CS 210 Project Phase 2

I propose exploring the complex relationship between Bitcoin prices, the Fear & Greed Index (FGI), and many technical indicators for this data research project. The aim of this project's reasoning is understanding the intricate dynamics of the cryptocurrency market, and objective measurements interact to drive price fluctuations. In order to shed light on market behavior and enable price predictions, we want to combine datasets on Bitcoin prices, FGI, and technical indicators such as exponential moving averages (EMA_15) to explore underlying patterns and correlations.

Bitcoin has gained popularity as an alternative investment vehicle for institutional and individual investors looking to diversify their portfolios and hedge against more conventional financial products like stocks. The Fear and Greed Index is a sentiment indicator used in financial markets to gauge investors' emotions and market sentiment.

In conclusion, this study looks at the connections between Bitcoin prices, investor sentiment, and technical indications in an effort to better understand the complex structure of the cryptocurrency market. I hope to improve our comprehension of market dynamics and open the door for better informed trading and investment decisions in the cryptocurrency space by utilizing the power of data-driven analysis. I think there is a high correlation between them and I believe that future price prediction can be made using this data and machine learning methods.

Data Description:

- **Bitcoin Price**: The dataset includes the prices of Bitcoin over time, spanning from February 4, 2019, to February 2, 2024.

- **Fear and Greed Index Value**: Similarly, the dataset contains values of the Fear and Greed Index, which represents market sentiment towards Bitcoin.

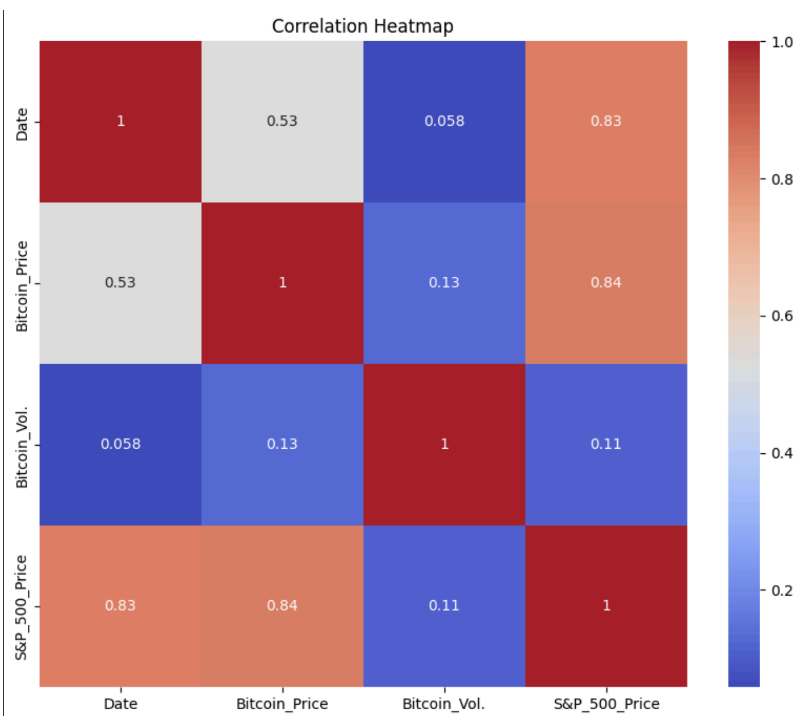
- **S&P 500 Price:** The dataset includes the prices of S&P 500 over time.

Firstly, I examined all of them by calculating statistics such as the mean, standard deviation and ranges to understand. The csv file containing the bitcoin price did not have weekend days because it corresponded to the stock market days, but the csv file containing the fear & greed index also includes weekends. For this reason, the number of columns was not equal, I fixed this by taking common dates. After that I sorted by date, and set the correct types of the columns like I converted date column type to datetime. After that, I merged them to view them as a whole. Also, I formed histograms of all of them to visualize them.

Bitcoin Dataframe and Correlation Map of Bitcoin Dataframe:

Date	Bitcoin_Price	Bitcoin_Vol.	S&P_500_Price
2019-02-04	3462.8	503920.0	2724.87
2019-02-05	3468.4	460950.0	2737.70
2019-02-06	3404.3	514210.0	2731.61
2019-02-07	3397.7	471360.0	2706.05
2019-02-08	3661.7	699230.0	2707.88
...	...	...	...
2024-01-29	43299.8	45230.0	4927.93
2024-01-30	42946.2	55130.0	4924.97
2024-01-31	42580.5	56480.0	4848.87
2024-02-01	43081.4	47690.0	4906.19
2024-02-02	43194.7	42650.0	4958.61

Correlation map of Bitcoin dataframe:



Dataframe created by merging Bitcoin Df and Fear & Greed Index Df:

Date	Bitcoin_Price	Bitcoin_Vol.	S&P_500_Price	fear_greed
2019-02-04	3462.8	503920.0	2724.87	27
2019-02-05	3468.4	460950.0	2737.70	21
2019-02-06	3404.3	514210.0	2731.61	14
2019-02-07	3397.7	471360.0	2706.05	18
2019-02-08	3661.7	699230.0	2707.88	37
...	...	...	...	...
2024-01-29	43299.8	45230.0	4927.93	55
2024-01-30	42946.2	55130.0	4924.97	61
2024-01-31	42580.5	56480.0	4848.87	60
2024-02-01	43081.4	47690.0	4906.19	63
2024-02-02	43194.7	42650.0	4958.61	63

Histogram of Fear and Greed Index:

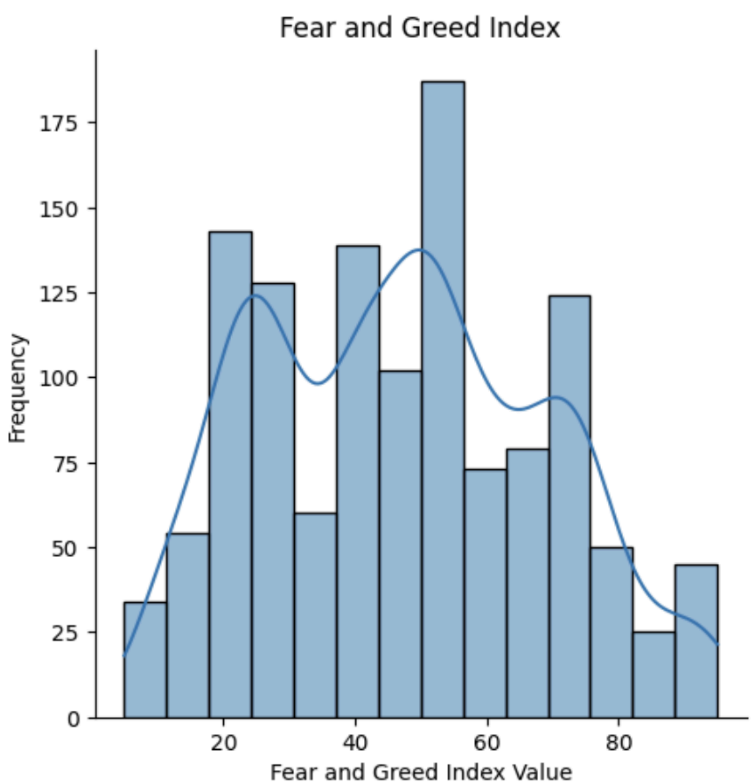
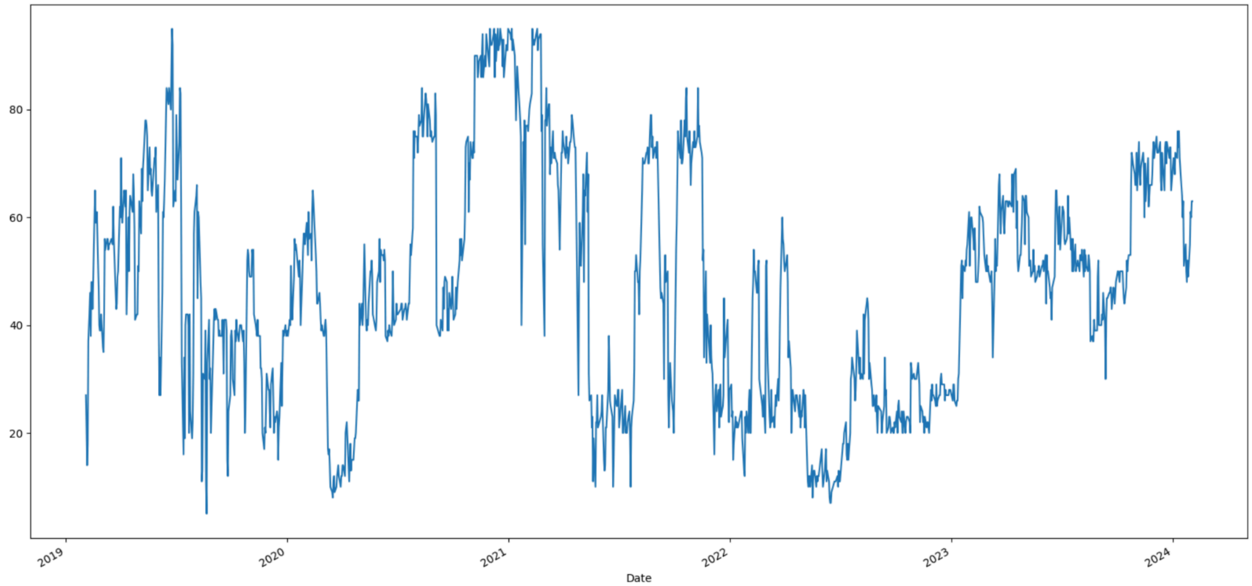
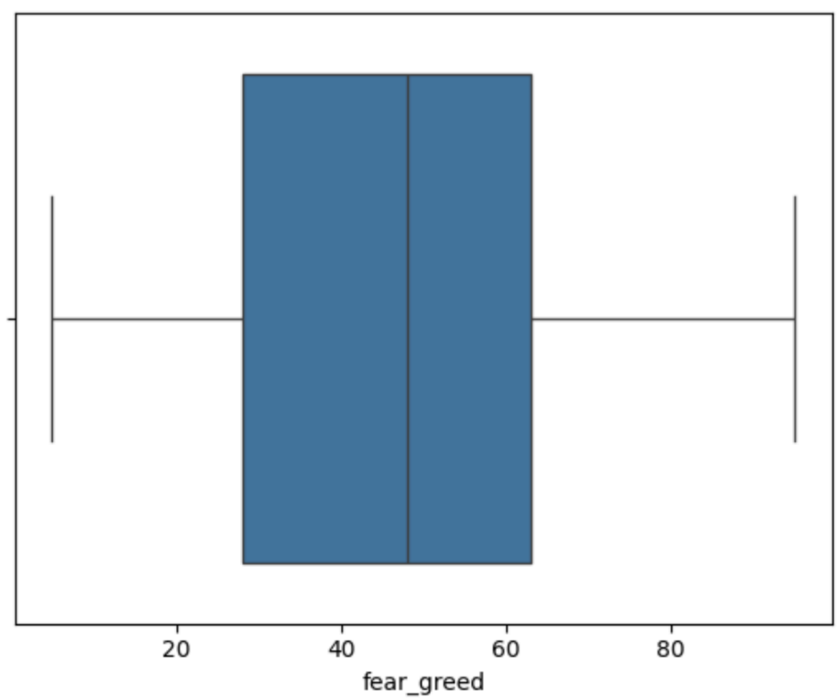


