# A close up of a logo Description automatically generated

# SCHOOL OF COMPUTER AND INFORMATION TECHNOLOGY

MINI PROJECT

ON

**“Cryptocurrency Price Prediction Using**

**Machine Learning.**

SUBJECT

Machine Learning

# Bachelor of Technology

In

# Information Science and Engineering

Submitted by

**LINGARAJ**

**R23EQ802**

**1.Problem Description**

Cryptocurrencies like Bitcoin, Ethereum, and others have become increasingly popular in financial markets. Their price volatility is both a challenge and an opportunity. Predicting the price of cryptocurrencies is a complex task due to their highly volatile nature. Factors like market sentiment, geopolitical events, technological advancements, and regulatory news contribute to price fluctuations. The problem is to forecast the future price of a cryptocurrency based on historical price data using machine learning models.

**Challenges:**

* Volatility: Cryptocurrency markets are known for their high volatility, making price predictions difficult.
* Time Series Nature: Cryptocurrencies follow time series patterns, which require specialized models (like LSTM or ARIMA) to capture sequential dependencies.
* Data Noise: Price data can have a lot of noise, especially with daily price fluctuations.
* Feature Engineering: Selecting the right features, such as historical prices, moving averages, and market data, is crucial for model accuracy.

The goal is to build a model that can predict the next day's price of a cryptocurrency using historical data, and evaluate the model’s performance with appropriate metrics**.**

**2.Algorithm Used**

**Several machine learning algorithms are employed to solve the price prediction problem. We will compare both traditional machine learning algorithms and deep learning models to identify the best approach.**

1. **Linear Regression:**
   * A simple model that assumes a linear relationship between the features and the target variable (price).
   * Suitable for establishing baseline performance.
2. **Random Forest Regressor:**
   * A non-linear, ensemble method that works well with complex data.
   * Robust to overfitting and can capture non-linear relationships in the data**.**
3. **Long Short-Term Memory (LSTM) Networks:**
   * A type of Recurrent Neural Network (RNN) that is particularly well-suited for sequential data like time series.
   * It remembers long-term dependencies, which is crucial for predicting time series data like cryptocurrency prices.
4. **Support Vector Regression (SVR):**
   * A machine learning model that can handle high-dimensional feature spaces and is effective in situations with small sample sizes.
5. **XGBoost:**
   * A highly effective and efficient implementation of gradient boosting. It is an ensemble technique that can be used for both regression and classification tasks.
6. **Model Comparison:**

* We compare these algorithms' performance in terms of their prediction accuracy and robustness using multiple evaluation metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-Squared.

1. **Performance Metrics: We will compare the algorithms based on the following:**
   * Mean Absolute Error (MAE): The average of absolute errors between predicted and actual values. It provides a straightforward metric of model accuracy.
   * Mean Squared Error (MSE): Similar to MAE, but penalizes larger errors more heavily by squaring the difference between predicted and actual values.
   * Root Mean Squared Error (RMSE): The square root of MSE. It gives us an idea of the magnitude of prediction error in the same units as the target variable.
   * R-Squared (R²): Measures the proportion of the variance in the target variable explained by the model. An R² value closer to 1 indicates a good fit.

**3.Dataset Used with Dataset Link**

For cryptocurrency price prediction, datasets generally contain historical price information of one or more cryptocurrencies. Common features included are:

* **Open**: The price at which the cryptocurrency opened on that day.
* **High**: The highest price reached during the trading session.
* **Low**: The lowest price reached during the trading session.
* **Close**: The closing price of the cryptocurrency for that day.
* **Volume**: The total number of units traded during that day.
* **Market Cap**: The market capitalization of the cryptocurrency (Price × Circulating Supply).

**Sources for Dataset:**

1. **Yahoo Finance**: The Yahoo Finance API can be used to download cryptocurrency data.
   * Example: BTC-USD for Bitcoin.
   * You can download data directly using the yfinance Python library.
2. **Kaggle**: Datasets available on Kaggle can provide historical data for multiple cryptocurrencies.
   * Example Dataset: Bitcoin Historical Data (Kaggle).
3. **CoinMarketCap**: Provides extensive historical data for a variety of cryptocurrencies. Some data may be available via API or as downloadable CSV files.
4. **CryptoCompare API**: Another data source that provides cryptocurrency historical data via API access.

