1. PROJECT TITLE

PREDICTING BITCOIN PRICES MOVEMENT USING TWITTER SENTIMENT AND MACHINE LEARNING.

2. PROJECT STATEMENT

The objective of this project is to predict Bitcoin's next-day price movement (up or down) by analyzing Twitter sentiment along with historical Bitcoin market data. Since Bitcoin is a highly volatile and sentiment-driven asset, the project explores how social media discussions — specifically on Twitter — influence short-term price direction.

3. HIGH LEVEL ARCHITECTURE

1. Data Collection

- **Bitcoin Price Data:** Historical BTC prices (Open, High, Low, Close, Volume) from Kaggle.
- **Twitter Sentiment Data:** Tweets mentioning Bitcoin, analyzed using BERT for sentiment classification (positive, neutral, negative).

2. Data Preprocessing

- Converted timestamps to datetime formats.
- Aggregated both Bitcoin and Twitter data to daily averages
- Created new features:

→**Sentiment features:** Average sentiment, sentiment lags.

→**Price features:** Returns, moving averages, volatility, volume change.

3. Model Training

- Split data into training and testing sets (80–20).

Trained multiple models: Logistic Regression

Naive Bayes

SVM (RBF Kernel)

Random Forest

XGBoost (best performing)

4. Model Evaluation

- Metrics used: Accuracy, Precision, Recall, F1-score, and ROC-AUC.
- Evaluated models using confusion matrices and classification reports.

RUNNING INSTRUCTIONS:

1. Place the Dataset and Files
   - `data/btcusd_1-min_data.csv`
   - `data/BTC_Tweets_Updated.csv`

2.Notebook sequence — step-by-step (run cells in this exact order) [kaggle notebook]

a. Imports & load data

b. Date/time conversion & fix columns

c. Aggregate daily (important to reduce size & align with tweets)

d. Create target and basic features

e. Feature engineering

f. Train/test split

g. Scale features

h. XGBoost (recommended)

Random Forest

i. Save test-only results

j. Plots

k. run prediction of output.

CONCLUSION RESULTS:

4. RESULTS

→ LOGISTIC REGRESSION

Accuracy - 51% ROC-AUC – 0.51 OBS: Slightly above random

-→RANDOM FOREST

Accuracy- 71% ROC-AUC – 0.71 OBS: Best traditional ML

→XGBOOST(AFTER TUNING)

Accuracy – 54% ROC-AUC – 0.54 OBS: More stable

→Naïve Bayes

Accuracy- 53% ROC-AUC – 0.53 OBS: Bias toward positive class

→Intially we implemented with logistic regression and random forest. To increase a accuracy , gone with tuning methods (XGBOOST), which improves slightly more. It is difficult to create higher accuracy, because bitcoin datasets is too noisy, and it also influenced by some many unpredictable factors