Table showing the change of Fear & Greed Index in the date range



The boxplot of Fear & Greed Index



Histogram of Bitcoin Price:

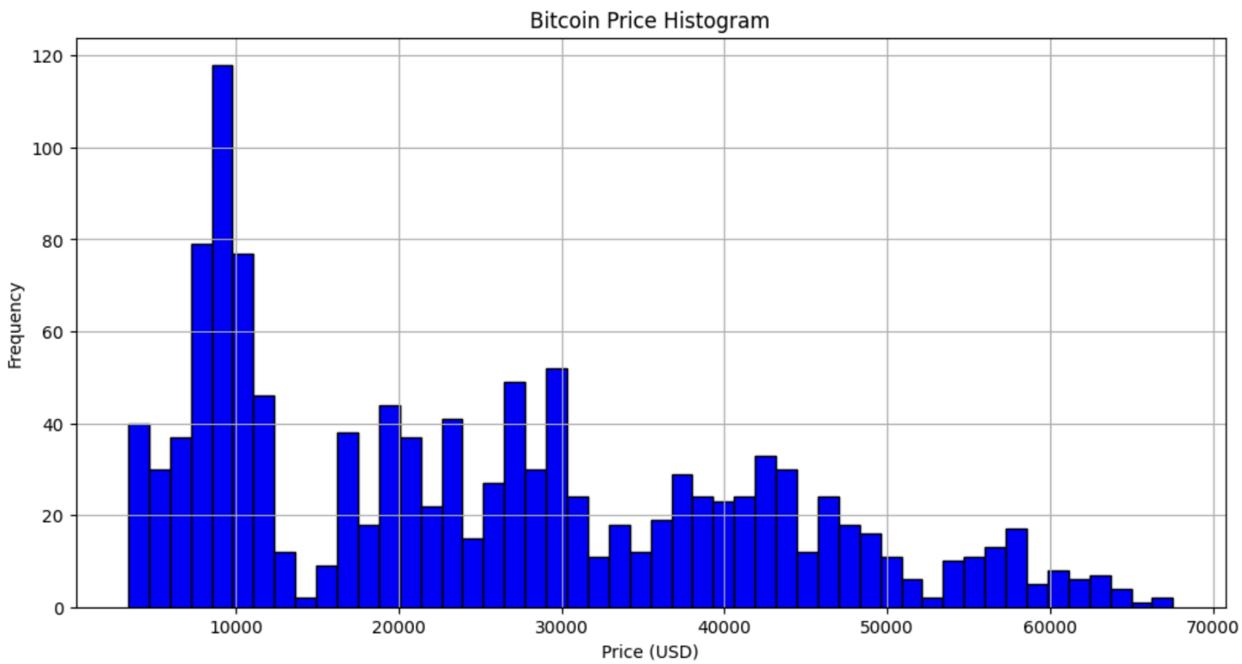
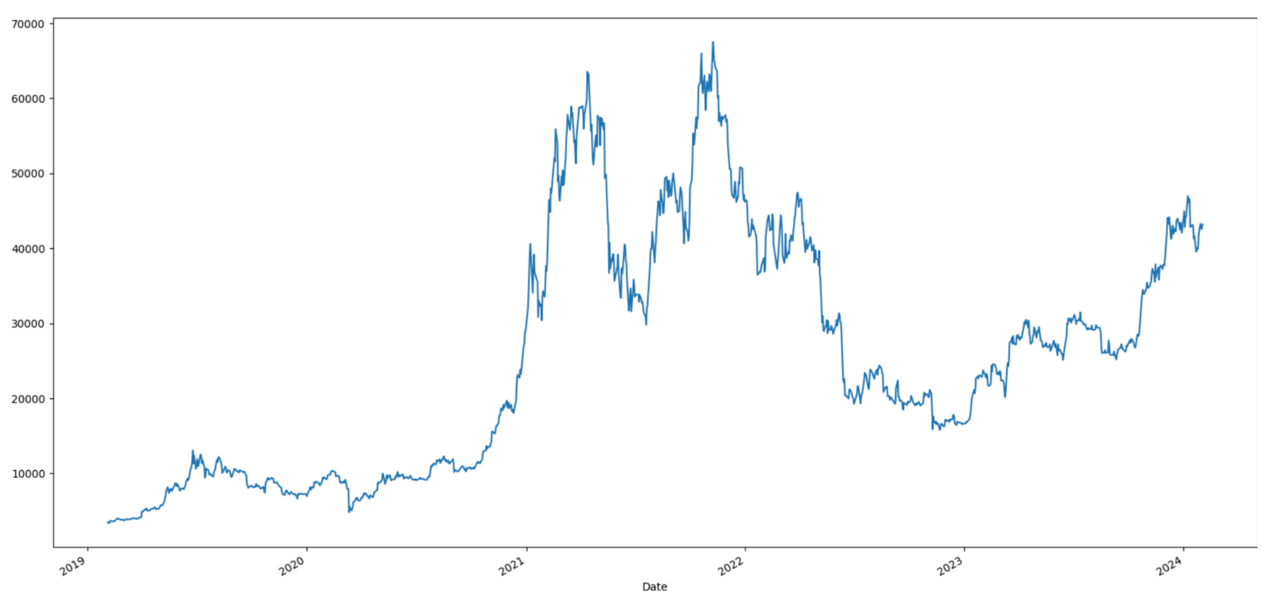
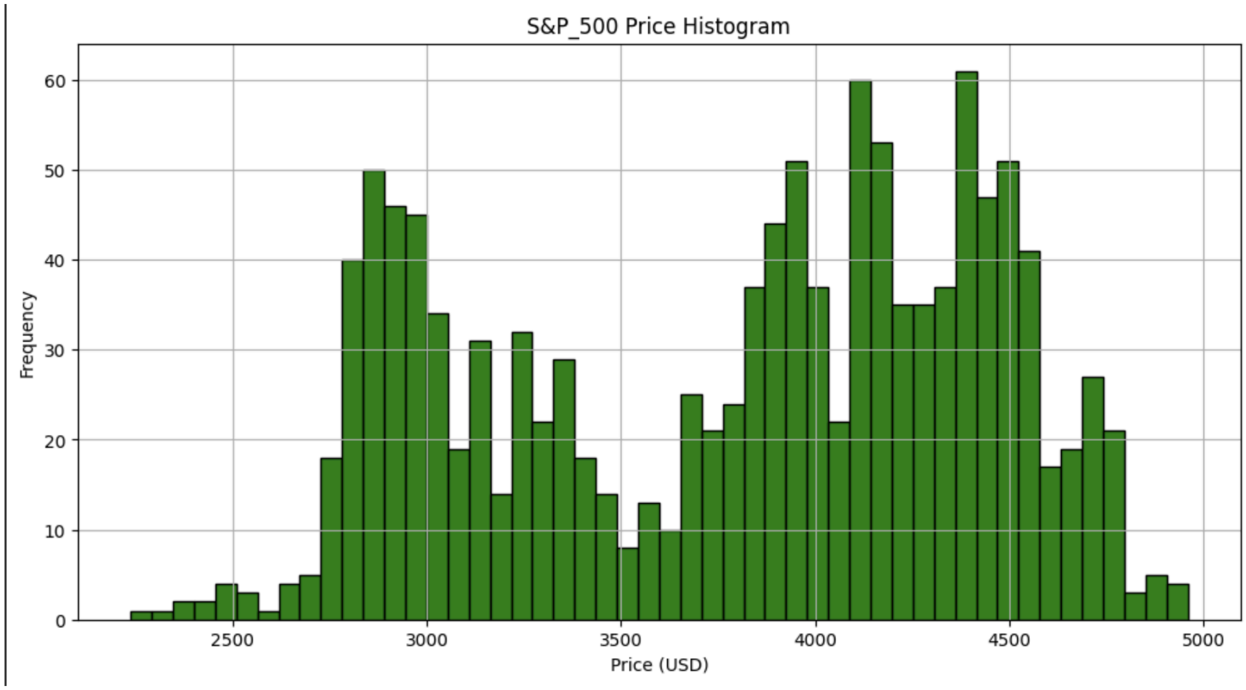