**Sample Data:**

The dataset for Bitcoin from Yahoo Finance might look like this:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Date | Open | High | Low | Close | Volume | 7\_day\_MA |
| 2015-01-08 | 315.00 | 330.00 | 310.00 | 320.00 | 1000000 | 322.14 |
| 2015-01-09 | 320.00 | 335.00 | 315.00 | 325.00 | 1200000 | 323.57 |
| 2015-01-10 | 325.00 | 340.00 | 320.00 | 330.00 | 1500000 | 325.71 |
| 2015-01-11 | 330.00 | 345.00 | 325.00 | 335.00 | 1800000 | 328.00 |
| 2015-01-12 | 335.00 | 350.00 | 330.00 | 340.00 | 2000000 | 330.71 |
| 2015-01-13 | 340.00 | 355.00 | 335.00 | 345.00 | 2200000 | 333.57 |
| 2015-01-14 | 345.00 | 360.00 | 340.00 | 350.00 | 2500000 | 336.14 |
| 2015-01-15 | 350.00 | 365.00 | 345.00 | 355.00 | 2700000 | 338.57 |

**4.Features**

Feature engineering is crucial to improving the performance of the machine learning model. Features will typically include:

* **Price Data**: Open, High, Low, Close prices.
* **Volume**: Trading volume can give insights into market activity and volatility.
* **Moving Averages**: These are key indicators in technical analysis.
  + **7-day Moving Average (7-day MA)**: The average price over the past 7 days.
  + **30-day Moving Average (30-day MA)**: The average price over the past 30 days.
  + **200-day Moving Average (200-day MA)**: A longer-term moving average used to capture long-term trends.
* **Relative Strength Index (RSI)**: Measures the speed and change of price movements. Used to identify overbought or oversold conditions.
* **Volatility Index**: Measures the price fluctuations over time.
* **Momentum Indicators**: Rate of change of the price to understand the market momentum.
* **Lagged Variables**: Previous day or week’s closing price can be important predictors for price forecasting.

Example Feature Set:

|  |
| --- |
| ***[Open, High, Low, Close, Volume, 7-day MA, 30-day MA, RSI, Volatility Index]*** |

**Target Variable:**

* **Closing Price**: The target variable is usually the cryptocurrency's closing price on the next day or the next n-days' closing price for multi-step prediction.

**5.Project Implementation**

For cryptocurrency price prediction using machine Jeaming, the following topics are crucial in the project workflow.

**1. Collecting Datasets:**

* Collect historical data on cryptocurrency prices, trading volumes, and other relevant indicators from sources like CoinMarketCap, CoinGecko, etc.
* Ensure the dataset includes a sufficient timeframe to capture various market conditions and trends.

**2. Data Cleaning:**

* Remove errors, missing values, and outliers from the collected data to ensure data quality.
* Normalize and scale the data to make it suitable for machine learning algorithms.
* Handle any inconsistencies or discrepancies in the dataset that could affect model performance.

**3.Data Preprocessing:**

* Transform the data into a format that can be effectively used by machine learning models.
* Perform feature scaling, encoding categorical variables, and handling any data transformations required for the specific algorithms being used.
* Split the data into training and testing sets for model evaluation.

**4.Model Selection:**

* Choose appropriate machine learning models for cryptocurrency price prediction, such as Linear Regression, Support Vector Machines (SVM), Random Forest, or Neural Networks.
* Consider the complexity of the problem, dataset size, and performance metrics when selecting the model.

**5. Model Building:**

* Develop the machine learning model using the selected algorithm and the preprocessed data.
* Define the architecture of the model, including the number of layers, nodes, activation functions, and other relevant parameters.
* Train the model on the training data to learn patterns and relationships in the cryptocurrency price data.

**6.Model Training:**

* Train the machine learning model on the training dataset to optimize its parameters and improve its predictive performance.
* Use techniques like backpropagation for neural networks or optimization algorithms for other models to adjust the model weights.
* Monitor the model's performance during training to prevent overfitting or underfitting.

