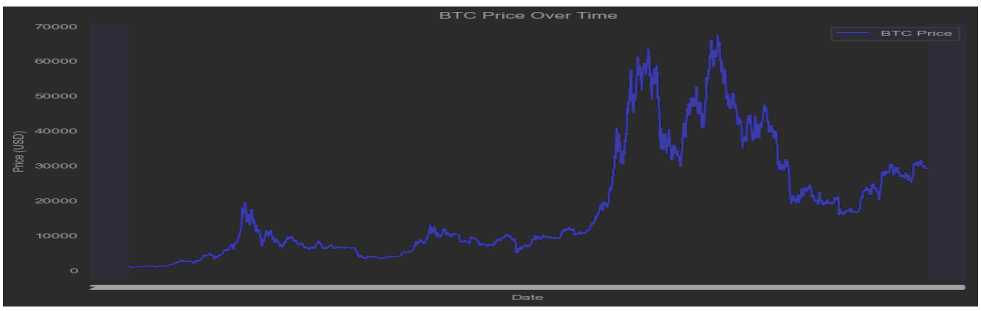

Figure 3. Shows Bitcoin Price over Time.

4. **Correlation Analysis: c**orrelations between different cryptocurrency pairs were done to Calculate correlation coefficients and visualize them using heatmaps. This helps identify relationships between different cryptocurrencies' prices movements, an example of such is shown below.

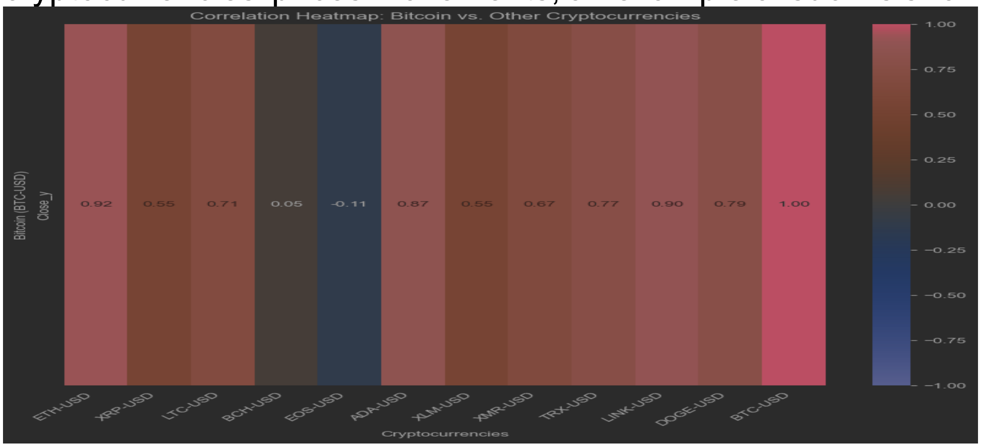

Figure 4. Shows the correlation of Bitcoin with other coins

5. **Volume Analysis:** Analyzed trading volumes over time to Plot volume trends to identify periods of high trading activity and potential market shifts.

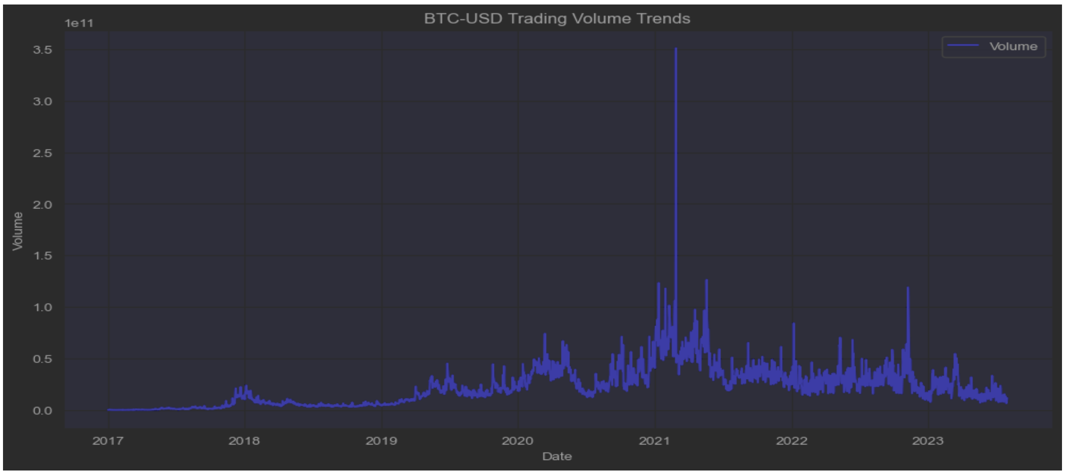

Figure 5. Shows Bitcoin Trading Volume Trends

6. **Volatility Assessment:** the daily price fluctuations (high - low) to assess cryptocurrency volatility was calculated and added to the columns so as to visualize volatility trends and compare them across different cryptocurrencies.

7. **Moving Averages:** Computed and visualized moving averages (e.g., Figure 7) to identify trends and potential support/resistance levels.

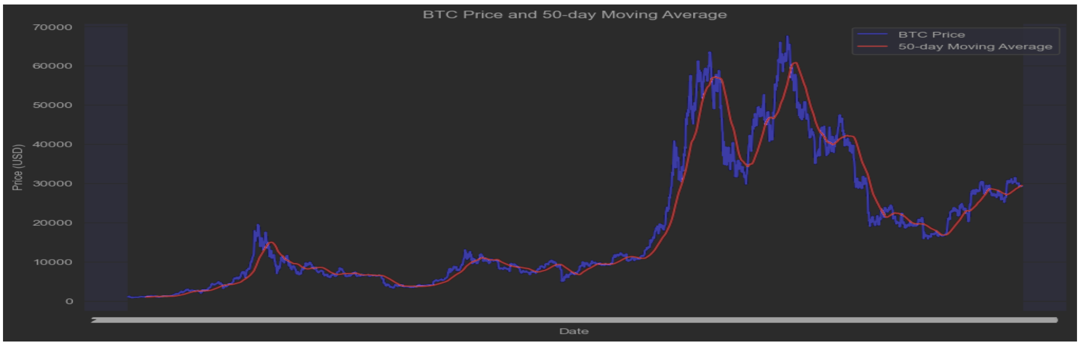

Figure 6. Shows 50days Moving Average of Bitcoin-USD

8. **Outlier Detection:** potential outliers or extreme values in the data that might affect model performance was identified using Box plots.