Table showing the change of Bitcoin Price in the date range

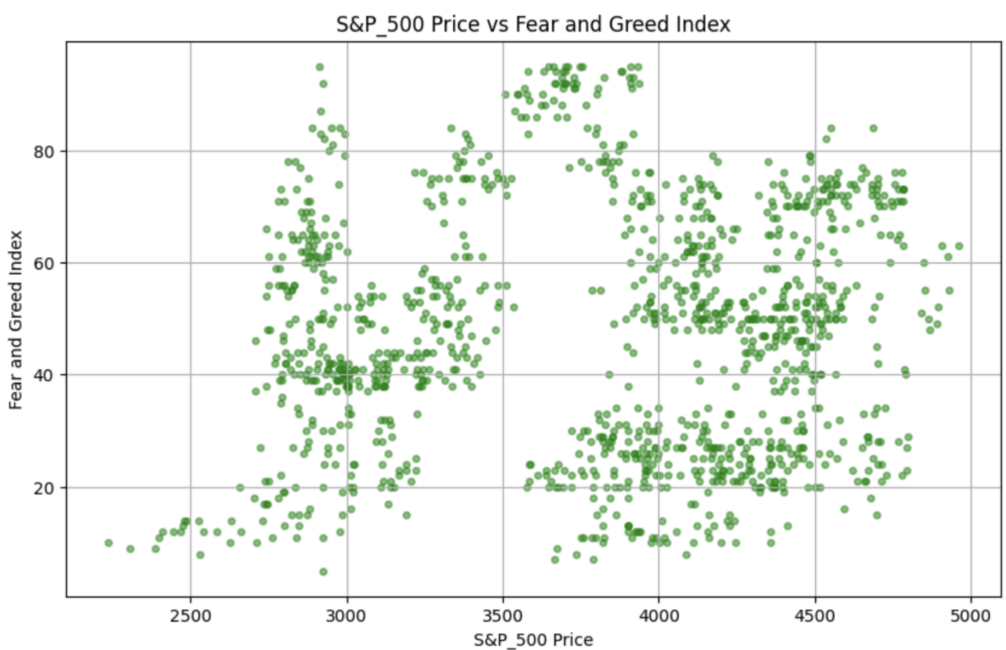


Histogram of S&P 500 Price:

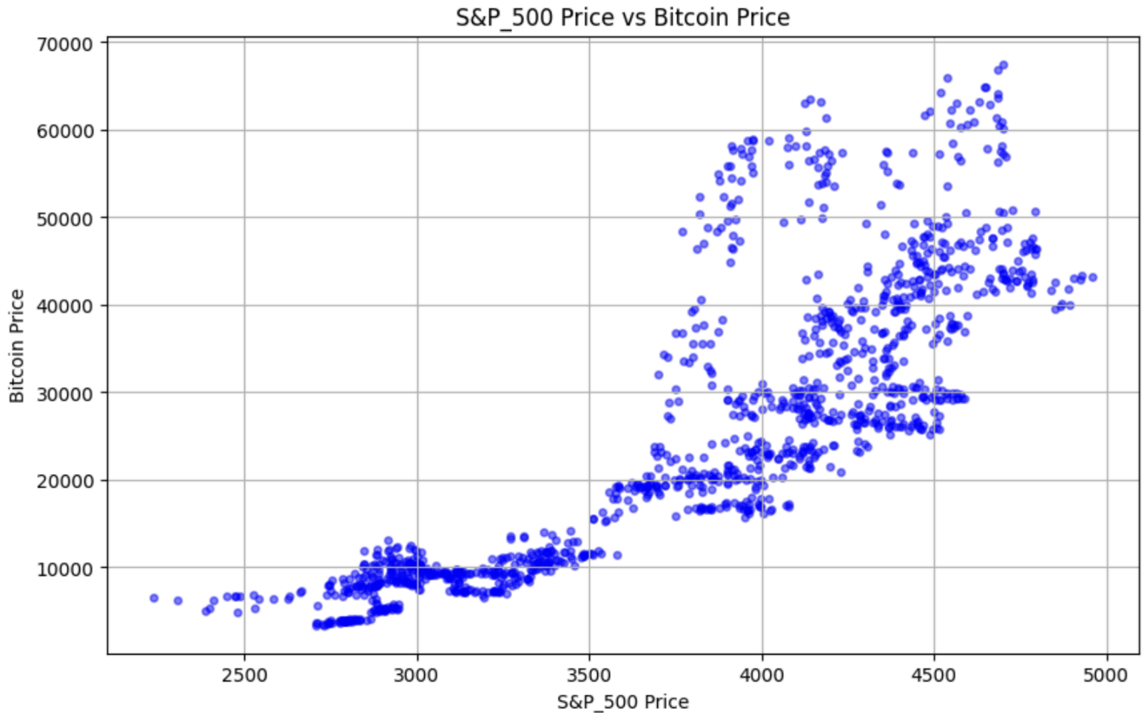


After forming histograms, I want to find correlation between these data.

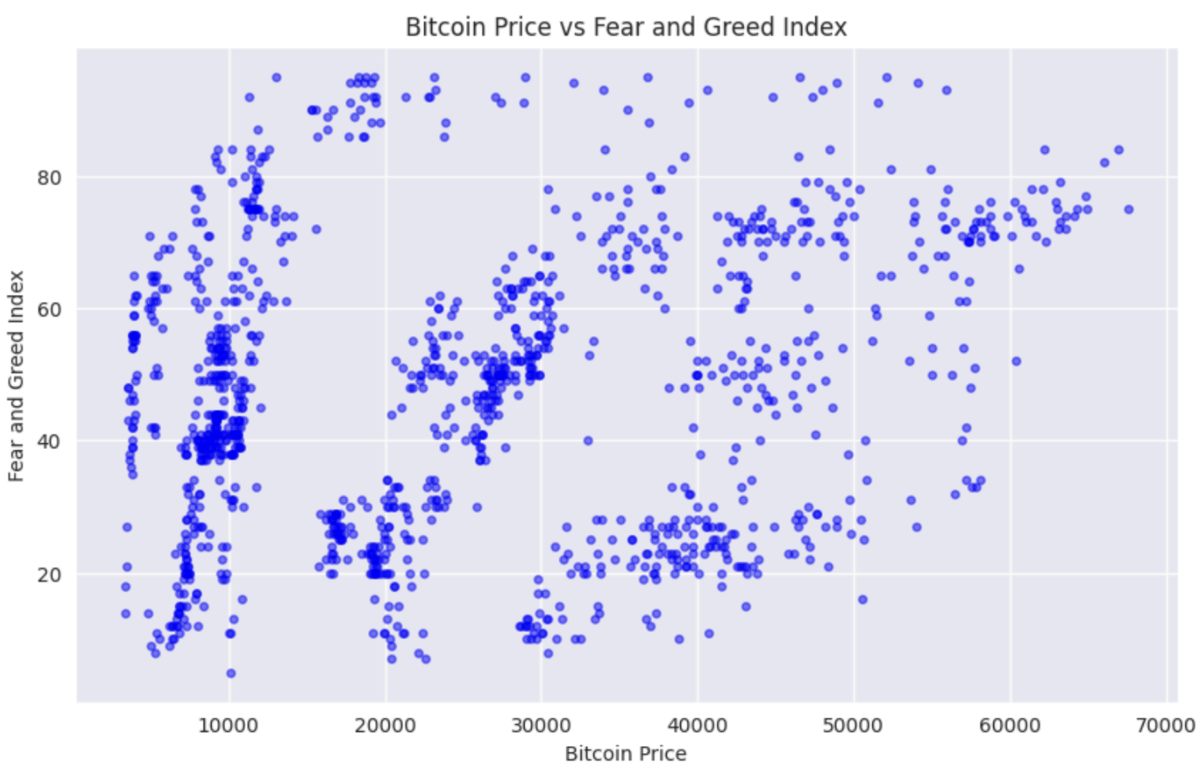
Scatter plot of S&P 500 Price and Fear & Greed Index



