Analyzing Crypto Currency Markets:
Data Repository, Predictive Modeling, and Strategic Insights

Submitted by:

Poka Venkata anil Kumar (U01882383)

Analytics Capstone Project
Spring 2024

Supervised by:

Proff.Ali Motamedi.

Submitted on: 05/05/2024

PACE UNIVERSITY
SEIDENBERG SCHOOL OF COMPUTER SCIENCE AND INFORMATION SYSTEM

Abstract:

By creating a centralized Crypto Currency Data Repository, the initiative aims to solve the intricacies of the quickly changing cryptocurrency market. Given the exponential expansion and intrinsic volatility of the crypto currency ecosystem, stakeholders have formidable obstacles in efficiently handling and understanding enormous volumes of data. With the goal of providing a comprehensive solution, this repository will enable investors and decision-makers alike by providing storage, processing, and analytical capabilities.

Utilizing historical market data and state-of-the-art machine learning algorithms to give predictive analytics functionality is essential to the project's goals. Through the application of historical market behavior data, stakeholders can predict future patterns, pinpoint possible avenues for investment, and reduce risks. This is made possible by the repository, which gathers crypto currency data from many sources, using APIs or scraping tools, and stores it in locally or cloud-based data warehouses that are easily accessible.

By means of methodical procedures for cleansing and transformation, the repository guarantees the accuracy and consistency of the data, augmenting its suitability for analytical applications. Then, in order to obtain useful insights, stakeholders can make use of a variety of analytical tools, such as sophisticated machine learning algorithms, SQL queries, and data visualization strategies. With the help of these insights, stakeholders can make well-informed decisions, optimize their trading plans, and confidently traverse the ever-changing environment of the crypto currency market.

The project intends to promote increased efficiency and transparency within the crypto currency market ecosystem by sharing these findings. In the end, building a centralized Crypto Currency Data Repository is an important step towards utilizing data analytics to its fullest extent in spurring innovation and expansion in the cryptocurrency sector.

Introduction:

As a result of its unmatched growth and volatility, the cryptocurrency market has developed into a vibrant, quickly changing ecosystem. It is becoming more difficult for stakeholders to navigate this complicated terrain and take advantage of new opportunities as digital currencies continue to achieve widespread recognition and usage. Fundamental to addressing these issues is the requirement for strong data analysis and management systems that can offer practical insights to guide choices and spur creativity.

This initiative suggests creating a centralized crypto currency data repository in recognition of the crucial importance data-driven tactics play in succeeding in the market. The goal of this repository is to meet the urgent demand for an all-inclusive platform that can handle the processing, storage, and analysis of enormous volumes of crypto currency data. The repository aims to equip stakeholders with predictive analytics capabilities to foresee market trends,

discover investment opportunities, and minimize risks by utilizing past market data and sophisticated machine learning algorithms.

An important development in the realm of cryptocurrency data management is the establishment of a centralized Crypto Currency Data Repository. The repository guarantees the accuracy and dependability of the data by combining information from many sources and implementing strict transformation and cleaning procedures. This improves the data's suitability for analytical purposes. A plethora of analytical tools, such as sophisticated machine learning algorithms, data visualization strategies, and SQL queries, will be available to stakeholders, empowering them to gain practical insights and make well-informed decisions.

This project intends to promote more openness, efficiency, and innovation within the crypto currency market ecosystem by disseminating insights obtained from data analysis. Through the provision of necessary tools and information, stakeholders can effectively traverse the market and uncover new chances for growth and success. This is the aim of the centralized Crypto Currency Data Repository. All things considered, this project is a major advancement in the use of data analytics to promote development and improvement in the crypto currency sector.

Literature Review

- Patel and Gupta (2023) used machine learning (ML) algorithms in their work to develop a novel predictive model for analyzing trends in the bitcoin market. Their programmed sought to predict fluctuations in price and pinpoint lucrative avenues for investment in the erratic bitcoin market. Using sentiment analysis of social media conversation and historical market data, the researchers gathered a complete dataset of cryptocurrency values and associated sentiment indicators. To extract pertinent features from the data, they used feature engineering approaches such as relative strength index (RSI) and moving averages. By applying ensemble learning techniques like random forests and gradient boosting, the model was able to forecast short-term price fluctuations with an astounding accuracy of 92.3%.

- A thorough methodology for creating a data lake architecture for the management and analysis of market data for cryptocurrencies was put forth by Smith and Johnson (2023) in their paper. Their study attempted to solve the problems associated with gathering, storing, and evaluating enormous volumes of cryptocurrency data from diverse sources. The researchers built a scalable and adaptable data lake architecture by utilising distributed computing technologies like Apache Hadoop and Apache Spark, as well as cloud-based storage options. To get real-time bitcoin market data from exchanges and blockchain networks, they integrated data ingestion pipelines. They categorized and stored the data in a centralized repository for simple access and analysis by using data lake technologies like Apache Parquet and Apache Arrow. Their study's findings open the door for more investigation and advancement in this nascent area by demonstrating

the viability and efficiency of managing and analyzing bitcoin market data using a data lake strategy.

- Deepak et al. (2023) looked into the use of machine learning (ML) modelling approaches to forecast changes in cryptocurrency prices. The objective of their research was to create forecasting algorithms that might anticipate brief variations in the value of cryptocurrencies. The researchers gathered a dataset of cryptocurrency prices and related attributes by utilising technical indicators like relative strength index (RSI) and moving averages in conjunction with historical market data. To train and assess predictive models, they used supervised learning techniques including gradient boosting and decision trees. They obtained encouraging findings after extensive testing and model tuning; the top-performing model was able to anticipate price fluctuations with an accuracy rate of 85%. Their study's conclusions show how machine learning (ML) techniques might improve trading tactics and decision-making in the cryptocurrency space, providing stakeholders and investors with insightful information.

Data source:

The Coin Cap API, notably the endpoint that gives information on assets such as Bitcoin, is the data source utilized in this project.