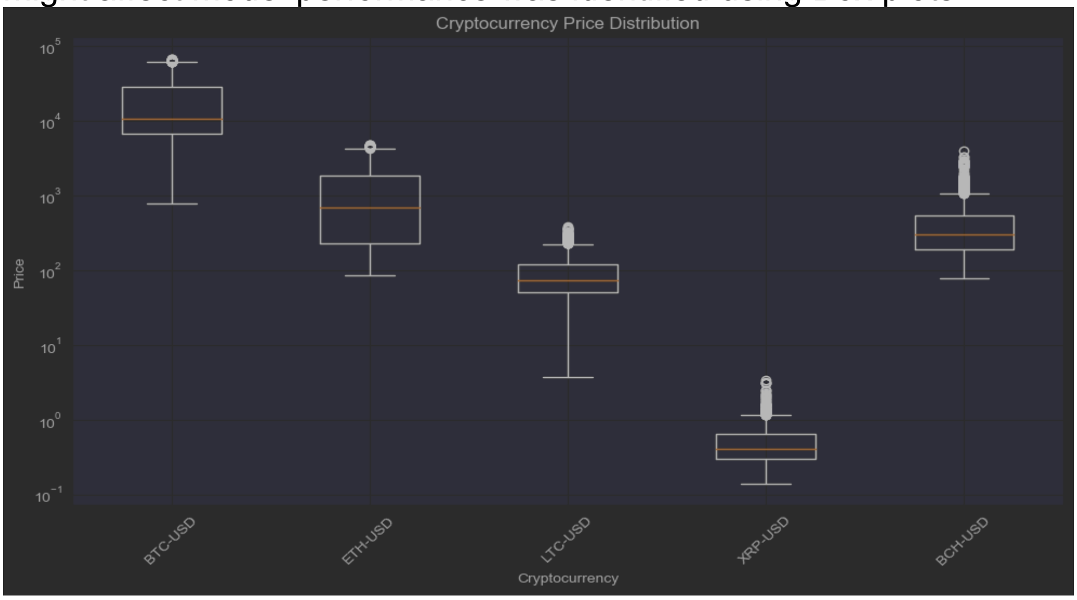

Figure 7. Shows the Boxplot for the Crytocurrecnies (BTC,LTC,BCH,ETH,XRP)

9. **Distribution Analysis:** the distribution of prices and other variables using histograms and density plots was examined. This provides insights into the data's overall shape and potential skewness as shown in Figure 8 below.