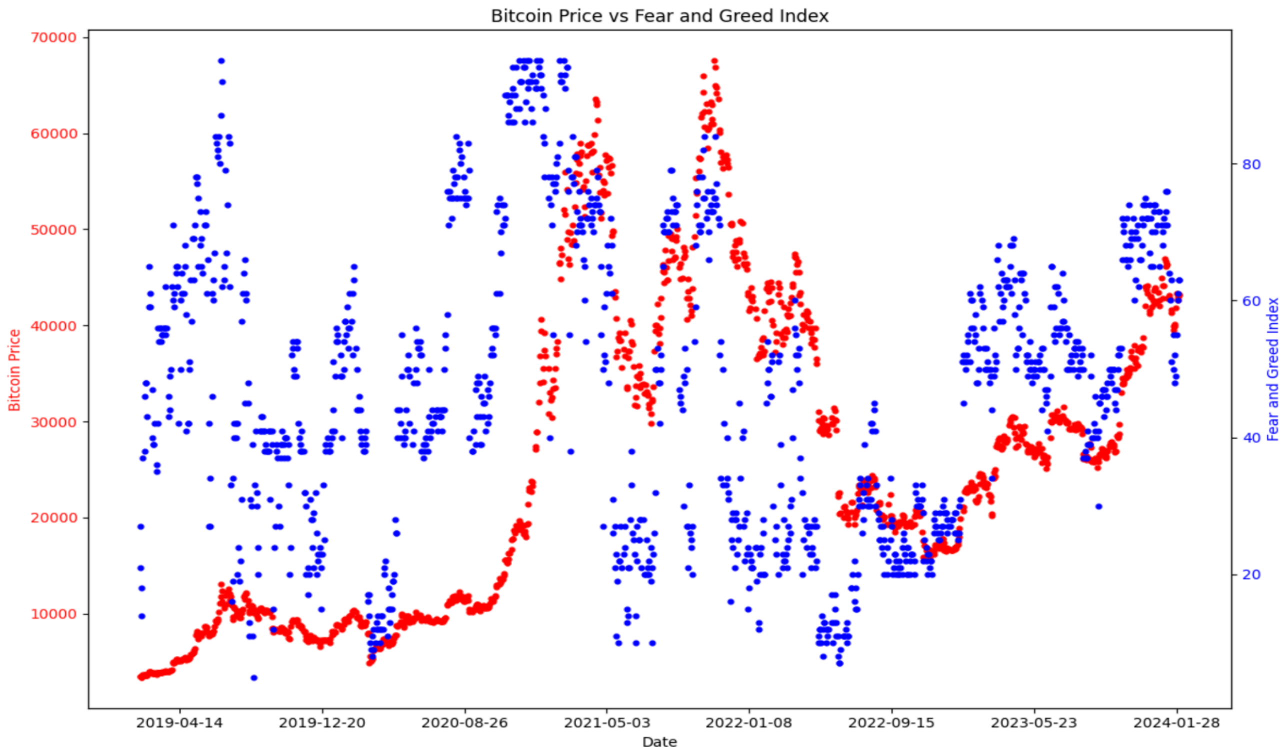
Scatter plot of Bitcoin Price and S&P 500 Price



Scatter plot of Bitcoin Price and Fear & Greed Index



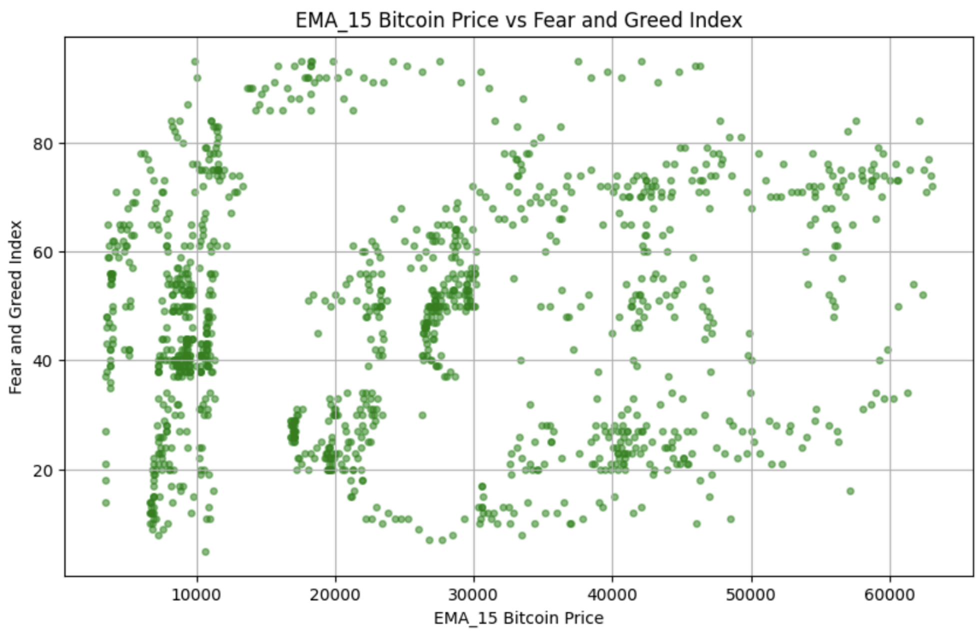
In addition, I created a plot to compare Bitcoin Price and Fear & Greed index over the time frame.



After forming scatterplots, I want to analyze the correlation between Bitcoin Price and Fear & Greed index. Firstly, I did not find any logical correlation between Bitcoin Price and Fear & Greed index by using scatterplot, it is not observable. Also we see the S&P 500 Price and Fear & Greed index does not have a high correlation, it is hard to predict.