**7.Final Prediction Using Built Model:**

* Once the model is trained and validated on the testing dataset, use it to make predictions on new, unseen cryptocurrency price data.
* Evaluate the model's performance using metrics like mean absolute error (MAE), mean squared error (MSE), or root mean squared error (RMSE).
* Deploy the trained model to generate real-time predictions for cryptocurrency prices, enabling users to make informed decisions based on the model's forecasts.

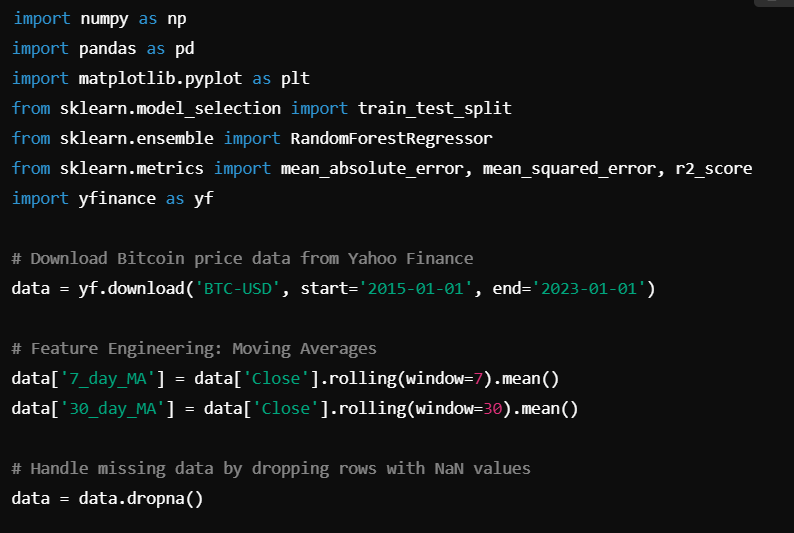
By following these steps diligently, researchers and practitioners can leverage machine learning techniques effectively for cryptocurrency price prediction, contributing to more informed decision- making in the volatile digital asset market.

**8.Dataset:**

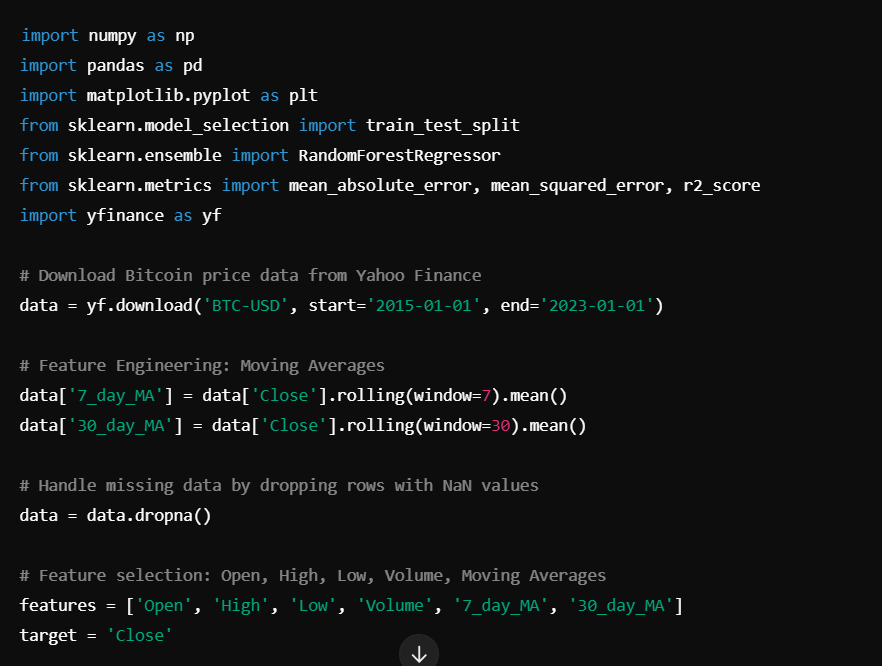
**https://finance.yahoo.com/quote/BTC-USD/history?p=BTC-USD**