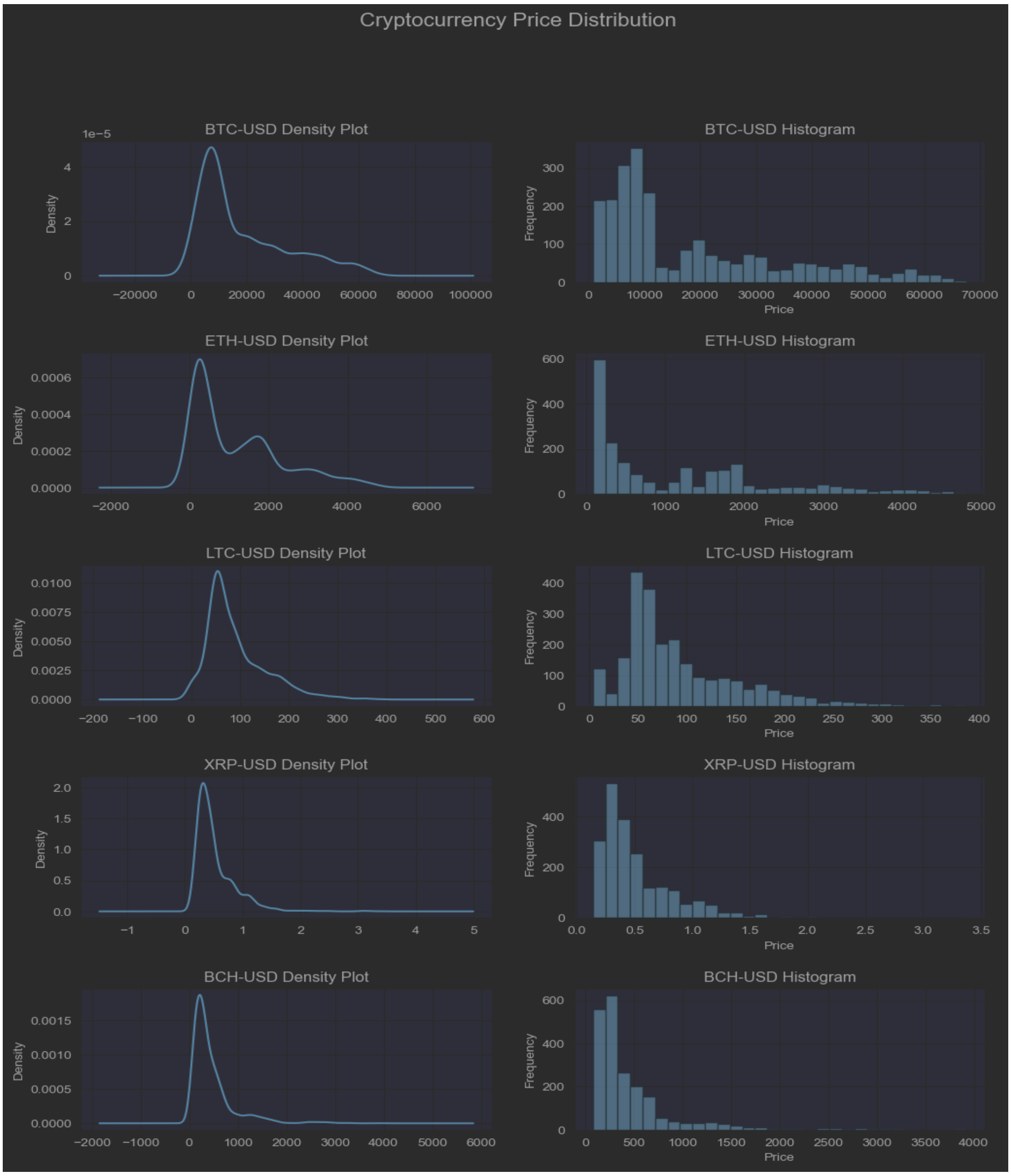

Figure 8. Shows Different Cryptocurrency Distributions Price over Time.

10. Feature Engineering Ideas: EDA inspired potential features for my predictive models as shown in the figure below.

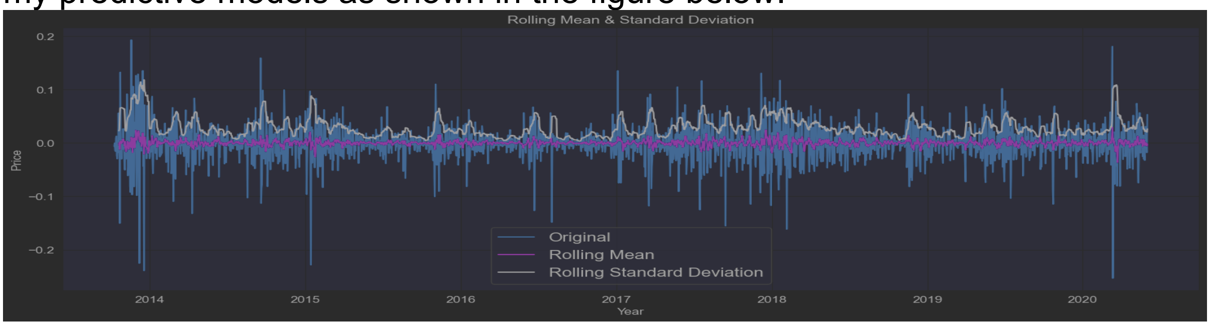

Figure 9. Shows the Rolling Mean

Data Pre-processing

It's important to note that the dataset is devoid of any missing values. This can be attributed to the fact that the dataset was directly acquired from reliable sources, namely Yahoo Finance, Crypto compare, and Coin Gecko. As a result, the integrity of the dataset remains intact, ensuring that all required data points are complete and available for analysis.

Let's consider the Bitcoin dataset as an illustrative case, which was also extended to other cryptocurrencies. The original dataset covered a span of 4980 days of Bitcoin data, ranging from January 1, 2017, to January 1, 2023, collected from crytpo compare, Coin Gecko and yahoo finance sources. Each data entry in the dataset is a six-dimensional vector containing daily values of various features extracted from the Bitcoin blockchain.

For the purpose of predicting Bitcoin prices, the analysis employed sequences of past data. Given a sequence size of 'm', a total of 9980 - m + 1 sequences were utilized for training and testing predictive models. These sequences spanned from S[1:m] to S[9980-m+1:9980], wherein the initial 80% of the sequence data were designated as the training set and the remaining as the test set.

In the context of regression, the goal was to predict the Bitcoin price for the (i+m)-th day based on the sequence S[i:i+m-1]. For classification, the task was to predict whether the price on the (i+m)-th day would rise or fall compared to the (i+m-1)-th day. Essentially, leveraging historical Bitcoin price data, including the current day's price, enabled the prediction of the subsequent day's price in the case of regression. For classification, it facilitated forecasting whether the next day's price would experience an increase or decrease relative to the present day's price.

To facilitate the development and assessment of prediction models, normalized sequences denoted as Ys were utilized instead of the original sequences Xs. This normalization technique proved more effective in capturing local trends compared to the typical min-max normalization. The conventional "min-max normalization" technique involves translating each data point according to the formula:

$S'[i][j]=S[i][j]-minjmaxj-minj$

This approach scales each value to fall within the 0 to 1 range. However, since our dataset includes a significant range between the highest and lowest prices (68033.86 times), adopting the usual min-max normalization resulted in minimal variation in Bitcoin prices across most sequence data. Consequently, this diminished the accuracy of predictions.

Experimental outcomes confirmed that the alternative value-based normalization approach was more effective than the traditional min-max normalization. By implementing this methodology, the prediction models exhibited enhanced performance and were better equipped to capture the intricacies of Bitcoin price trends.

In Summary, Training begins with preparing the dataset. This involves splitting the data into features (input) and target values (output) pairs. Since this is time-series data, it involves creating sequences of past values as inputs and the next value as the target output.

Long Short Term Memory (LSTM) Model Implementation

Using LSTM training process aims to minimize the difference between predicted and actual values, allowing the model to learn patterns and relationships in the data. It's an iterative process that requires finding a balance between fitting the training data well and generalizing to new data.

The architecture of the model was defined which involves selecting the type and number of layers, the activation functions, and other hyperparameters after importing all necessary libraries.. The model architecture includes an LSTM layer followed by a dropout layer and a dense layer with an activation function.

A loss function is defined to measure the difference between the predicted output and the actual target. The optimizer is selected to adjust the model's parameters during training to minimize the loss as shown in the Figure below.