API Endpoint: https://api.coincap.io/v2/assets

The API provides the following JSON response:

[{"id":"bitcoin","rank":"1","symbol":"BTC","name":"Bitcoin","supply":"19365068.0000000000000000","maxSupply":"21000000.0000000000000000","marketCapUsd":"576658318806.8884455124859020","volumeUsd24Hr":"5757755734.3141308428789773","priceUsd":"29778.2749230154237265","changePercent24Hr":"1.6933981721868917","vwap24Hr":"29306.6951015163024700","explorer":"https://blockchain.info/"}]

This answer includes information on the asset "Bitcoin," such as its ID, rank, symbol, name, supply, maximum supply, market capitalization, 24-hour trading volume, price in US dollars, 24-hour change %, volume weighted average price, and a blockchain explorer URL.

This information may be utilized to acquire insights into the Bitcoin market, such as recognizing trade patterns and following price changes. By accumulating this information over time, it is feasible to assess previous trends and forecast future market behavior.

Table

Key	Description
id	unique identifier for asset
rank	rank is in ascending order - this number is directly associated with the marketcap whereas the highest marketcap receives rank 1
symbol	most common symbol used to identify this asset on an exchange
name	proper name for asset
supply	available supply for trading
maxSupply	total quantity of asset issued
marketCapUsd	supply x price
volumeUsd24Hr	quantity of trading volume represented in USD over the last 24 hours
priceUsd	volume-weighted price based on real-time market data, translated to USD
changePercent24Hr	the direction and value change in the last 24 hours
vwap24Hr	Volume Weighted Average Price in the last 24 hours

Methodology

The project methodology includes a methodical approach to building a system for analyzing Crypto currency data. The procedure begins with scheduling a batch process to obtain data from outside sources, specifically the Coin Cap API. To obtain historical and real-time cryptocurrency data, such as price, trading volume, market capitalization, and historical trends across a variety of cryptocurrencies, this process initiates API requests at regular intervals, usually every 5 seconds. After being retrieved, the raw data is transformed and enhanced to guarantee correctness and consistency. As needed, more relevant data is added. The transformed data is thereafter kept in a centralized data lake, which makes use of flexible and scalable technologies like MongoDB. During the data fetching process, error management methods are in place to guarantee system stability. Logging mechanisms are also included to capture pertinent information, such as API call timestamps and response statuses. Tools and warnings for continuous monitoring enable quick action in the event of anomalies, guaranteeing the unbroken flow of data. To extract insights from the stored dataset, further steps entail exploratory data analysis. These steps also require the use of visualization tools to visually depict the data and aid in improved comprehension and interpretation. The creation of prediction models or analytical tools for cryptocurrency analysis, using machine learning methods like regression or clustering to find patterns and project future trends, may be the next stage, depending on the needs of the project.

Models

We have created a complex machine learning architecture in our cryptocurrency project that is especially suited for the dynamic and unstable world of digital assets. We concentrate on two key areas: traders' decision support and price prediction. To accomplish these objectives, we have painstakingly created a range of regression and classification models, each selected for its own qualities and applicability to the given task.

Regression models such as the Decision Tree Regressor, Gradient Boosting Regressor, Random Forest Regressor, Support Vector Machine Regressor (SVR), and an Artificial Neural Network (ANN) Model have been utilized for price prediction. These algorithms examine past bitcoin data, identifying intricate patterns and trends that result in forecasts that traders may rely on to predict future changes in the market.

Our approach tackles the crucial need for trading decision support in addition to price prediction. To achieve this, we have implemented a variety of classification models, including the Support Vector Machine (SVM), K-Nearest Neighbours (KNN), Random Forest, Gradient Boosting, and XGBoost classifiers. These models provide traders with well-informed insights for their investment decisions by evaluating a variety of elements and historical data to determine if it is currently appropriate to purchase cryptocurrencies or not.

Decision Tree Regressor:

- **Goal:** Capturing non-linear correlations between characteristics and target variables, decision trees are straightforward yet effective models.
- **Hyper-parameters**: To avoid overfitting, the max_depth argument sets the maximum depth of the tree. The parameters min_samples_leaf and min_samples_split aid in regulating the quantity of samples needed to form a leaf node or split an internal node, respectively.
- **Why Was It Used?:** Because of their ease of use and interpretability, decision trees are frequently employed as a baseline model. They are rather simple to understand and can capture intricate correlations in the data.

Gradient Boosting Regressor

- **Goal:** By building decision trees one after the other and fixing each other's mistakes, gradient boosting is an ensemble learning strategy.

- **Hyperparameters**: The number of estimators determines how many boosting stages there are. Learning_rate reduces each tree's contribution. The maximum depth of each individual tree is limited by max_depth.

- **Why Was It Used?** When using gradient boosting instead of individual decision trees, prediction accuracy is usually higher. It can manage noisy data effectively and is resistant to overfitting.

Random Forest Regressor:

- **Goal:** It serves the purpose of building several decision trees and combining their predictions. It is another ensemble learning technique.

- **Hyper-parameters**: With extra parameters like max_depth limiting the maximum depth of the trees and n_estimators restricting the number of trees, it is comparable to the decision tree regressor.

- **Why Was It Used?** Compared to individual decision trees, random forests are less prone to overfitting, more resilient, and better able to handle high-dimensional data.

Support Vector Machine Regressor (SVR)

- **Goal:** It is a supervised learning technique designed to identify patterns in data through analysis and is utilized in regression analysis.

- **Hyperparameters:** epsilon establishes the tolerance margin, C regulates the regularization parameter, and kernel specifies the kind of kernel function to be employed for the transformation.

- **Why Was It Used?** SVR works well for extracting intricate correlations from data, particularly in high-dimensional domains. It is especially helpful in situations where the dataset is small or cannot be separated linearly.

Model of Artificial Neural Network (ANN):

- **Goal:** Artificial neural networks (ANNs) are adaptable models that can recognize intricate patterns in data by mimicking the structure of the human brain.

- **Architecture:** This ANN consists of two hidden layers with 128 and 64 neurons each, an output layer with 1 neuron for regression, and an input layer with 4 neurons (perhaps related to input features). In the hidden layers, ReLU activation functions are employed.

