**Leveraging the momentum effect in machine learning-based cryptocurrency**

**trading**

This paper explores the application of machine learning in cryptocurrency trading, specifically targeting the momentum effect. Traditional trading models struggle to adapt to the volatile nature of cryptocurrency markets. Leveraging recent empirical evidence, the study proposes a novel approach that combines market forecasting with supervised momentum analysis. By employing machine learning, the system aims to predict the likelihood and direction of momentum in cryptocurrency prices for the next trading day. Backtesting on popular cryptocurrencies reveals promising results, with the machine learning-based strategy outperforming heuristic approaches in terms of classification accuracy and return on investment. The proposed method offers a more robust and effective solution for managing excessive price volatility in cryptocurrency trading, potentially improving trading outcomes for investors navigating these dynamic markets.

The paper utilizes historical cryptocurrency price data sourced from Kaggle, specifically CSV files for Bitcoin (BTC), Ethereum (ETH), and Litecoin (LTC) from the year 2020, each containing 10,000 data points initially. Technical indicators including Simple Moving Averages (SMA), Exponential Moving Averages (EMA), Moving Average Convergence Divergence (MACD), Relative Strength Index (RSI), Average True Range (ATR), and On-Balance Volume (OBV) are computed for each cryptocurrency.

An indicator function is applied to label each dataset sample, categorizing them into positive overreaction, negative overreaction, or normal day classifications. Utilizing some machine learning techniques, the code performs classification tasks to assign dataset samples into the respective categories based on the labeled data.

To enhance the dataset for machine learning models, additional technical indicators are incorporated as features. The classification performance metrics, including the weighted F1 measure, are reported for detecting one-day ahead overreaction conditions, showcasing the effectiveness of the implemented machine learning algorithms outlined in the paper.

1.Data Acquisition and Preparation: Import historical cryptocurrency price data for BTC, ETH, and LTC from CSV files.

2.Dataset Labeling: Apply an indicator function to label each dataset sample as positive overreaction, negative overreaction, or normal day.

3.Feature Design for ML Models: Generate various technical indicators like SMA, EMA, MACD, RSI, ATR, and OBV for each cryptocurrency.

4.Classification: Utilize the machine learning algorithm to classify dataset samples into different categories based on labeled data.

5. Classification Results: Report classification performance metrics, including weighted F1 measure.

**Inferences of your EDA / value extracted from feature extraction:**

* The dataset consists of 9981 entries and 23 columns.
* The columns include features such as Unix timestamp, time stamp, data symbol, open, high, low, close prices, volume, technical indicators (SMA5-20, EMA5-20, MACD, RSI, ATR, etc.), labels, price changes, moving averages (MA20), standard deviation (20dSTD), upper and lower bands, and volume rate of change (Volume\_ROC).
* There are missing values in some columns, particularly in MACD, RSI, OBV, Price\_Diff, MA20, 20dSTD, UpperBand, LowerBand, and Volume\_ROC.

Index: 9981 entries, 19 to 9999  
Missing Values:

Unix 0

Time Stamp 0

Data Symbol 0

Open 0

High 0

Low 0

Close 0

Volume 0

SMA5-20 0

EMA5-20 0

MACD 25

RSI 13

ATR 0

OBV 1

Price\_Diff 1

Oc 0

Label 0

Price\_Change 0

MA20 19

20dSTD 19

UpperBand 19

LowerBand 19

Volume\_ROC 1

dtype: int64  
dtypes: float64(19), int64(2), object(2)

Based on the models mentioned in the paper:

LR - Logistic Regression

GNB - Gaussian Naive Bayes

HE -Heuristic Function

KNN - K-Nearest Neighbors

MLP - Multi-Layer Perceptron

MNB - Multinomial Naive Bayes

SVC- Support Vector Classification

RF-Random Forest

RESULTS FOR BTC\_DATA

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Random\_forest | KNN | SVC | MNB | MLP | LR | GNB | HE |
| Accuracy | 0.48 | 0.44 | 0.54 | 0.47 | 0.47 | 0.46 | 0.36 | 0.49 |
| Precision | 0.48 | 0.44 | 0.53 | 0.46 | 0.44 | 0.43 | 0.42 | 0.24 |
| Recall | 0.48 | 0.43 | 0.54 | 0.47 | 0.46 | 0.46 | 0.36 | 0.5 |

2-Class

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | RFC | KNN | SVC | MNB | MLP | LR | GNB |
| BTC | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 | 0.89 |

3-Class

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | RFC | KNN | SVC | MNB | MLP | LR | GNB |
| BTC | 0.74 | 0.51 | 0.51 | 0.66 | 0.57 | 0.61 | 0.53 |

Conclusion:

1. Random Forest and SVC (Support Vector Classification) achieved the highest accuracy of 0.48 and 0.54, respectively. This indicates that these models performed relatively well in classifying the data compared to the other models.
2. Random Forest and SVC also demonstrated the highest precision scores of 0.48 and 0.53, respectively. Precision measures the proportion of true positive predictions among all positive predictions made by the model. Higher precision indicates fewer false positives.
3. Random Forest, SVC, and KNN (K-Nearest Neighbors) exhibited the highest recall scores, with SVC having the highest recall of 0.54. Recall measures the proportion of true positive predictions among all actual positive instances in the data. Higher recall indicates fewer false negatives.
4. MLP (Multi-Layer Perceptron) also showed relatively good performance across accuracy, precision, and recall metrics, with scores close to those of Random Forest and SVC.
5. Gaussian Naive Bayes (GNB) and HE (Heuristic Function) had the lowest accuracy, precision, and recall scores among the models evaluated.

In conclusion, Random Forest, SVC, and KNN appear to be the most promising models for the classification task based on the provided evaluation metrics.