CASSANDRA METRICS
TECH SOLUTIONS

# Cassandra Metrics

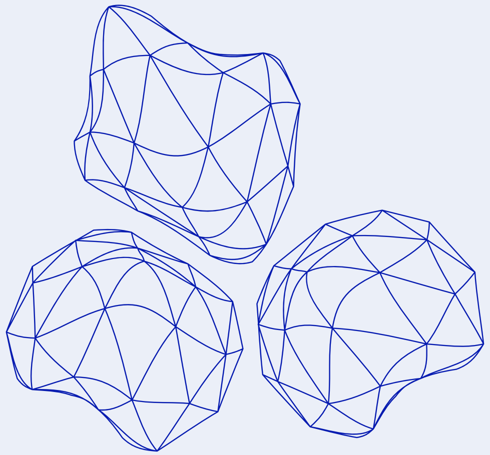
**- BETA VERSION -**

**MID- BOOTCAMP PROJECT - DATA ANALYTICS**

# Cassandra Metrics

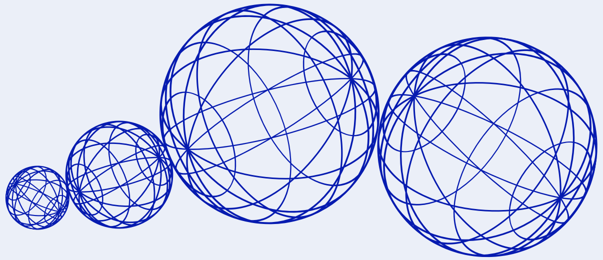
Cassandra Metrics is an AI-driven forecast platform designed to offer users valuable insights and predictions about cryptocurrencies. By harnessing the power of historical data, Cassandra Metrics aims to employ sophisticated algorithms and machine learning techniques to analyze and interpret complex patterns and trends in the cryptocurrency market.

# Why Cassandra Metrics?



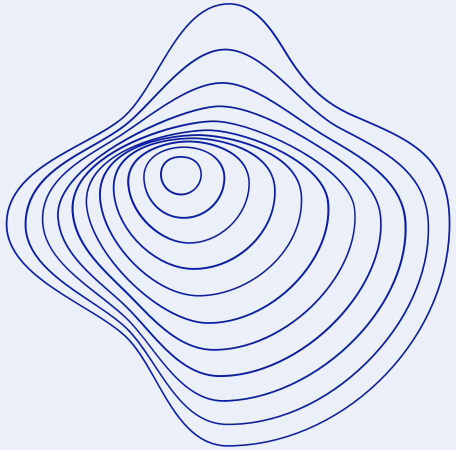
### Data-Driven Insights

Gain valuable and actionable information from comprehensive data analysis.



### Cutting-edge Technology

AI-driven platform utilizes advanced algorithms for precise predictions.



### Prediction Insights

Make informed decisions based on reliable forecasts.

# Cassandra Beta - Data

### Stage 1

The Cassandra Metrics project analyzes Bitcoin price movements using a limited dataset of 90 days and 10 statistical metrics (3 indexes collected manually). It aims to check different correlations of Bitcoin prices.

### Stage 2

The project's objective is to explore metric relationships, analyze correlations, and predict Bitcoin prices. It is in beta with limited data but provides insights into metric relationships on Bitcoin prices.

### Stage 3

In the next stage, AI and machine learning techniques will be utilized to enhance analysis and prediction for multiple cryptocurrencies, improving accuracy and reliability.

| | Date | Price | Open | High | Low | Vol. | Change % | FGI | Puell Multiple | Mayer Multiple |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Apr 01, 2023 | 28456.100000 | 28473.700000 | 28795.100000 | 28285.600000 | 38.090000 | -0.060000 | 61 | 1.170000 | 1.400000 |
| **1** | Apr 02, 2023 | 28198.300000 | 28456.100000 | 28522.800000 | 27871.700000 | 45.040000 | -0.910000 | 63 | 1.230000 | 1.380000 |
| **2** | Apr 03, 2023 | 27802.100000 | 28194.700000 | 28458.400000 | 27256.900000 | 98.000000 | -1.410000 | 63 | 1.080000 | 1.360000 |
| **3** | Apr 04, 2023 | 28164.400000 | 27802.200000 | 28429.100000 | 27668.900000 | 64.060000 | 1.300000 | 62 | 1.340000 | 1.370000 |
| **4** | Apr 05, 2023 | 28173.500000 | 28164.400000 | 28744.400000 | 27823.500000 | 77.320000 | 0.030000 | 62 | 1.170000 | 1.370000 |