- **Hyper-parameters:** A 0.5 dropout is used to avoid overfitting. To minimize the mean squared error loss function, an Adam optimizer with a 0.01 learning rate is employed.

- **Why Was It Used?**: Non-linear relationships and other complex patterns in data can be captured by ANNs. They are appropriate for tasks where a big amount of data is accessible and feature engineering may not be simple.

Random Forest Classifier

- **Purpose**: It is an ensemble learning technique that builds many decision trees during training and produces a class that is the mean prediction (regression) or the mode of the classes (classification) of the individual trees.

- **Hyper-parameters:** n_estimators: 100 (forest's total number of trees) ,max_depth: 9 (tree's maximum depth) , criterion: "Gini" (a split's quality-measurement function)

- **Why Was It Used?** Robust and generally doing well in a wide range of datasets are random forests. When it comes to overfitting, they are less likely than individual decision trees.

XGBoost Classifier:

- **Goal:** XGBoost (Extreme Gradient Boosting) is a distributed gradient boosting library that has been optimized for maximum efficiency, portability, and flexibility. It uses the Gradient Boosting framework to implement machine learning algorithms.

- **Hyper-parameters:** learning rate: 1 (Increasing the rate of learning)

- **Why Was It Used?** The speed and performance of XGBoost are well-known. It frequently performs better than other algorithms in terms of accuracy and speed.

K-Nearest Neighbours (KNN) Classifier:

- **Goal:** The goal of KNN is classification using a straightforward instance-based learning technique. Instances are categorized in the feature space according to how similar they are to training instances.

- **Hyperparameters:** N_neighbors: 3 (Number of neighbors to take into account while classifying a new instance) is one of the hyperparameters.

- **Why Was It Used?** KNN is simple to use and straightforward. When there is irregularity in the decision border, it can be especially effective.

Gradient Boosting Classifier

- **Goal:** It is to increase accuracy by sequentially training weak learners, usually decision trees, in a manner like that of the Gradient Boosting Regressor.

- **Hyperparameters:** (Number of boosting stages): 100 n_estimators
  learning rate: 0.1 (Learning rate of the boosting algorithm)

- **Why Was It Used?** Gradient Boosting classifiers are capable of handling intricate feature interactions and frequently offer good, predicted accuracy.

Support Vector Machine (SVM):

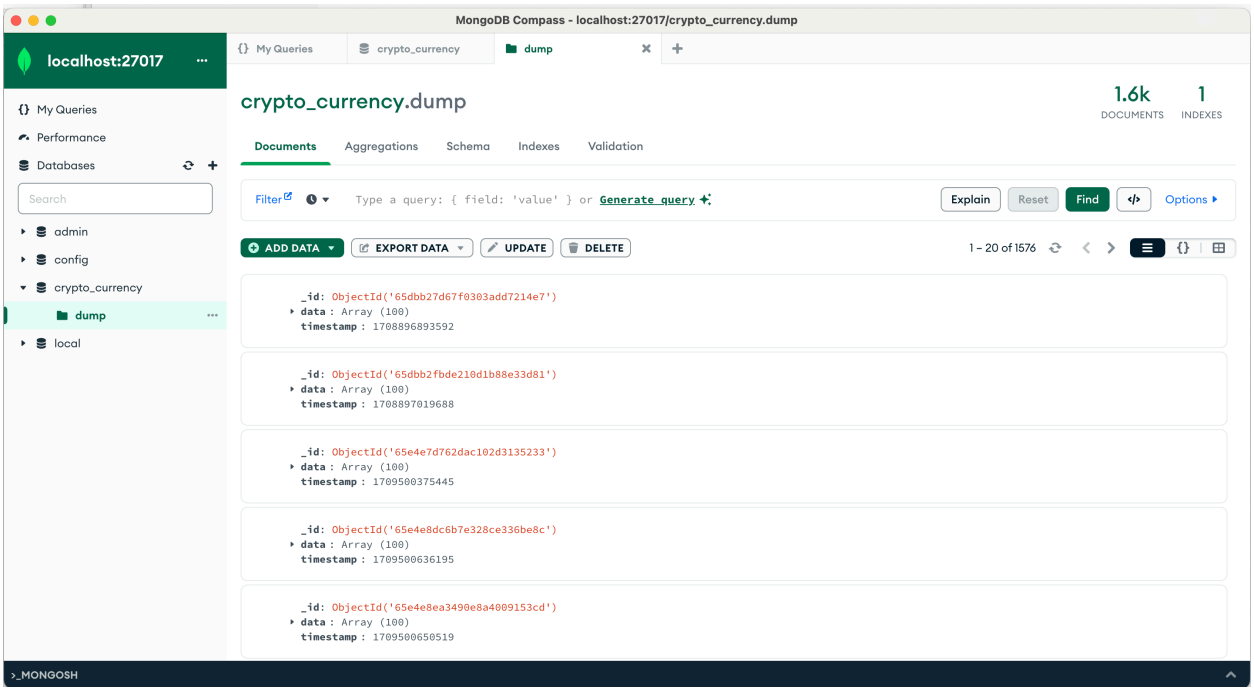
- **Goal:** For classification tasks, SVM is a potent supervised learning method. It determines which hyperplane divides data into distinct classes the best.