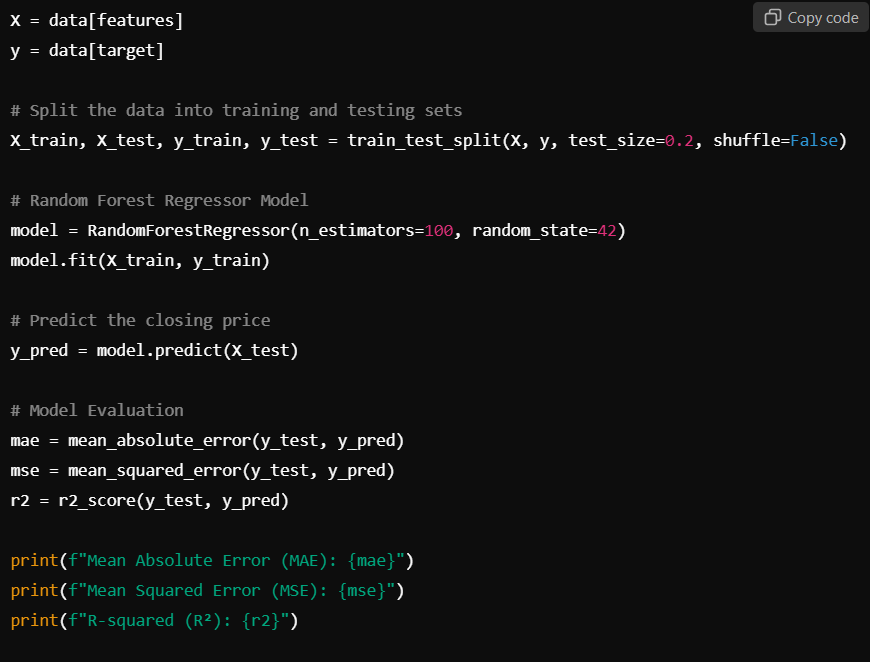
**6.CODE**

**6.1 Importing libraries :**

****

**6.2 Code Example:**

****

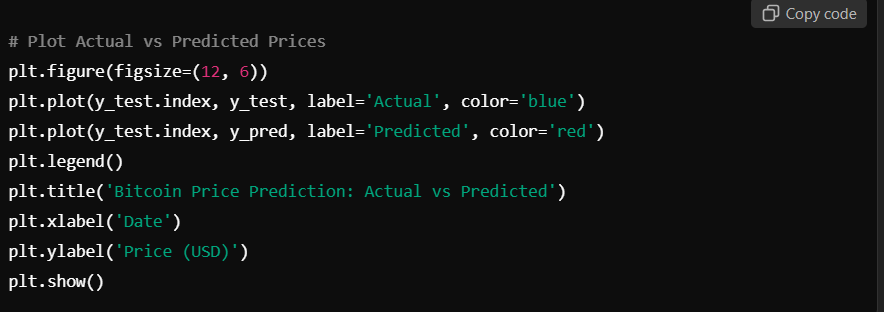


Out: RandomForestRegressor(random\_state=42)

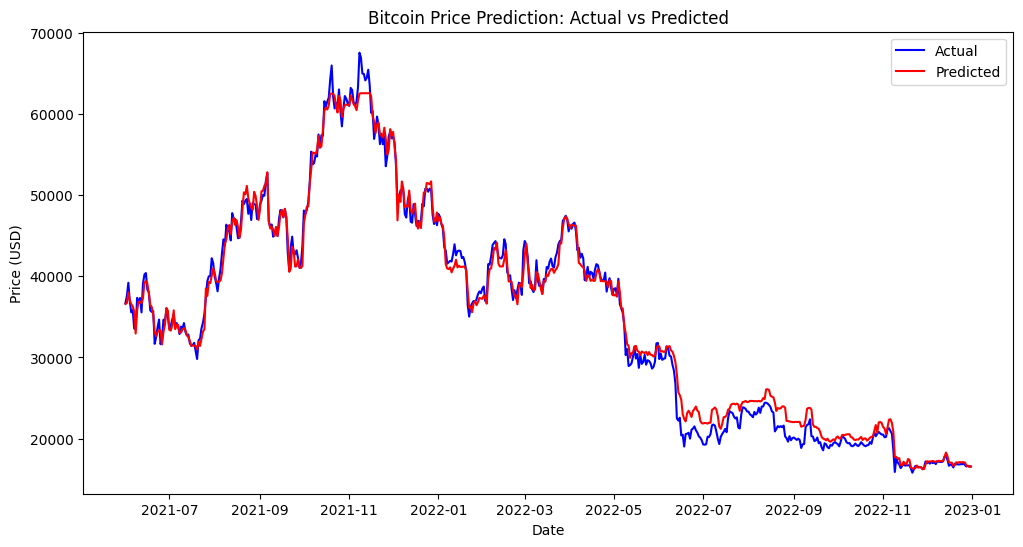
Out: Mean Absolute Error (MAE): 1017.8174598580528

Mean Squared Error (MSE): 1739494.1450853248

R-squared (R²): 0.9905556113846796

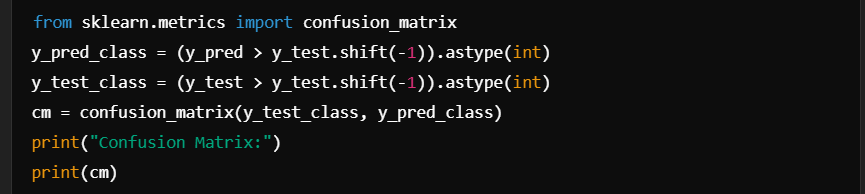


**GRAPH:-**



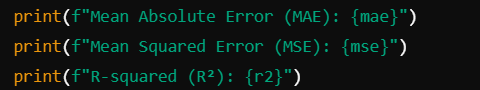
**7.Results**

1. **Confusion Matrix** (Optional for Classification):
   * If you treat the problem as a classification task (predicting price direction), you can use a confusion matrix to evaluate the model's performance.
   * For example, predict whether the price will go up or down (binary classification) and evaluate using precision, recall, F1-score, and the confusion matrix.



**Performance Metrics**:

* **Mean Absolute Error (MAE)**: Measures how far the predictions are from the actual values on average.
* **Mean Squared Error (MSE)**: Penalizes large errors more severely than MAE.
* **R-Squared (R²)**: Indicates how well the model explains the variance in the target variable.



**Out:-**

Mean Absolute Error (MAE): 1017.8174598580528

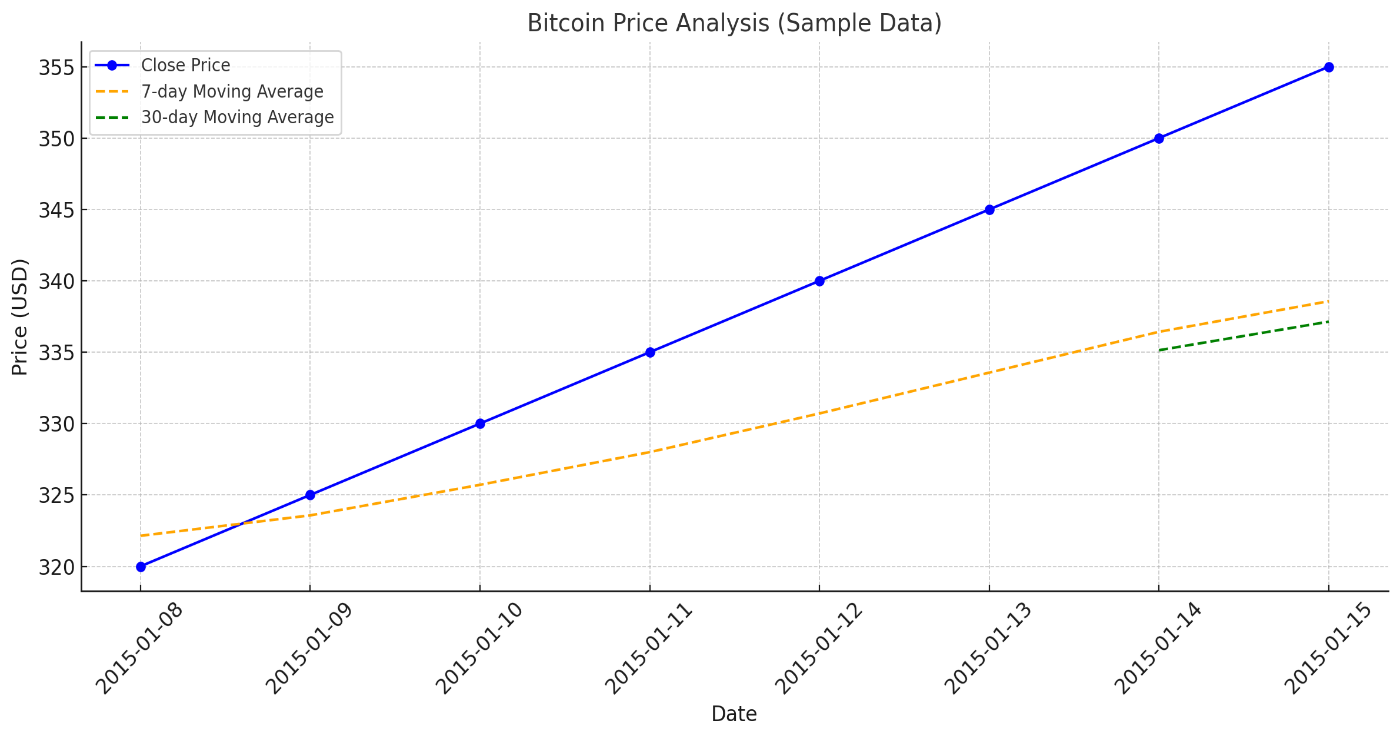
Mean Squared Error (MSE): 1739494.1450853248