Fig 1. Statistical Metrics of the Cassandra Beta

# Bitcoin Metrics

### Fear and Greed Index

A metric that measures the sentiment and emotions of cryptocurrency market participants, indicating levels of fear and greed. It helps assess market extremes and potential buying or selling opportunities.

### Puell Multiple

A metric that compares the daily issuance of new Bitcoins to its historical average, offering insights into Bitcoin's valuation relative to miner revenue. It helps identify periods of overvaluation or undervaluation in the market.

### Mayer Multiple

A metric that divides the current price of Bitcoin by its 200-day moving average, providing an indication of Bitcoin's long-term value. It helps identify potential buying or selling opportunities based on historical price.

# F&G Index

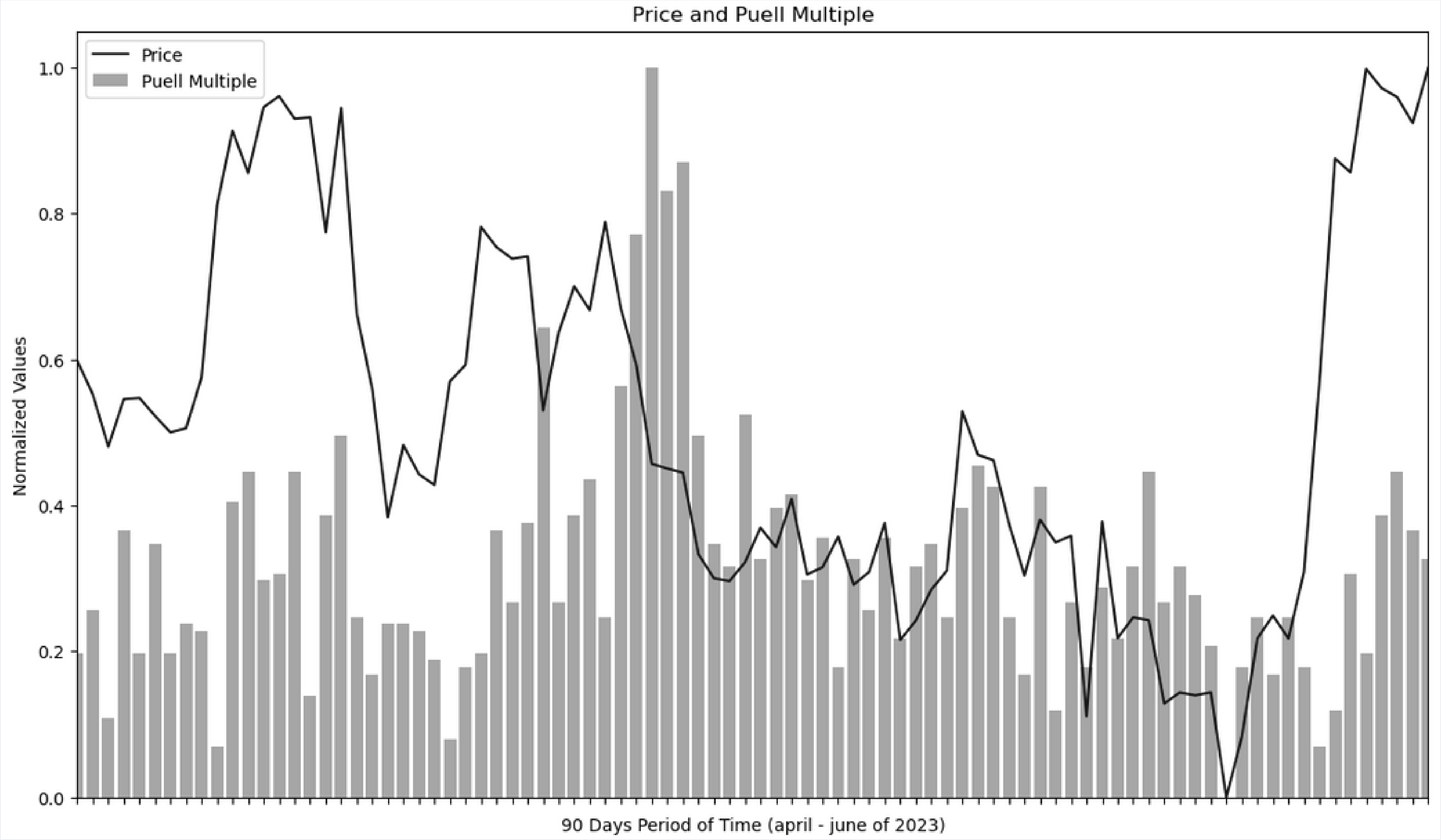
**Correlation of Price with F&G Index: 0.7498074750801293**