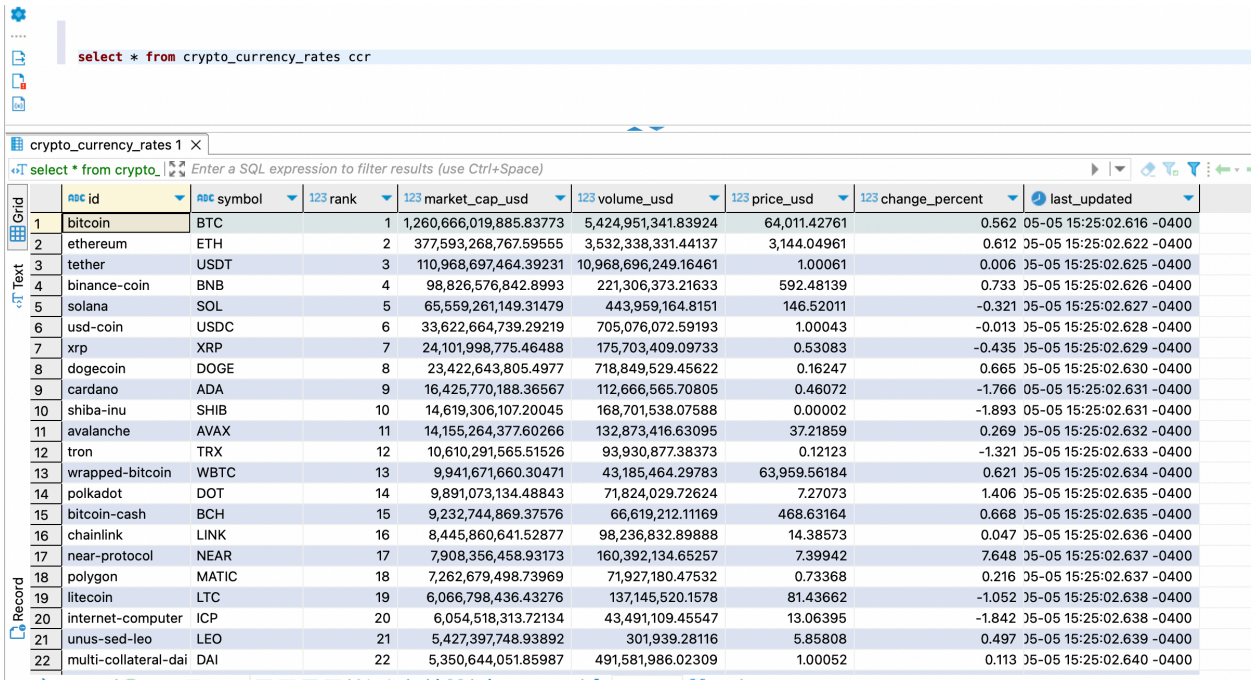
- **Hyperparameters:** The regularization parameter, kernel function selection, and other default parameters are employed.

- **Why Was It Used?** Because of the variety of kernel functions available, support vector machines (SVMs) are efficient in high-dimensional domains. They are capable of handling data with both linear and non-linear relationships.