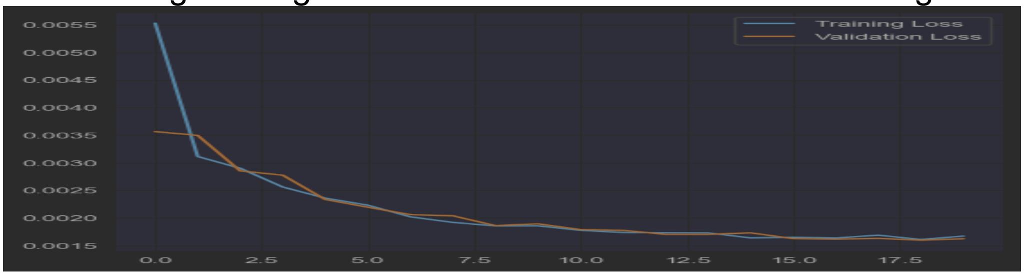

Figure 10. Shows the Validation Loss and the Training Loss

During training, the model takes input data through a forward pass. Each layer performs a transformation on the input data based on its parameters. The output of the model is compared to the actual target using the loss function to compute the loss.

The optimizer adjusts the model's parameters through backpropagation. It computes the gradients of the loss with respect to the model's parameters, indicating how much each parameter should change to minimize the loss. The model's parameters are updated in the direction that reduces the loss.

The training process involves iterating over the dataset multiple times (epochs). In each epoch, the model updates its parameters using different batches of data, gradually improving its performance. The training loss is monitored to ensure it's decreasing over epochs

Finally, The trained model was saved using the model.save method for each cryptocurrency.

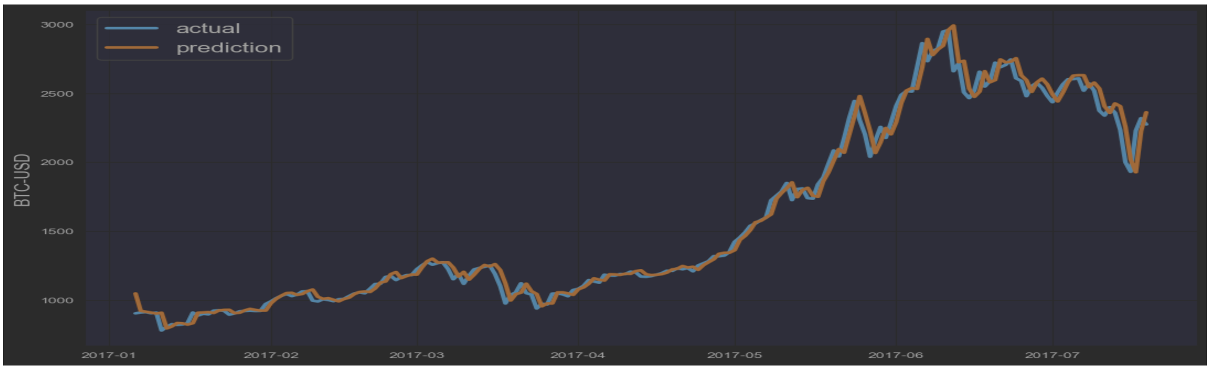

Figure 11. LSTM Actual and Prediction graph

Random Forest Model Implementation

Utilizing Random Forest for cryptocurrency price prediction offers benefits in handling non-linearity and noisy data. Careful feature selection and ensemble strategies can enhance accuracy in short-term forecasts amid market volatility.

Necessary libraries were imported, including NumPy for numerical operations, pandas for data manipulation, matplotlib for plotting, RandomForestRegressor for building the Random Forest model, and other evaluation metrics.

Hyperparameters like the window length (number of time steps used for prediction), test size (proportion of data for testing), prepare_data function and data (model data and aim) are defined, these function are used to prepare the training and testing data. The X_train and X_test sequences are reshaped into 2D arrays to be compatible with the RandomForestRegressor.

An instance of RandomForestRegressor with specified hyperparameters (e.g., 100 estimators and a random seed of 42) was created for the training.

Lastly the trained Random Forest model was saved as a file named 'btc_rf_model.joblib' using the joblib.dump function. Each coin was trained and saved with their coin names. Here is an example of the true values and predicted values after training of BTC in the figure 12. below.

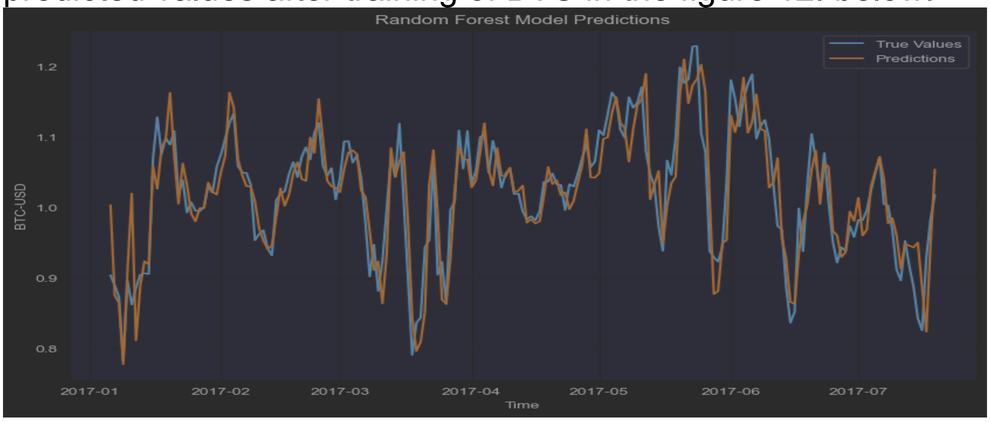

Figure 12. Random Forest Actual and Prediction value graph for BTC

Autoregressive Integrated Moving Average (ARIMA) Model

ARIMA, a versatile time series forecasting method, is well-suited for this task due to its ability to capture temporal dependencies and patterns in data. Here's the steps I followed:

Step 1: Data Preparation and Exploration

- Calculate rolling mean and rolling standard deviation for the Bitcoin price data.
- Perform Augmented Dickey-Fuller test for stationarity.