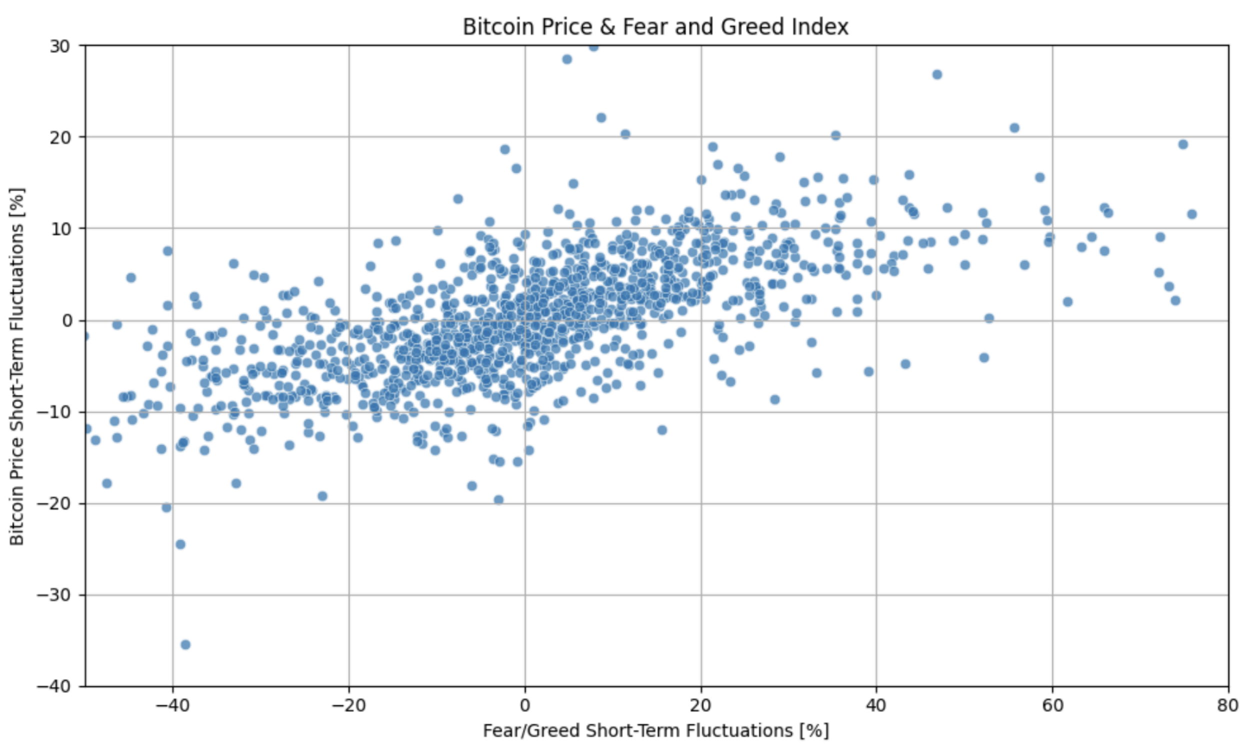
After my observation, I want to examine more different techniques so I added a new column for the 15-day exponential moving average (EMA) to my DataFrame. Also, there is no logical correlation between EMA15 of Bitcoin Price and Fear & Greed index.

Scatter plot of EMA15 of Bitcoin Price and Fear & Greed Index



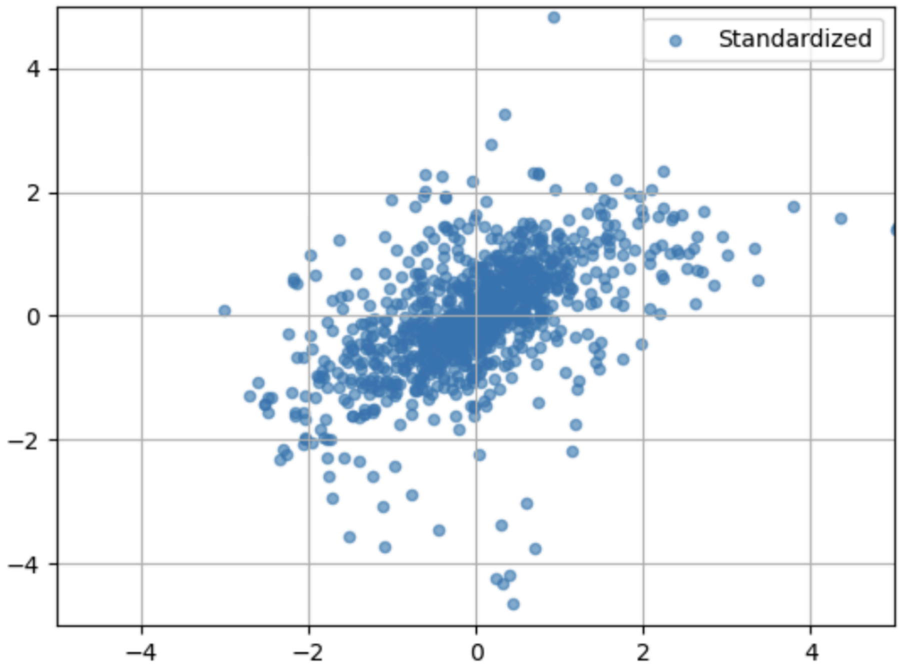
After that, I decided to look at short term fluctuations of them, and used a short_term function to remove high-frequency components. After that I computed short term fluctuations and I formed scatter plots of it.

Scatter plot of Short term fluctuations of Bitcoin Price and Fear & Greed Index



When I look at this scatter plot, it seems to be an observable positive correlation. After that, I used a standardized function to look at a standardized version of the scatter plot. As a result of analyzing data and relations between variables I can conclude our Hypothesis and test it

Standardized version of the scatter plot



Formulated Hypothesis:

Null Hypothesis (H0): There is no significant correlation between short-term fluctuations in Bitcoin prices and short-term fluctuations in the Fear and Greed Index.

Alternative Hypothesis (H1): There is a significant correlation between short-term fluctuations in Bitcoin prices and changes in the Fear and Greed Index.

Result: Null hypothesis (H0) is rejected. There is a statistically significant linear relationship between short-term fluctuations in Bitcoin prices and changes in the Fear and Greed Index.

```
import statsmodels.api as sm
from scipy import stats

# Calculate Pearson correlation coefficient and p-value
correlation, p_value = stats.pearsonr(short_term_fluctuations_df['Bitcoin_Price_Short'], short_term_fluctuations_df['FearGreed_Short'])

# Set significance level (alpha)
alpha = 0.05

# Perform hypothesis test
if p_value < alpha:
    print("Reject null hypothesis (H0): There is a statistically significant linear relationship between short term fluctuations of Bitcoin_Price and  short term fluctuations o
else:
    print("Fail to reject null hypothesis (H0): Insufficient evidence for a statistically significant linear relationship.")
```

```
Reject null hypothesis (H0): There is a statistically significant linear relationship between short term fluctuations of Bitcoin_Price and  short term fluctuations of Fear&Greed I
```

Hypothesis Testing:

To test this hypothesis, I standardized them as I showed its plot and I performed a correlation analysis between short-term fluctuations in Bitcoin prices and short-term fluctuations in the Fear and Greed Index. I used a statistical test which is Pearson's correlation coefficient. I computed the correlation coefficient between these two variables. I set a significance level (alpha), typically 0.05. The hypothesis test supported the alternative hypothesis, so I proceeded with building a regression model to further analyze and predict the relationship between Bitcoin prices and the Fear and Greed Index.