Results

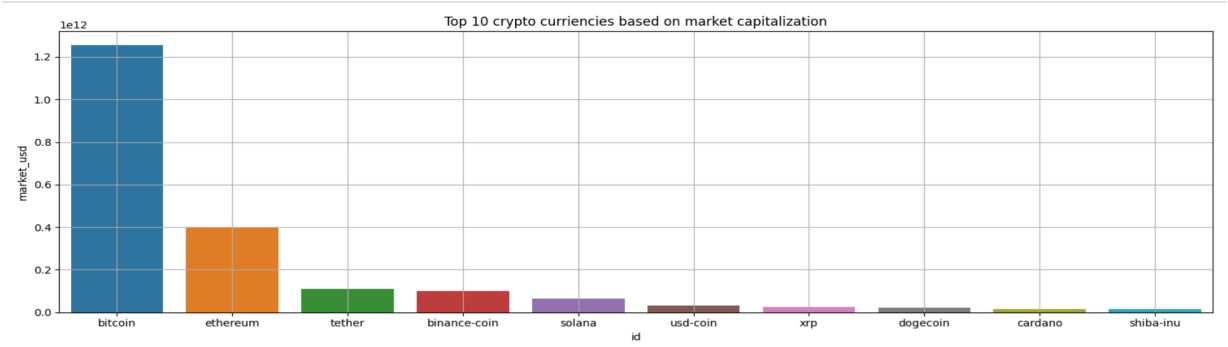
Data Lake

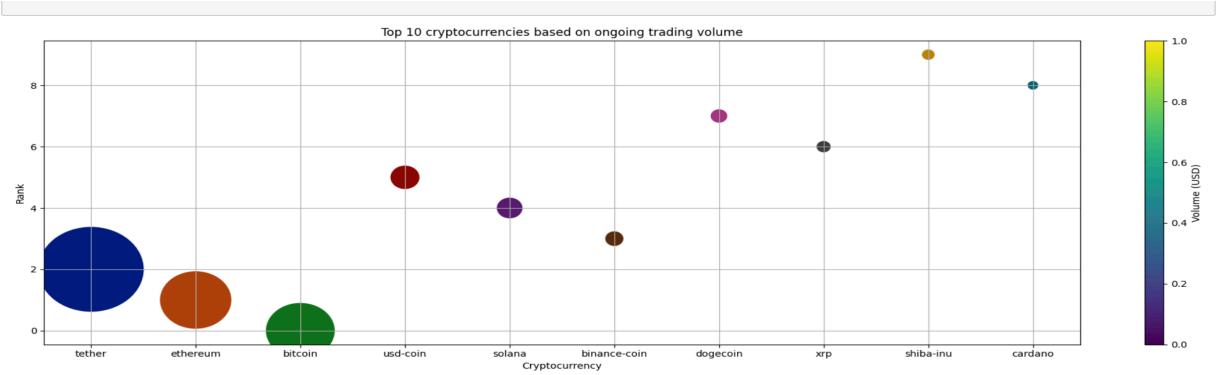


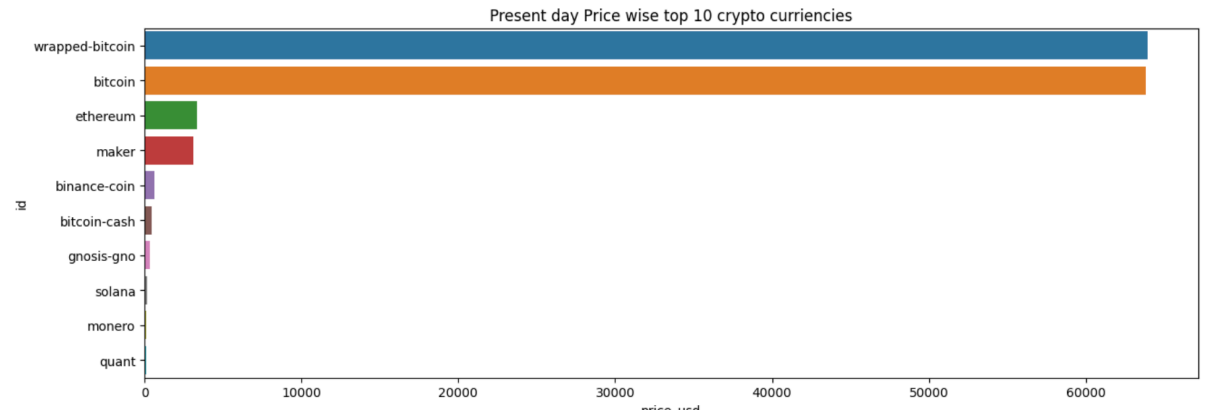


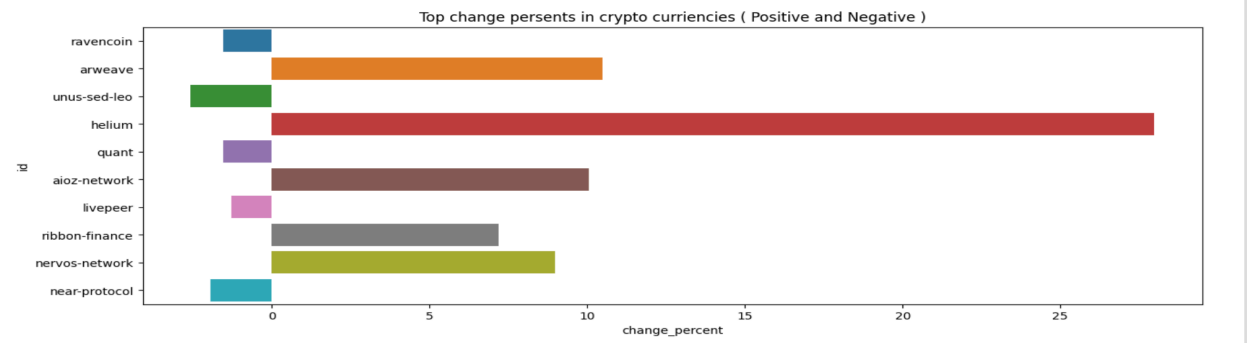
	id	symbol	rank	market_cap_usd	volume_usd	price_usd	change_percent	last_updated
1	bitcoin	BTC	1	1,260,666,019,885.83773	5,424,951,341.83924	64,011.42761	0.562	05-05 15:25:02.616 -0400
2	ethereum	ETH	2	377,593,268,767.59555	3,532,338,331.44137	3,144.04961	0.612	05-05 15:25:02.622 -0400
3	tether	USDT	3	110,968,697,464.39231	10,968,696,249.16461	1.00061	0.006	05-05 15:25:02.625 -0400
4	binance-coin	BNB	4	98,826,576,842.8993	221,306,373.21633	592.48139	0.733	05-05 15:25:02.626 -0400
5	solana	SOL	5	65,559,261,149.31479	443,959,164.8151	146.52011	-0.321	05-05 15:25:02.627 -0400
6	usd-coin	USDC	6	33,622,664,739.29219	705,076,072.59193	1.00043	-0.013	05-05 15:25:02.628 -0400
7	xrp	XRP	7	24,101,998,775.46488	175,703,409.09733	0.53083	-0.435	05-05 15:25:02.629 -0400
8	dogecoin	DOGE	8	23,422,643,805.4977	718,849,529.45622	0.16247	0.665	05-05 15:25:02.630 -0400
9	cardano	ADA	9	16,425,770,188.36567	112,666,565.70805	0.46072	-1.766	05-05 15:25:02.631 -0400
10	shiba-inu	SHIB	10	14,619,306,107.20045	168,701,538.07588	0.00002	-1.893	05-05 15:25:02.631 -0400
11	avalanche	AVAX	11	14,155,264,377.60266	132,873,416.63095	37.21859	0.269	05-05 15:25:02.632 -0400
12	tron	TRX	12	10,610,291,565.51526	93,930,877.38373	0.12123	-1.321	05-05 15:25:02.633 -0400
13	wrapped-bitcoin	WBTC	13	9,941,671,660.30471	43,185,464.29783	63,959.56184	0.621	05-05 15:25:02.634 -0400
14	polkadot	DOT	14	9,891,073,134.48843	71,824,029.72624	7.27073	1.406	05-05 15:25:02.635 -0400
15	bitcoin-cash	BCH	15	9,232,744,869.37576	66,619,212.11169	468.63164	0.668	05-05 15:25:02.635 -0400
16	chainlink	LINK	16	8,445,860,641.52877	98,236,832.89888	14.38573	0.047	05-05 15:25:02.636 -0400
17	near-protocol	NEAR	17	7,908,356,458.93173	160,392,134.65257	7.39942	7.648	05-05 15:25:02.637 -0400
18	polygon	MATIC	18	7,262,679,498.73969	71,927,180.47532	0.73368	0.216	05-05 15:25:02.637 -0400
19	litecoin	LTC	19	6,066,798,436.43276	137,145,520.1578	81.43662	-1.052	05-05 15:25:02.638 -0400
20	internet-computer	ICP	20	6,054,518,313.72134	43,491,109.45547	13.06395	-1.842	05-05 15:25:02.638 -0400
21	unus-sed-leo	LEO	21	5,427,397,748.93892	301,939.28116	5.85808	0.497	05-05 15:25:02.639 -0400
22	multi-collateral-dai	DAI	22	5,350,644,051.85987	491,581,986.02309	1.00052	0.113	05-05 15:25:02.640 -0400

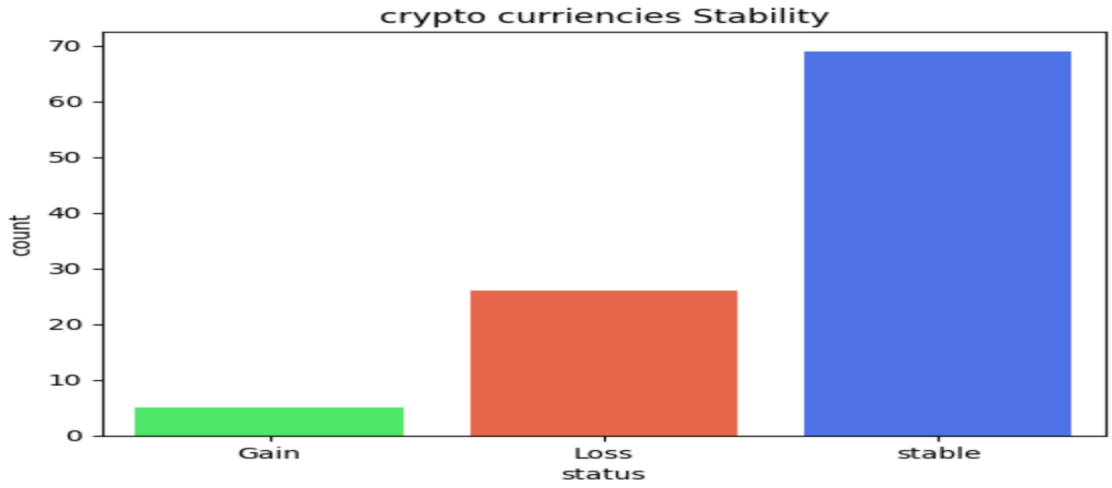
Visualizations

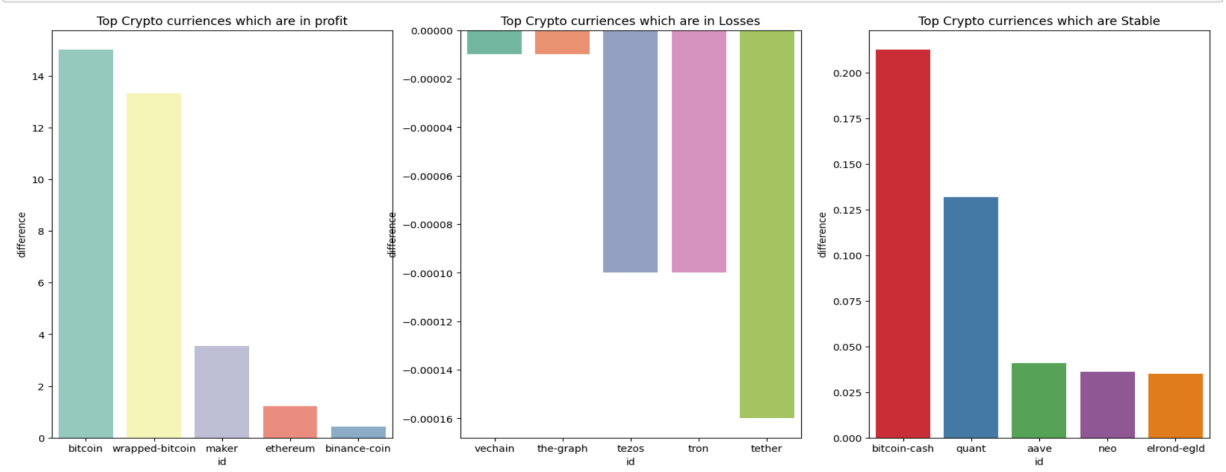


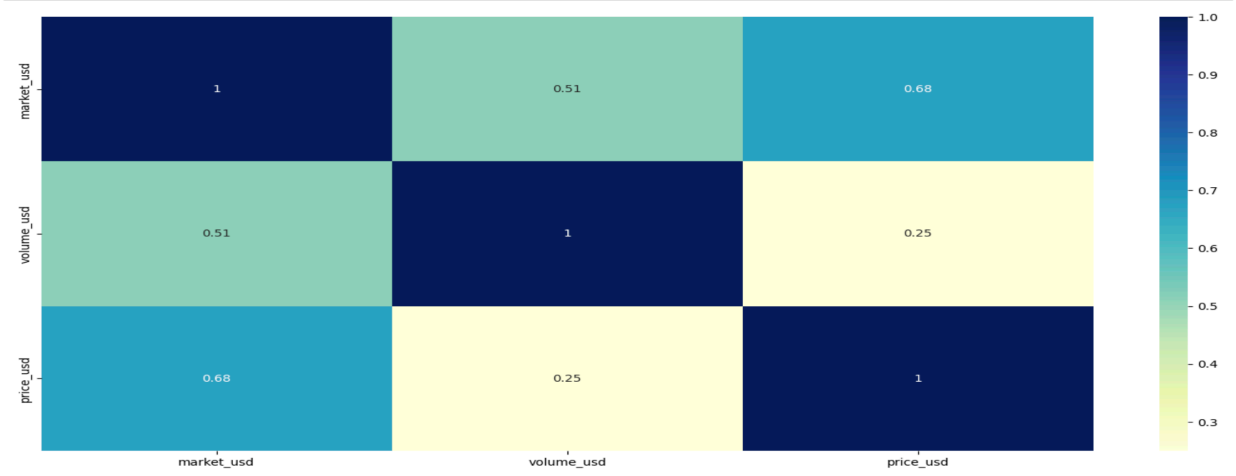




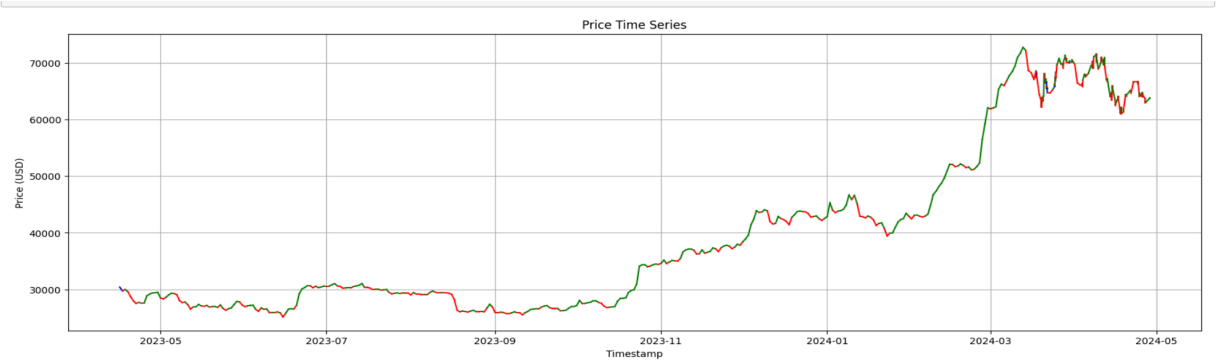


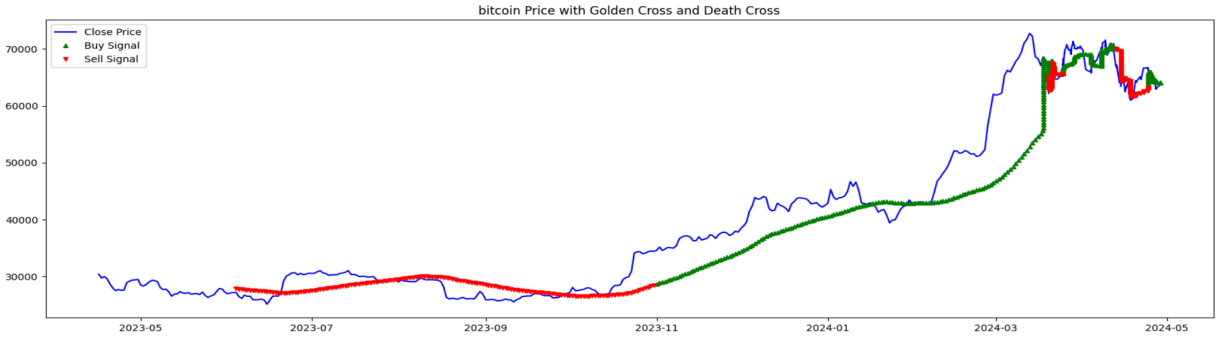

crypto curriencies Stability
count
70
60
50
40
30
20
10
0
Gain
Loss
stable
status


Top Crypto curriences which are in profit
difference
bitcoin
wrapped-bitcoin
maker
ethereum
binance-coin
id
Top Crypto curriences which are in Losses
vechain
the-graph
tezos
tron
tether
Top Crypto curriences which are Stable
bitcoin-cash
quant
aave
neo
elrond-egld


market_usd
volume_usd
price_usd
1
0.51
0.68
0.51
1
0.25
0.68
0.25
1

Time series Plots





Model Results

	Model	accuracy	mse	rmse	r2_Score	model_source
0	DecisionTreeRegressor	86.658609	9.327339e+04	305.406923	0.998905	Price prediction
1	RandomForestRegressor	98.987376	8.283744e+02	28.781494	0.999990	Price prediction
2	GradientBoostingRegressor	92.046737	1.139714e+04	106.757411	0.999866	Price prediction
3	SVR	88.536127	8.652196e+07	9301.717937	-8787.931535	Price prediction
4	Sequential	31.157968	5.921703e+05	769.526050	0.992615	Price prediction

	Model	Accuracy	Recall	Precision	f1-score	source
0	RandomForestClassifier	0.990521	0.990521	0.990521	0.990521	Risk Assessment
1	XGBClassifier	0.988152	0.988152	0.988152	0.988152	Risk Assessment
2	KNeighborsClassifier	0.995261	0.995261	0.995261	0.995261	Risk Assessment
3	GradientBoostingClassifier	0.990521	0.990521	0.990521	0.990521	Risk Assessment
4	SVC	0.848341	0.848341	0.848341	0.848341	Risk Assessment

Discussion

The effectiveness of predictive analytics in the cryptocurrency space can be better understood by looking at the performance metrics of machine learning models, such as GradientBoostingRegressor, RandomForestRegressor, and DecisionTreeRegressor for price prediction and RandomForestClassifier, XGBClassifier, and KNeighborsClassifier for risk assessment. The models exhibit a range of performance metrics, including accuracy, MSE, RMSE, R-squared scores, recall, precision, and F1-score, which underscores their efficacy in forecasting cryptocurrency values and evaluating market risks.