The correlation of 0.75 between Bitcoin price and the Fear and Greed Index (FGI) indicates a moderately strong positive relationship. This suggests that market sentiment, as captured by the FGI, plays a meaningful role in Bitcoin price movements.

# Puell Multiple

**Correlation of Price with Puell Multiple: 0.11719329315612984**

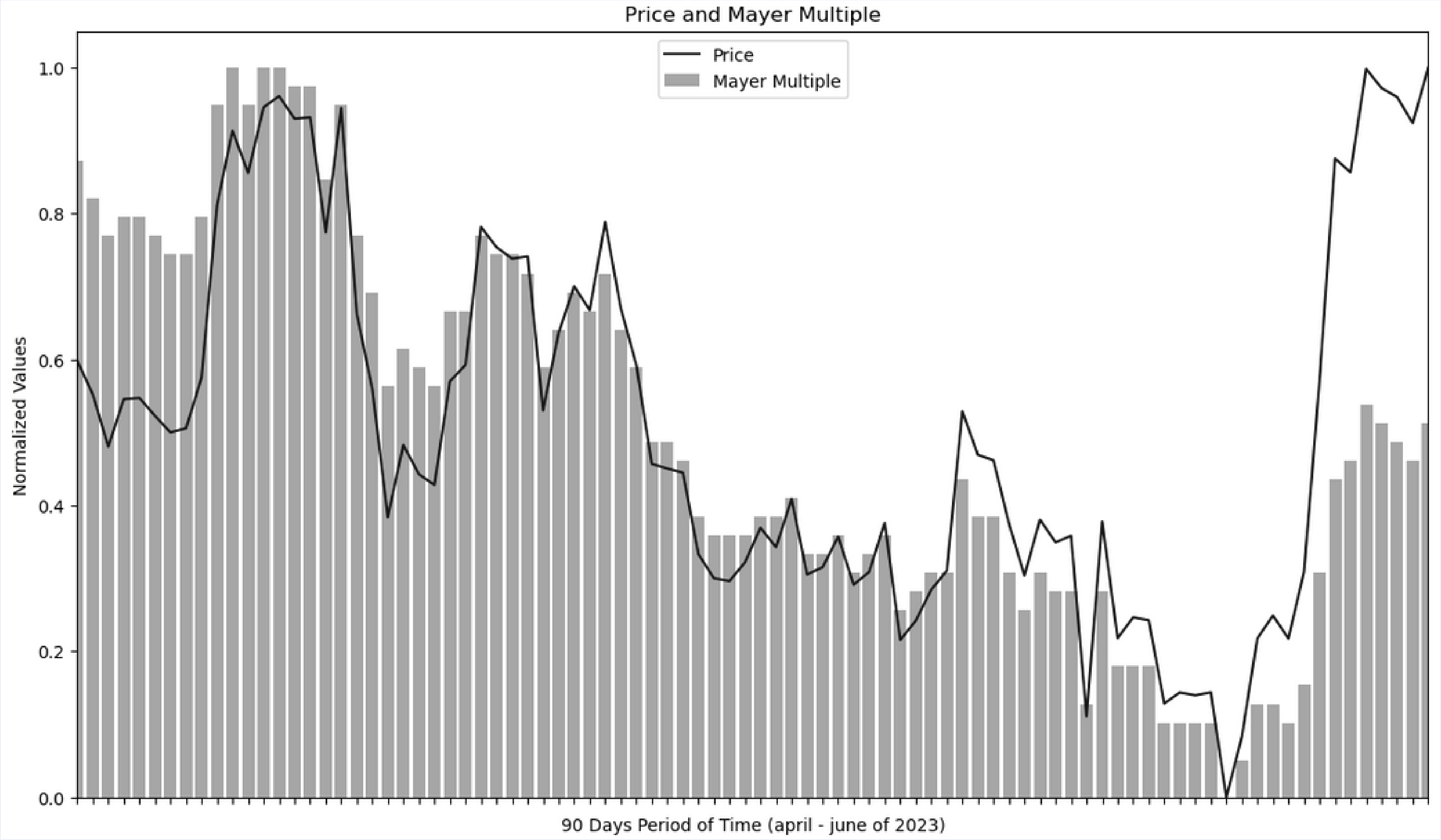
The correlation of 0.12 between Bitcoin price and the Puell Multiple indicates a weak positive relationship. This suggests that the valuation of Bitcoin based on miner revenue, as represented by the Puell Multiple, has limited impact on short-term price changes.



# Mayer Multiple

**Correlation of Price with Mayer Multiple: 0.795719157361489**

The correlation of ~ 0.80 between Bitcoin price and the Mayer Multiple suggests a strong positive relationship. This indicates that the long-term value assessment provided by the Mayer Multiple closely aligns with Bitcoin price movements.



# HYPOTHESIS TESTING (p-vlaue)

## Concept

Hypothesis testing is a statistical method used to assess the validity of a claim or assumption about a population. The p-value represents the probability of obtaining results as extreme or more extreme than the observed data, assuming the null hypothesis is true. In this case we created 2 comparable sets of dataframes.

- set 1: df_1, df_2 and df_3 = [(0, 29), (30, 59), (60, 89)]) .
- set 2: df_a, df_b and df_c = sorted_data.head(40); sorted_data.tail(40); by price.

## Set the hypothesis

Formulating the correlation hypotheses for Bitcoin Price and indexes.

## Choose the ranges

Determining the desired ranges for correlation strength.

## Get p-value

Calculating p-values to assess correlation significance.

# F&G Index (p-value)

**Stats: Price and F&G Index**
**p-value of 0.054**

The correlation analysis between Bitcoin price and the Fear & Greed Index (FGI) across the three datasets indicates a positive relationship. In **df_1**, the correlation coefficient is **0.563**, suggesting a moderate association. In **df_2**, the correlation strengthens to **0.787**, indicating a stronger positive relationship. In **df_3**, the correlation further increases to **0.857,** implying a robust positive association. **The one-way ANOVA** (using df_1, df_2 and df_3) test yields a p-value of **0.054**, suggesting that the difference in means of the Price to FGI ratio across the datasets is marginally significant.

# Puell Multiple (p-value)

**Stats: Price and Puell Multiple**
**p-value of 0.00057**

The correlation between Bitcoin price and the Puell Multiple shows varying associations across the three datasets. In **df_1,** the correlation coefficient is **0.472,** suggesting a moderate positive relationship. In **df_2**, the correlation weakens to **0.258**, indicating a weaker positive association. In **df_3**, the correlation further decreases to **0.326**, implying a relatively weaker positive relationship. **The one-way ANOVA** (using df_a, df_b and df_c) test yields a significantly low p-value of **0.00057** indicating a strong evidence against the null hypothesis, significant differences between groups.

# Mayer Multiple (p-value)

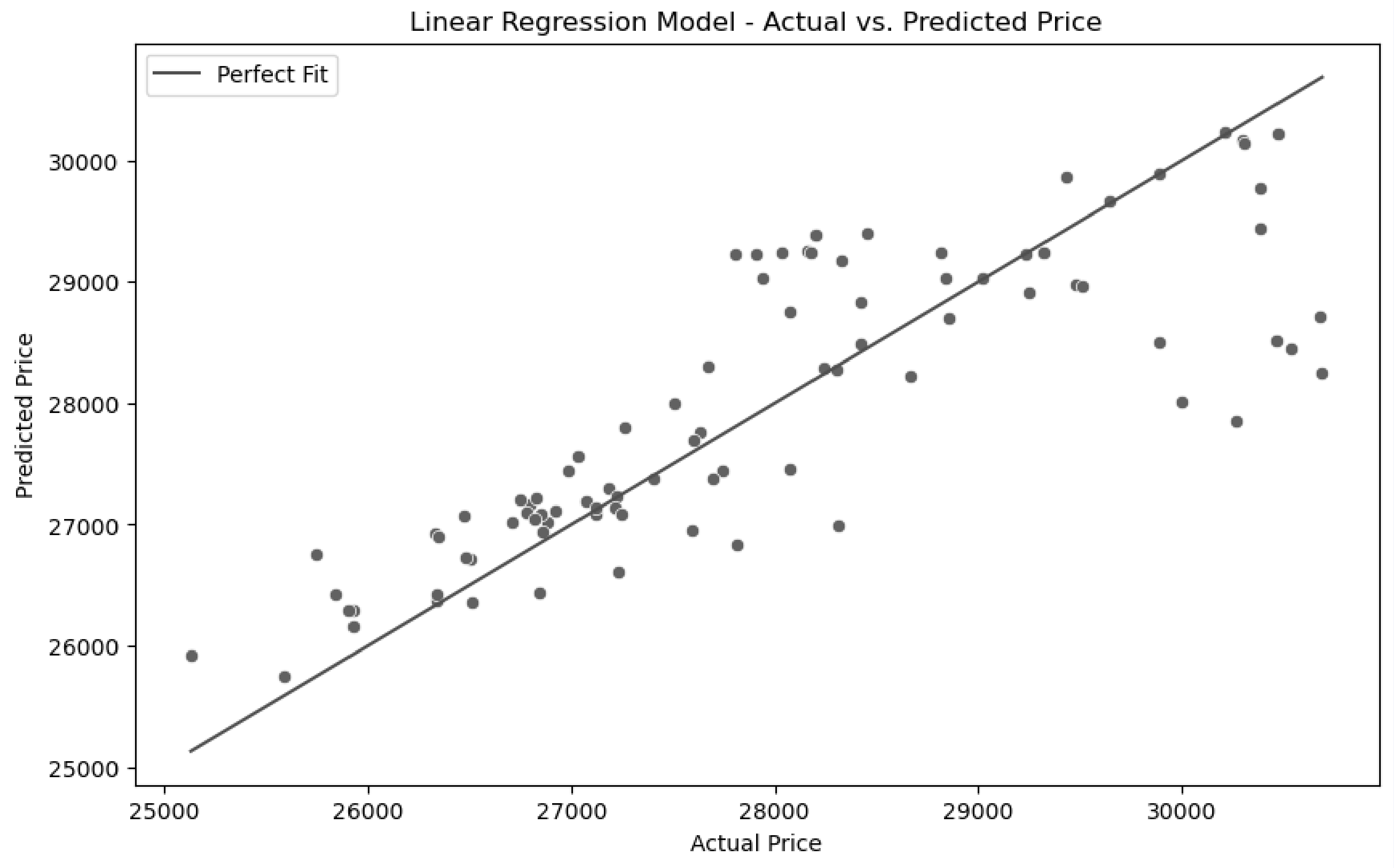
**Stats: Price and Mayer Multiple**
**p-value of 0.998**

The correlation between Bitcoin price and the Mayer Multiple demonstrates a consistently strong positive relationship across all three datasets. In **df_1**, the correlation coefficient is **0.850**, indicating a significant association. In **df_2**, the correlation strengthens to **0.964**, highlighting a very strong positive relationship. In **df_3**, the correlation further increases to **0.969**, indicating an even stronger positive association. **The one-way ANOVA** (using df_a, df_b and df_c) test yields an extremely high p-value of **0.998**, so there is no evidence against the null hypothesis, no significant differences between groups.

# Linear Regression

**Inefficiency of the Linear Regression Model**

The linear regression for BTC price prediction using 90-day historical data disregards daily variance, resulting in inaccurate high-value predictions.



# Bitcoin Price Prediction Model

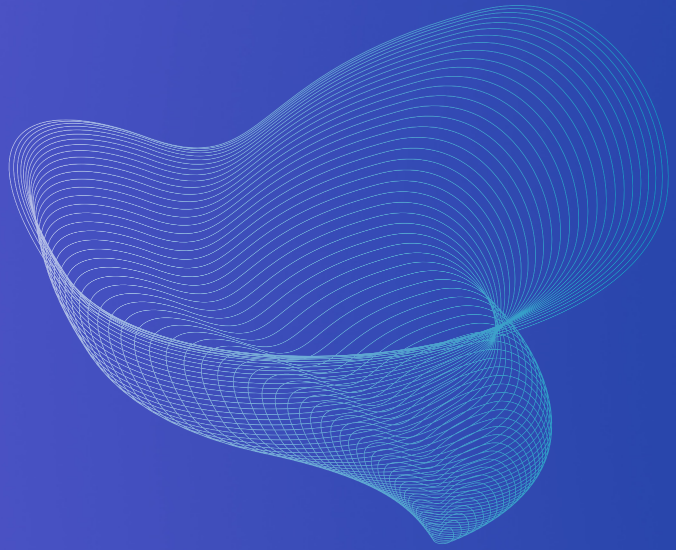
**Part 1:** Calculation of Metric Changes and Confidence Intervals. The code calculates the average percentage change for metrics (FGI, Puell Multiple, Mayer Multiple) and the price. It then determines confidence intervals using standard errors, degrees of freedom, and critical values.

**Part 2:** Estimation of Next Day's Price Change The code estimates the next day's price change based on weighted adjustments to the metric changes. It considers the previous day's price, adjusts the metric changes for target values, and applies a price change percentage.

**Part 3:** Estimation of Daily Variance and Printing Results The code calculates the high and low values for the estimated next day's price based on confidence intervals. It determines the estimated daily variance by taking the difference between the high and low values. The results are then printed.

# Bitcoin Price Prediction Model

```
# Ask the user to choose the confidence level
confidence_level = float(input("Choose the confidence level (0.80, 0.90, or 0.95): "))

# Set the desired confidence level
price_change_percentage = Data['Change %'].mean()

# Calculate the average percentage change for each metric
metrics_change_percentage = Data[['FGI', 'Puell Multiple', 'Mayer Multiple']].pct_change().mean()

# Calculate the confidence intervals for the average metric changes
std_errors = Data[['FGI', 'Puell Multiple', 'Mayer Multiple']].pct_change().sem()
df = len(Data) - 1  # Degrees of freedom
margins_of_error = st.t.ppf((1 + confidence_level) / 2, df) * std_errors
lower_bounds = metrics_change_percentage - margins_of_error
upper_bounds = metrics_change_percentage + margins_of_error

# Estimate the next day's price based on the previous day's price and the estimated price change
previous_day_price = Data['Price'].iloc[-1]
estimated_price_change = (
    (metrics_change_percentage['FGI'] + (54 - Data['FGI'].iloc[-1]) / Data['FGI'].iloc[-1]) * (price_change_percentage/
    (metrics_change_percentage['Puell Multiple'] + (0.99 - Data['Puell Multiple'].iloc[-1]) / Data['Puell Multiple'].il
    (metrics_change_percentage['Mayer Multiple'] + (1.23 - Data['Mayer Multiple'].iloc[-1]) / Data['Mayer Multiple'].il
)

estimated_next_day_price = previous_day_price * (1 + estimated_price_change)

# Calculate the high and low values of the estimated next day's price based on the confidence intervals
lower_bound_price = previous_day_price * (1 + lower_bounds.mean() * (price_change_percentage/100))
upper_bound_price = previous_day_price * (1 + upper_bounds.mean() * (price_change_percentage/100))

# Estimate the daily variance based on the high and low values of the estimated next day's price
estimated_daily_variance_high = upper_bound_price - lower_bound_price
estimated_daily_variance_low = lower_bound_price - upper_bound_price

# Print the estimated next day's price and estimated daily variance
print("Estimated Next Day's Price:", estimated_next_day_price)
print("Estimated Daily Variance (High):", estimated_daily_variance_high)
print("Estimated Daily Variance (Low):", estimated_daily_variance_low)
```



# The AI oracle that deciphers crypto's future whispers.

Cassandra Metrics

**PHONE**

(123) 456-7890

**EMAIL**

Cassandra@reallygreatsite.com