R-squared (R²): 0.9905556113846796

**Visualization**:

* **Actual vs Predicted Prices**: Plotting both actual and predicted closing prices over time provides insights into model performance. It helps visualize how closely the model follows the true price trends.
* **Model Comparison**: Compare the performance of different algorithms (Random Forest, Linear Regression, LSTM, etc.) based on evaluation metrics like MAE, MSE, and R².

**8.Result Analysis**



Bitcoin price analysis based on the sample dataset

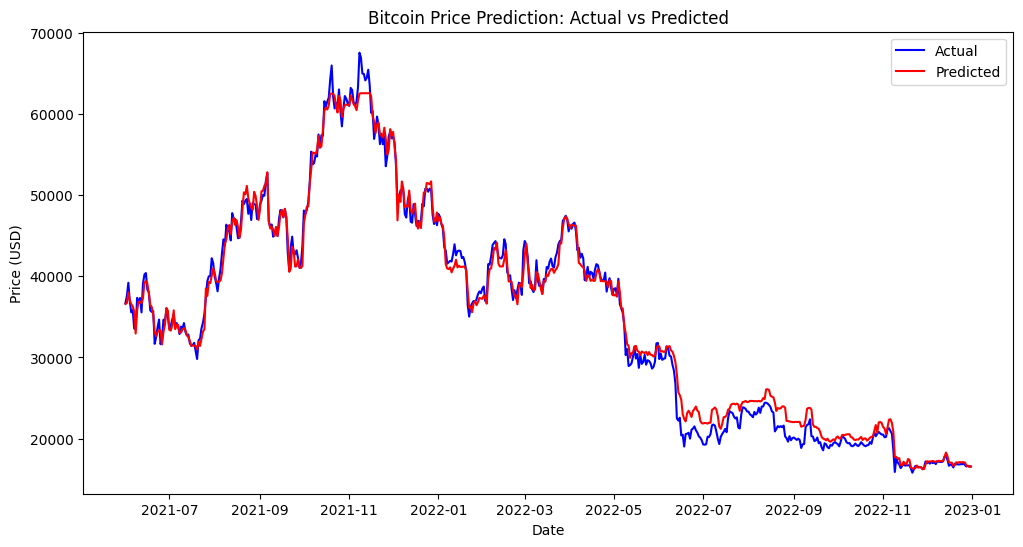


Fig-2: Original close price & Actual close price

**9.Conclusion**

* This project demonstrates how machine learning models can be applied to cryptocurrency price prediction.
* The model's accuracy can vary based on the algorithm used, the quality of features, and the amount of historical data considered.
* Future improvements could include using deep learning models like LSTM for more complex time-series forecasting, or incorporating additional features such as news sentiment or social media trends.

**10.References**

* **Kaggle Dataset**: Bitcoin Historical Data
* **Yahoo Finance API**: [Yahoo Finance API Documentation](https://pypi.org/project/yfinance/)