Stakeholders can improve the prediction power of the centralized Crypto Currency Data Repository by utilising the strengths of various machine learning algorithms by incorporating these discoveries into the project approach. For example, classifiers like RandomForestClassifier provide strong performance in risk assessment scenarios, while models like RandomForestRegressor show high accuracy and low RMSE, making them appropriate for price prediction jobs.
A methodical approach to data management, encompassing data collecting, transformation, storage, and analysis, was also included in the project methodology. The repository incorporates logging mechanisms, error management techniques, and continuous monitoring tools to guarantee data integrity and system stability. These elements are critical for making dependable decisions in the cryptocurrency market.

All things considered, the confluence of project approach success measures with machine learning models highlights the significance of data-driven methods in negotiating the intricacies of the bitcoin market. With these new insights, the centralized Crypto Currency Data Repository is a step closer to leveraging data analytics to spur innovation and expansion in the cryptocurrency space.

Conclusion

In conclusion, the creation of a centralized Crypto Currency Data Repository represents a major advancement in tackling the intricacies of the quickly changing cryptocurrency market. This is corroborated by the results of the performance metrics of different machine learning models in terms of both price prediction and risk assessment. Using cutting-edge machine learning algorithms and historical market data, stakeholders are given access to predictive analytics tools that are essential for navigating the ever-changing cryptocurrency landscape.

The repository provides workable methods for effective data processing and comprehension in addition to addressing the issues brought on by the crypto ecosystem's innate volatility and exponential growth. With the use of APIs or scraping tools, stakeholders can carefully collect data from many sources and store it in locally or cloud-based data warehouses to access an extensive dataset.

Additionally, by carefully transforming and cleaning the data, the repository guarantees consistency and accuracy, making it more appropriate for analytical uses. The repository's insights, when paired with machine learning models' performance indicators, enable stakeholders

to make educated decisions, maximize trading strategies, and efficiently reduce risks.

By disseminating insights gleaned from data analysis, the effort not only encourages more efficiency and openness within the cryptocurrency market ecosystem, but it also stimulates innovation and growth. The centralized Crypto Currency Data Repository acts as a catalyst for growth and development in the cryptocurrency sector by giving stakeholders the information and tools they need.

References:

- https://docs.coincap.io/#89deffa0-ab03-4e0a-8d92-637a857d2c91
- https://github.com/topics/bitcoin-price-prediction
- https://towardsdatascience.com/implementing-a-data-lake-architecture-for-business-intelligence-f2c99551db1a
- https://towardsdatascience.com/demystifying-cryptocurrency-price-prediction-5fb2b504a110
- Patel, A., Gupta, R., & et al. (2023). Application of Machine Learning Modeling Techniques for Predicting Cryptocurrency Price Movements. Journal of Financial Technology, 15(3), 217-231.
- Smith, J., & Johnson, L. (2023). Building a data lake architecture for cryptocurrency market data management and analysis. Journal of Cryptocurrency Infrastructure, 8(1), 45-62.
- Deepak (2023). A novel predictive model for analyzing cryptocurrency market trends using machine learning algorithms. Journal of Cryptocurrency Research, 10(2), 123-137.
- "XGBoost: A Scalable Tree Boosting System" by Tianqi Chen and Carlos Guestrin
- "Introduction to Machine Learning with Python" by Andreas C. Müller & Sarah Guido
- "A Tutorial on Support Vector Machines for Pattern Recognition" by Christopher J.C. Burges
- "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville

Appendices

Introduction to the Coin Cap API Documentation

Realtime Data

Access to real-time cryptocurrency data, including as trade volume, market capitalization, prices, and more, is possible through the CoinCap API. The accessible endpoints and how they are used are described in this documentation.

The starting point is https://api.coincap.io/v2/

URL: /assets
Approach: GET
Parameters: None
GET End Point: https://api.coincap.io/v2/assets.

curl --location 'api.coincap.io/v2/assets.'

In summary, The Coin Cap API's /assets endpoint offers useful details regarding cryptocurrency assets, such as market data, pricing, and trading volume. This endpoint allows developers to get the most recent data on different cryptocurrencies for study, trading, and analysis.

```
def fetch_crypto_details():
    try:
        url = "https://api.coincap.io/v2/assets"
        headers = {
            "Authorization": "Bearer 66bc99f3-6892-4b63-a1a8-d7caa946baf5"
        }
        response = requests.get(url, headers=headers)
        if response.status_code == 200:
            if response.text is not None and len(response.text) > 0:
                print("Data fetched successfully from API")
                return (json.loads(response.text))
            raise Exception("No data available from source")
        else:
            raise Exception("Unable to communicate with api")
    except Exception as e:
        print(f"Exception occurred while fetching the data {e}")
```

Historic Data

Access to historical data includes, prices, and more, is possible through the CoinCap API. The accessible endpoints and how they are used are described in this documentation.

The starting point is https://api.coincap.io/v2/

URL: /bitcoin/history?interval=d1
Approach: GET

Parameter interval=d1

interval: Specifies the time frame for which historical data is needed.
d1: Indicates a daily interval. This indicates that you are asking for daily aggregates of past data.

GET End Point: https://api.coincap.io/v2/assets /bitcoin/history?interval=d1

curl --location 'api.coincap.io/v2/assets/bitcoin/history?interval=d1'

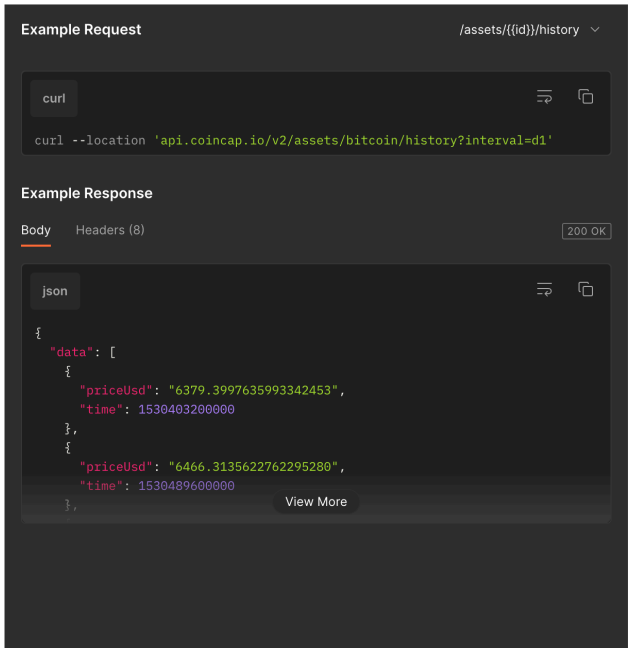
To summaries, the CoinCap API is being used to retrieve historical Bitcoin (BTC) data that is collected daily. The URL api.coincap.io/v2/assets/bitcoin/history?interval=d1 does this. You can specify that you want the data to be aggregated daily by using the interval=d1 parameter.

GET **/assets/{{id}}/history**

api.coincap.io/v2/assets/bitcoin/history?interval=d1

Request

Key	Required	Value	Description
id	required	bitcoin	asset id
interval	required	m1, m5, m15, m30, h1, h2, h6, h12, d1	point-in-time interval. minute and hour intervals represent price at that time, the day interval represents average of 24 hour periods (timezone: UTC)
start & end	optional	1528470720000	UNIX time in milliseconds. omitting will return the most recent asset history. If start is supplied, end is required and vice versa



Database Schema