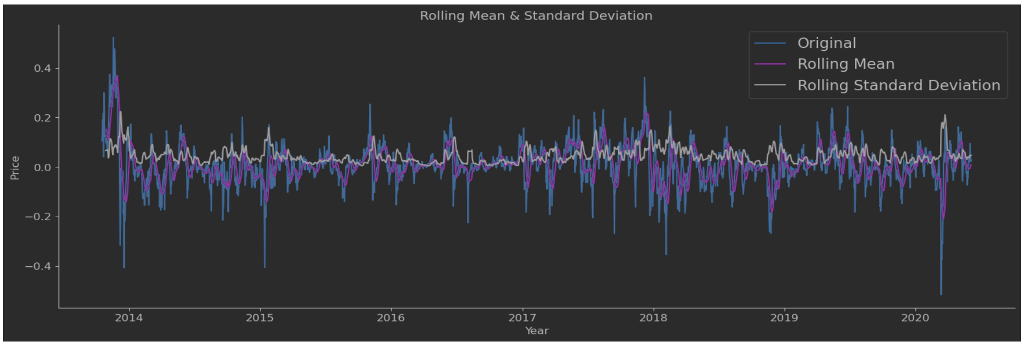

Figure 13. Rolling Mean and SD graph Using ARIMA Model

Step 2: Data Transformation and Stationarity

- Transformed the Bitcoin price data to log scale.
- Subtracted the rolling mean from the log-transformed data.
- Removed NaN values.
- Defined a function to test stationarity with rolling mean and standard deviation plots and Dickey-Fuller test.

Step 3: Exponential Moving Average (EMA)

- Calculated the exponential decay weighted average of the log-transformed Bitcoin data.
- Plot the original log data and EMA.

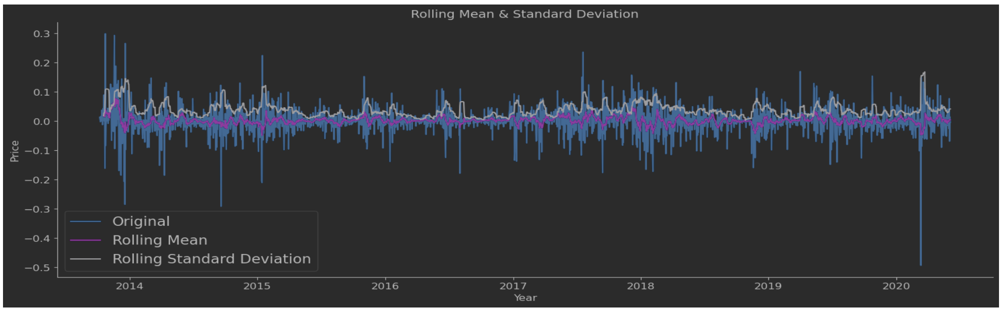

Figure 14. Original Log Data graph Using ARIMA Model

Step 4: Differencing and Stationarity

- Performed first-order differencing on the log-transformed Bitcoin data.
- Plot the differenced data.

Step 5: Seasonal Decomposition

- Used seasonal decomposition to extract trend, seasonal, and residual components from the log-transformed Bitcoin data.

Step 6: ARIMA Model Building

- Imported necessary libraries.
- Fitted an ARIMA model with order (2, 1, 2) to the differenced log-transformed Bitcoin data.
- Calculated the fitted values and convert them back to the original scale.

Step 7: Model Evaluation and Saving

- Plot actual prices and fitted prices from the ARIMA model.
- Calculated the Residual Sum of Squares (RSS) to evaluate the model.
- Saved the trained ARIMA model using pickle.

Step 8: Forecasting and Visualization

- Calculated the fitted values from the ARIMA model.
- Converted fitted values to cumulative sum.
- Converted cumulative sum back to log scale and then to the original price scale using the inverse of the log transformation.
- Plot the original Bitcoin prices and the predicted prices from the ARIMA model

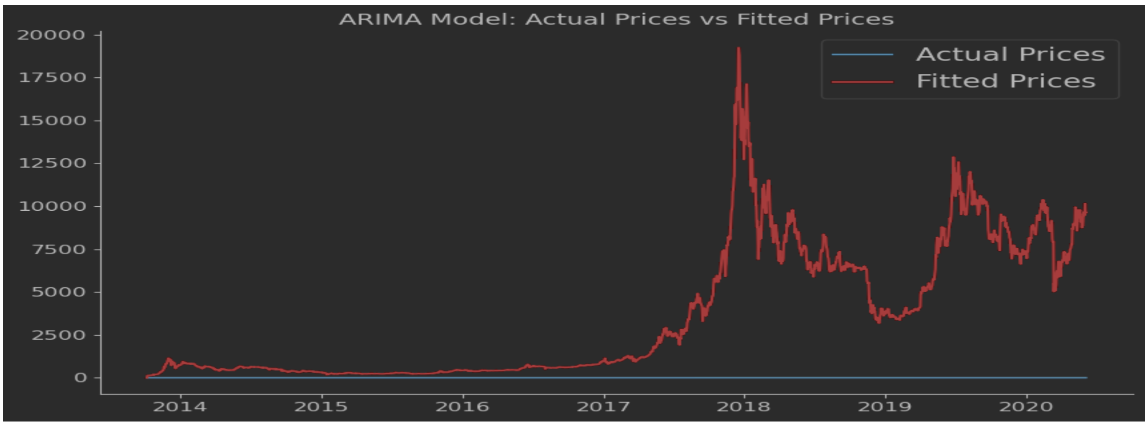

Figure 15. ARIMA Model Actual and Predicted Price

Evaluation

The Models were evaluated using MAE, R2 score, MAPE and MSE

The **mean absolute percentage error** (MAPE) —measures accuracy as a percentage, and can be calculated as the average absolute percent error for each time period minus actual values divided by actual values.

Mean Absolute Error (MAE) is a measure of the average size of the mistakes in a collection of predictions, without taking their direction into account

Mean squared error (MSE) measures the amount of error in statistical models. It assesses the average squared difference between the observed and predicted values

R-squared (R^2) is a statistical measure that represents the proportion of the variance for the dependent variable that's explained by an independent variable in a regression model

The Table below shows the evaluation result of each coin trained with LSTM, Random Forest and ARIMA.

LSTM Model Success Metric Table

	MSE	MAE	MAPE	R2 Score
BTC-USD	0.0016	0.0282	3.01%	80%
LTC-USD	0.0485	0.0046	5.20%	82%
ETH-USD	0.0145	0.0631	4.33%	70%

Random Forest Model Success Metric Table

	MSE	MAE	MAPE	R2 Score
BTC-USD	0.0018	0.0314	3.13%	77%
LTC-USD	0.0163	0.0712	4.89%	66%
ETH-USD	0.0046	0.0499	4.89%	82%

ARIMA Model Success Metric Table

	MSE	MAE	MAPE	R2Score
BTC-USD	0.0112	0.0451	7.10%	76%
LTC-USD	0.1011	0.0991	6.11%	76%
ETH-USD	0.1221	0.1211	7.20%	76%

Graphic User Interface(GUI)

Streamlit

The Streamlit Crypto Predictor Interface is a user-friendly web application that allows users to explore, analyze, and forecast cryptocurrency price trends. The interface provides a range of features, including the ability to view historical price data, perform exploratory data analysis (EDA), utilize different predictive models, and plan investment scenarios. With an intuitive and interactive design, users can gain insights into the cryptocurrency market and make informed investment decisions.

MainStream.py:

- **Purpose:** This file serves as the main entry point for the Streamlit application. It defines the user interface layout and logic, integrating various functionalities from other modules.
- **Responsibilities:**
  - Imports necessary libraries, modules, and functions.
  - Defines the main function that initializes the Streamlit app.
  - Handles user interactions, menu selections, and input parameters.
  - Coordinates data loading, visualization, model predictions, and investment scenario simulations.
  - Manages the flow of the user interface, guiding users through different menus and functionalities.

data_loader.py:

- **Purpose:** This module focuses on data-related operations, including fetching and preprocessing cryptocurrency price data from external sources.
- **Responsibilities:**
  - Imports libraries for data manipulation, visualization, and fetching.
  - Defines functions to fetch historical price data from Yahoo Finance using yfinance API.
  - Implements data pre-processing tasks, such as calculating rolling means and MACD indicators.
  - Generates visualizations for rolling means, moving averages, and MACD analysis using libraries like Seaborn and Matplotlib.

model_loader.py:

- **Purpose:** This module handles loading pre-trained predictive models for cryptocurrency price forecasting.
- **Responsibilities:**
  - Imports machine learning libraries, model objects, and utility functions.
  - Defines functions to load saved LSTM, Random Forest, and ARIMA models for different coins.
  - Provides functions to make predictions using the loaded models.

- Calculates evaluation metrics for model performance, such as MSE, MAE, MAPE, and R-squared.
- Enables users to simulate investment scenarios by predicting future prices and calculating potential profits/losses.

These backend modules, MainStream.py, data_loader.py, and model_loader.py, collectively contribute to the functionality and user experience of the Streamlit Crypto Predictor Interface. They handle data acquisition, pre-processing, predictive modeling, visualization, and investment scenario planning. Through effective organization and integration, these modules empower users to explore and analyse cryptocurrency trends while making informed investment decisions.

Features:

1. **Home Page - Soligence Crypto Predictor:** The interface starts with the Soligence Crypto Predictor as the home page, providing a seamless entry point for users.

Figure 16. showing GUI interface(Home Page)

Main Menus:

- **Cryptocurrencies:** Users can select between Bitcoin (BTC), Ethereum (ETH), and Litecoin (LTC). The interface displays historical price data, top stories, and RSS feeds from Yahoo Finance.

Figure 17. Soligence Interface Showing Cryptocurrencies Menu displaying RSS feeds

- Models: Users can choose from three predictive models: Long Short-Term Memory (LSTM), Random Forest, and AutoRegressive Integrated Moving Average (ARIMA). They can assess each model's performance, prediction and training graph, and suitability for forecasting.

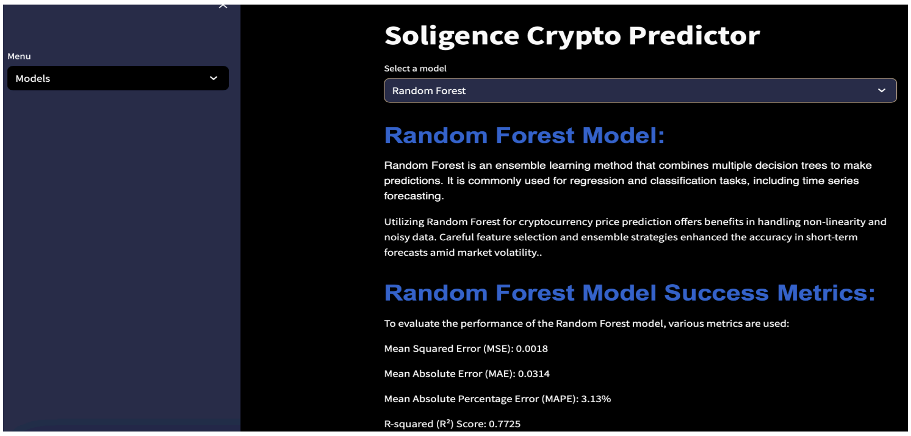

Figure 18 Interface showing Models Menu for Random Forest Model

- Purpose: This section encompasses Exploratory Data Analysis (EDA), t detection, price prediction, and Trends (investment scenario planning).

Purpose Menu:

- Exploratory Data Analysis: shows the raw data for the present day. Users can also adjust the rolling window to visualize rolling means, moving averages, and price distributions to understand price dynamics through graph as shown in the figures below.

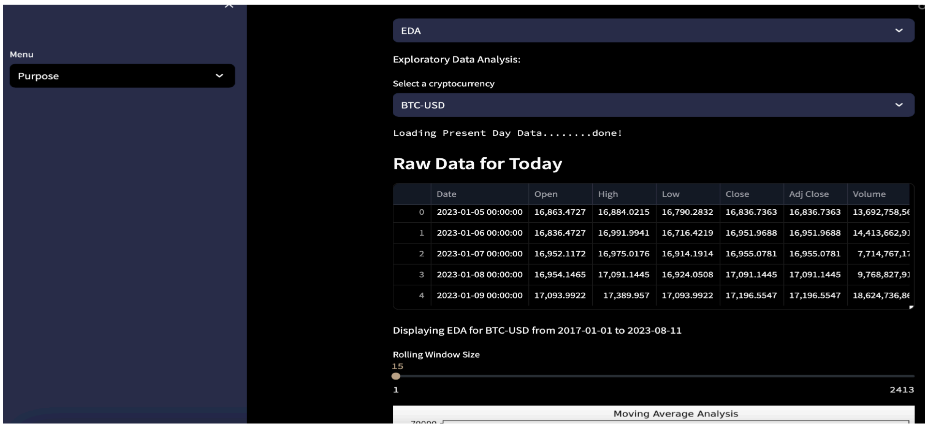

Figure 19. Interface Showing Purpose submenu EDA

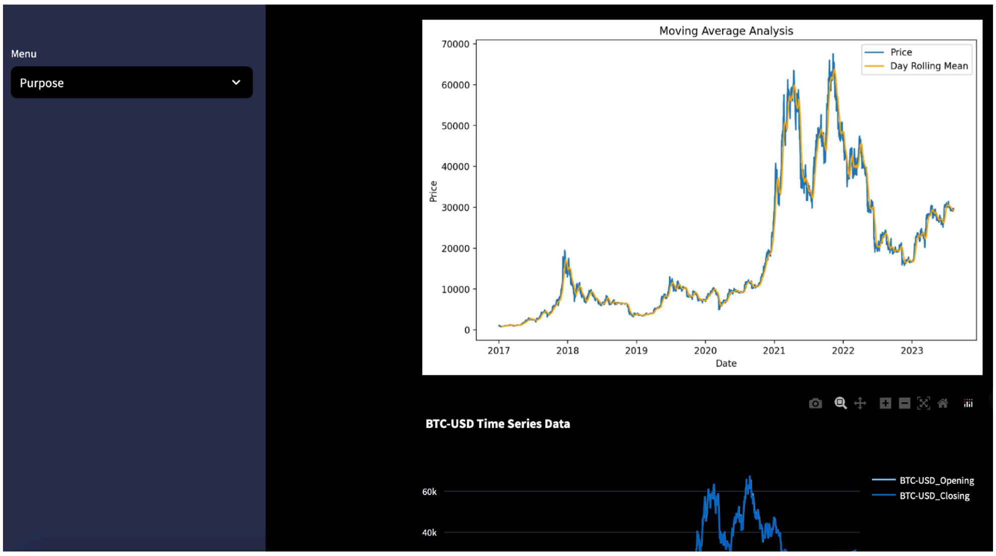

Figure 20. Interface showing Purpose Submenu EDA for Moving Average Analysis

- Detection: Identify bullish and bearish signals for investment decision-making.

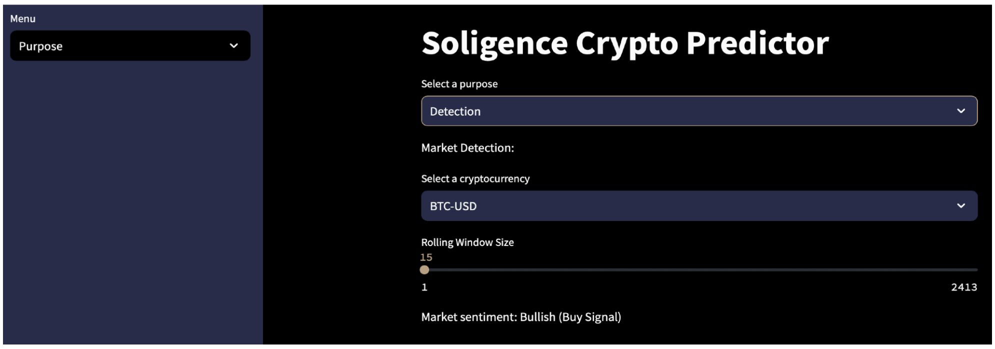

Figure 21.Interface showing Purpose submenu Detection

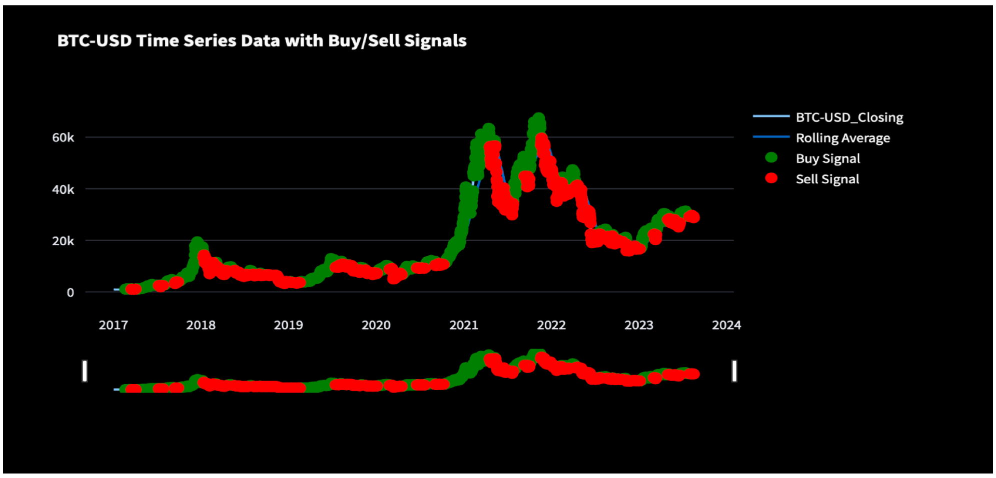

Figure 22. Buy and Sell signal Chart

- Price Prediction: Utilize LSTM, Random Forest, and ARIMA models to forecast future prices. Users can Input sell price and coin quantity to simulate potential profit/loss scenarios. The system will give buy or sell suggestions based on the profit outcome.

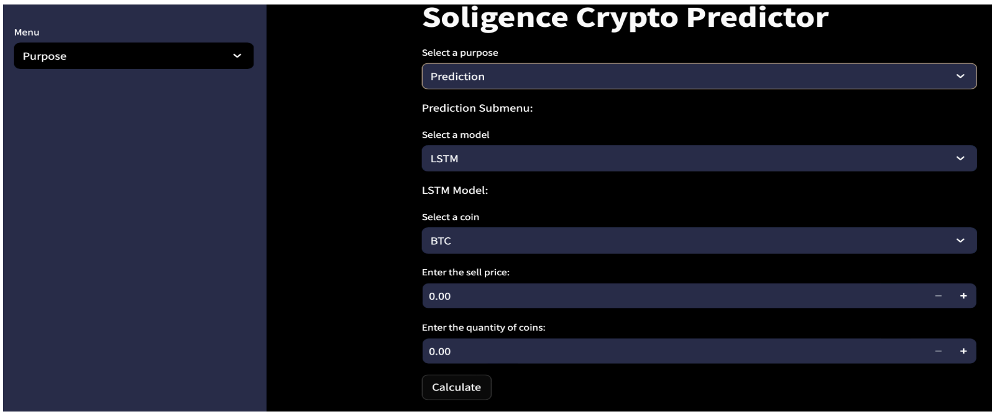

Figure 23. Interface showing Purpose submenu Prediction

- Trends: calculates the MACD indicators for a selected cryptocurrency's price data and visualizes the MACD line, signal line, and histogram which is used for identifying potential trend changes and generating buy or sell signals.

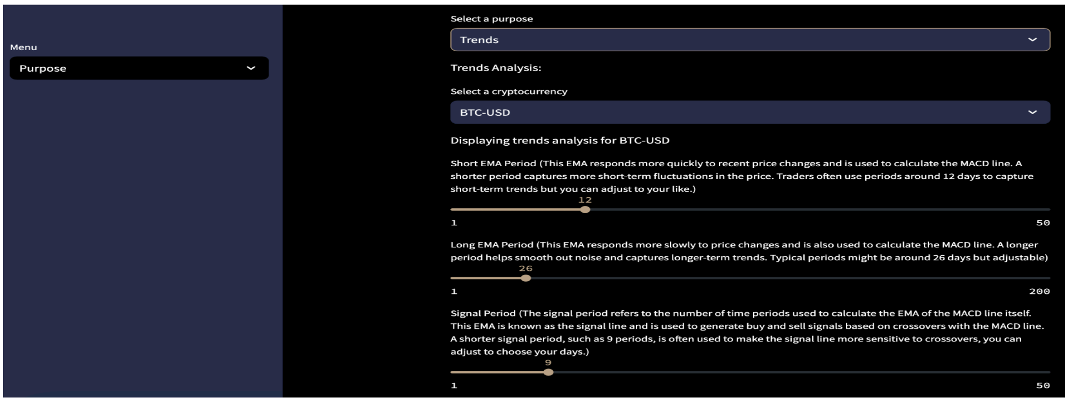

Figure 24. Interface showing Purpose submenu Trends (calculating MACD Indicators)

Conclusion

I n summary, the journey towards developing an accurate cryptocurrency prediction system for SOLigence involved addressing challenges in data preprocessing and merging from various coin bases. Overcoming data noise, inconsistencies, and aligning different datasets demanded meticulous cleansing and harmonization.

During the assessment, LSTM, ARIMA, and RandomForest models were trained and evaluated. Notably, LSTM outperformed the others due to its prowess in capturing intricate temporal patterns, leading to superior accuracy demonstrated by metrics like MSE and MAE.

However, implementing this system faced challenges in terms of creating a user-friendly GUI interface. Ensuring a seamless and intuitive experience demanded proficient integration of the chosen model with a visually appealing and responsive Streamlit interface. Bridging the gap between technical complexity and user-friendliness emerged as a significant hurdle.

In conclusion, navigating data complexities and harnessing LSTM's capabilities positions SOLigence for a reliable cryptocurrency prediction system. Integrating this with a Streamlit GUI interface signifies another milestone, enhancing accessibility and usability for users in a dynamic financial landscape.

References

1. Brown, S. (2021). Machine learning, explained. [online] MIT Sloan. Available at: https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained.
2. Deepchecks. (n.d.). What is Mean Absolute Error. [online] Available at: https://deepchecks.com/glossary/mean-absolute-error/.
3. Fernando, J. (2021). R-Squared Definition. [online] Investopedia. Available at: https://www.investopedia.com/terms/r/r-squared.asp.
4. Glen, S. (2017). Mean absolute percentage error (MAPE). [online] Statistics How To. Available at: https://www.statisticshowto.com/mean-absolute-percentage-error-mape/.
5. Hitam, N.A.; Ismail, A.R.; Samsudin, R.; Alkhammash, E.H. The Effect of Kernel Functions on Cryptocurrency Prediction Using Support Vector Machines. In Proceedings of the International Conference of Reliable Information and Communication Technology; Springer: Cham, Switzerland, 2022; pp. 319–332.
6. Huang, J.-Z., Huang, W. and Ni, J. (2019). Predicting bitcoin returns using high-dimensional technical indicators. The Journal of Finance and Data Science, 5(3), pp.140–155. doi:https://doi.org/10.1016/j.jfds.2018.10.001.
7. Ji, S., Kim, J. and Im, H. (2019). A Comparative Study of Bitcoin Price Prediction Using Deep Learning. Mathematics, 7(10), p.898. doi:https://doi.org/10.3390/math7100898.
8. Qiu L., Fanbin W., Longxiu Y. et al.: 'Prediction of Bitcoin price based on wavelet analysis', Neijiang sci. Technol., 2015, 36, (5), pp. 129 –130.
9. Wisdomtree.eu (2023). Crypto Insight Centre. [online] Available at: https://www.wisdomtree.eu/en-gb/strategies/crypto-currency?