Value of correlation coefficient = [0.66794596]
Value of interception = -0.0036941

The correlation coefficient is approximately 0.6679, indicating a moderately positive linear relationship between the variables. When the value variable is set to zero, the intercept indicates the expected value of Bitcoin_Price_Short. It provides us with the model's regression line.

Building the Regression Model:

Feature Selection: In this case, the Fear and Greed Index will likely be my independent variable, and Bitcoin prices will be my dependent variable.

Splitting the Data: I divided my dataset into training and testing sets to evaluate the performance of the regression model. A common split is 80% for training and 20% for testing.

Model Training: I fit a linear regression model to the training data. I used sklearn.linear_model in Python to do this easily.

Model Evaluation: After the model is trained, I evaluate its performance using the testing data. I calculated metrics R-squared and RMSE to assess how changes in the Fear & Greed Index are associated with changes in the short-term fluctuations of Bitcoin price,

Regression Equation: Bitcoin Price = -0.0036941 + (0.66794596) × Fear & Greed Index

R-squared value (R^2): 0.44
With an R-squared (R^2) value of 0.44, the independent variable(s) in the regression model may account for 44% of the variability in the dependent variable. Put differently, the model can explain roughly 44% of the variability in the data points, leaving the remaining 56% unexplained and potentially attributable to measurement error, random variation, or other factors not covered by the model. While a lower R-squared value implies that the model may not be capturing much of the variability in the data, a higher R-squared value shows that the regression model fits the data better.

MSE: 0.4943704255065399

The squared difference between the real and projected Bitcoin_Price_Short values is, on average, about 0.494, according to the MSE of 0.494. Although they have the same issue with the R^2 value, lower values are preferable.

```
# extracting the input and output vectors
X = short_term_fluctuations_df['FearGreed_Short'].values.reshape(-1,1)  # converting to column vector
y = short_term_fluctuations_df['Bitcoin_Price_Short'].values

# train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)
```

```
# creating the model
model = LinearRegression()
model.fit(X_train, y_train)

# intercept
b = model.intercept_
# slope
m = model.coef_

print(m, b, sep="\n")

# Evaluate the model
r2_score = model.score(X, y)
print(f"R-squared value: {r2_score}")
# Print model coefficients
```

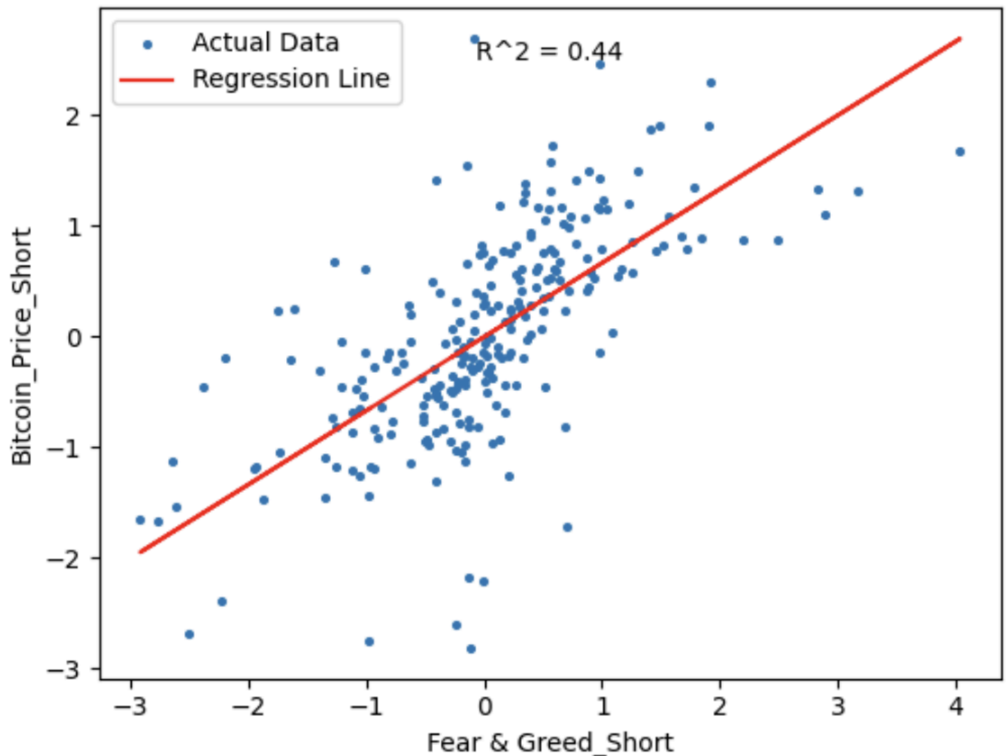
```
[0.20218126]
0.05453618958333882
R-squared value: 0.4373436634955137
```

Interpretation: I analyzed the coefficients of the regression model to understand the relationship between Fear and Greed Index values and Bitcoin prices. There is a positive coefficient indicating a positive relationship.

Visualization:

I visualized the relationship between Fear and Greed Index values and Bitcoin prices using scatter plots and regression lines.

Graph of model for Test Data:



Graph of regression model for all data: