GOLD PRICE REGRESSION ANALYSIS

BY JAGRUTHI MANCHI SARAF

DECEMBER 23, 2024

Abstract

This project investigates the volumes of gold alongside other key commodities such as palladium, platinum, silver, and oil. Additionally, correlations between these predictors and economic factors like EUR/USD and S&P500 volumes are explored. The aim is to analyze trends and relationships between these variables and provide actionable insights that assist in understanding market behaviors. Through data preparation, visualization, and analysis, key insights into commodity trading volumes and their predictors are extracted.

Introduction

Gold is not only a commodity but also a key economic indicator that reflects market sentiments during various economic scenarios. For example, in times of financial crises, investors often flock to gold as a safe haven, resulting in increased trading volumes. Conversely, during periods of economic growth, gold volumes may stabilize or decline, as riskier assets like equities become more attractive. By studying gold's interrelation with other metals and economic predictors, such as foreign exchange rates, we gain critical insights into its multifaceted role in the global economy. For instance, during periods of economic instability, gold volumes typically rise as investors seek a safe haven, demonstrating its role as a barometer of financial market sentiment. Understanding the interplay between gold and these factors can yield insights into market dynamics. This project focuses on comparing gold volume with related commodities and economic variables to determine trends and derive actionable insights.

Objectives:

1. Compare the yearly trends in gold volume with other commodities, examining potential seasonal variations and correlations between price and volume.
2. Investigate how economic indicators like EUR/USD influence gold and related commodity volumes.
3. Develop an understanding of market trends and interdependencies among variables.

Scope:

This project spans the analysis of multiple commodities, including palladium, platinum, silver, and oil, with an emphasis on understanding their interactions with gold volume. It examines regional and global market dynamics, highlighting trends such as seasonality, cross-commodity influences, and price-volume correlations. Additionally, the scope delves into the economic predictors like foreign exchange rates (e.g., EUR/USD) and indices (S&P500, Nasdaq), exploring their impact on commodity volumes to discern comprehensive market patterns and interactions. Additional predictors include foreign exchange rates and indices like S&P500 and Nasdaq volumes.

Why Gold Price Regression?

Gold has historically been viewed as a store of value, preserving wealth over long periods, especially during times of inflation, currency devaluation, or economic instability. Its limited supply and intrinsic value make it less susceptible to the fluctuations that impact fiat currencies and other assets. It’s seen as a hedge against uncertainties, providing stability when traditional assets (like stocks and bonds) are underperforming or at risk.
Central banks hold significant gold reserves as part of their foreign exchange reserves, reflecting the metal’s long-standing importance in the global financial system.

In real world applications, sometimes data will come in different granularities. In this dataset we can find daily, monthly and trimonthly data. Normalizing this inconsistencies and handling nan values should be one of the first challenges when dealing with this dataset.

Gold plays a pivotal role in global trade and economics. The analysis of its volume compared to other commodities offers:

Market Trends: Exploring significant patterns, such as seasonal variations, yearly trends, and price-volume correlations, to uncover consistent and emerging dynamics in the commodity markets.

Correlations: Exploring relationships with predictors.

Strategic Insights: Insights derived from the analysis can guide stakeholders in making informed decisions, such as identifying optimal trading periods, developing robust investment strategies, and implementing effective risk management practices. For instance, during market fluctuations, stakeholders can rely on historical correlations to better navigate the commodities market and enhance decision-making.

About the dataset

The data is a time series dataset with financial info for some market indices, commodities, economic indicators and forex rates. Market indices and commodities are represented via the respective exchange traded fund. It includes values from 2010 to 2024.

The dataset includes:

- **Time Range:** Daily data over several years, aggregated to yearly means.
- **Columns:** Commodities volumes such as gold volume, palladium volume, platinum volume, silver volume, and oil volume.
- **Predictors:** Economic indicators including eur_usd, s&p500 volume, and nasdaq volume.
- **Features:** Each column provides valuable insights into commodity or market trends.
- **Source:** Extracted from reliable financial databases to ensure accuracy.

challenges:

- Missing values and unclean data were addressed during the data cleaning phase.

Data Analysis:

I have used Jupyter notebook in vscode for my analysis, where i have first imported all the required libraries like pandas, numpy, matplotlib, seaborn and scipy.

Transition of Data

Before diving into the insights derived from the dataset, it is crucial to ensure the reliability and consistency of the data. Effective data cleaning lays the foundation for accurate and meaningful analysis by addressing discrepancies, handling missing values, and formatting inconsistencies. In this step, we aim to refine the raw dataset into a structured format, making it suitable for further exploration and visualization.

Before starting the data analysis of gold price regression, i have installed all the required libraries that i needed in this project.

Importing the required libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[1] ✓ 45.0s

Uploading the Data

Uploaded the data using the pandas library.

Uploading the dataset

```
file_path = open("C:/Users/msnchary/OneDrive/Desktop/anudip/Python_project/financial_regression.csv")
data = pd.read_csv(file_path)
```

✓ 0.4s

Data Cleaning:

Checking the shape of the data

DATA CLEANING: Checking the shape of the dataset

```
print(data.shape)
```

[3] ✓ 0.0s

```
(3904, 47)
```

My data has 3904 rows and 47 columns before data cleaning

Checking nulls

```
[5]
data.isnull().sum()
```

```
date              0
sp500 open      185
sp500 high      185
sp500 low       185
sp500 close     185
```

By using isnull keyword we can know how many blanks are there in each column.

After knowing that the below mentioned columns are empty that means there is no trading on that particular day, so i have deleted blanks in that particular columns.

```
# Columns to check for 185 nulls
columns_to_drop_nulls = [
    'sp500 open', 'sp500 high', 'sp500 low', 'sp500 close', 'sp500 volume', 'sp500 high-low',
    'nasdaq open', 'nasdaq high', 'nasdaq low', 'nasdaq close', 'nasdaq volume', 'nasdaq high-low',
    'silver open', 'silver high', 'silver low', 'silver close', 'silver volume', 'silver high-low',
    'oil open', 'gold high', 'gold low', 'gold close', 'gold volume'
]

# Drop rows where any of these columns have nulls
data_cleaned = data.dropna(subset=columns_to_drop_nulls)

# Replace nulls in specific columns with 0
columns_to_fill = ['GDP', 'us_rates_%','usd_chf','eur_usd']
data_cleaned[columns_to_fill] = data_cleaned[columns_to_fill].fillna(0)

# Verify the results
print(f"Original dataset shape: {data.shape}")
print(f"Cleaned dataset shape: {data_cleaned.shape}")
print("Remaining null values:\n", data_cleaned.isnull().sum())
```

```
Original dataset shape: (3904, 47)
Cleaned dataset shape: (3719, 47)
Remaining null values:
```

After the data cleaning my data has 3719 rows and 47 columns.

There is no change in columns as i have not deleted any columns.

```
data= data_cleaned
print(data.shape)
```

```
(3719, 47)
```

Data Processing

Printed the first 1o rows of my dataset.

```
data.head(10)
```

Python

	date	sp500 open	sp500 high	sp500 low	sp500 close	sp500 volume	sp500 high-low	nasdaq open	nasdaq high	nasdaq low	...	palladium high	palladium low	palladium close	palladium volume	palladium high-low	gold open
0	1/14/2010	114.49	115.14	114.42	114.93	115646960.0	0.72	46.26	46.520	46.22	...	45.020	43.86	44.84	364528.0	1.160	111.51
1	1/15/2010	114.73	114.84	113.20	113.64	212252769.0	1.64	46.46	46.550	45.65	...	45.760	44.40	45.76	442210.0	1.360	111.35
3	1/19/2010	113.62	115.13	113.59	115.06	138671890.0	1.54	45.96	46.640	45.95	...	47.080	45.70	46.94	629150.0	1.380	110.95
4	1/20/2010	114.28	114.45	112.98	113.89	216330645.0	1.47	46.27	46.604	45.43	...	47.310	45.17	47.05	643198.0	2.140	109.97
5	1/21/2010	113.92	114.27	111.56	111.70	344747028.0	2.71	46.06	46.350	45.30	...	46.980	45.07	45.30	388457.0	1.910	108.48
6	1/22/2010	111.20	111.74	109.09	109.21	345627282.0	2.65	45.34	45.480	44.04	...	44.386	42.60	43.30	343595.0	1.786	106.93
7	1/25/2010	110.21	110.41	109.41	109.77	186751367.0	1.00	44.39	44.600	44.12	...	44.420	43.43	44.36	184431.0	0.990	107.44
8	1/26/2010	109.34	110.47	109.04	109.31	210788669.0	1.43	44.28	44.890	44.05	...	43.100	42.38	42.75	338167.0	0.720	106.87
9	1/27/2010	109.17	110.08	108.33	109.83	271482263.0	1.75	44.29	44.850	44.01	...	42.440	40.93	41.34	419800.0	1.510	107.53
10	1/28/2010	110.19	110.25	107.91	108.57	315523641.0	2.34	44.40	44.430	43.32	...	42.630	40.85	42.16	281792.0	1.780	107.18

Data Analysis

Comparing commodities with Gold price

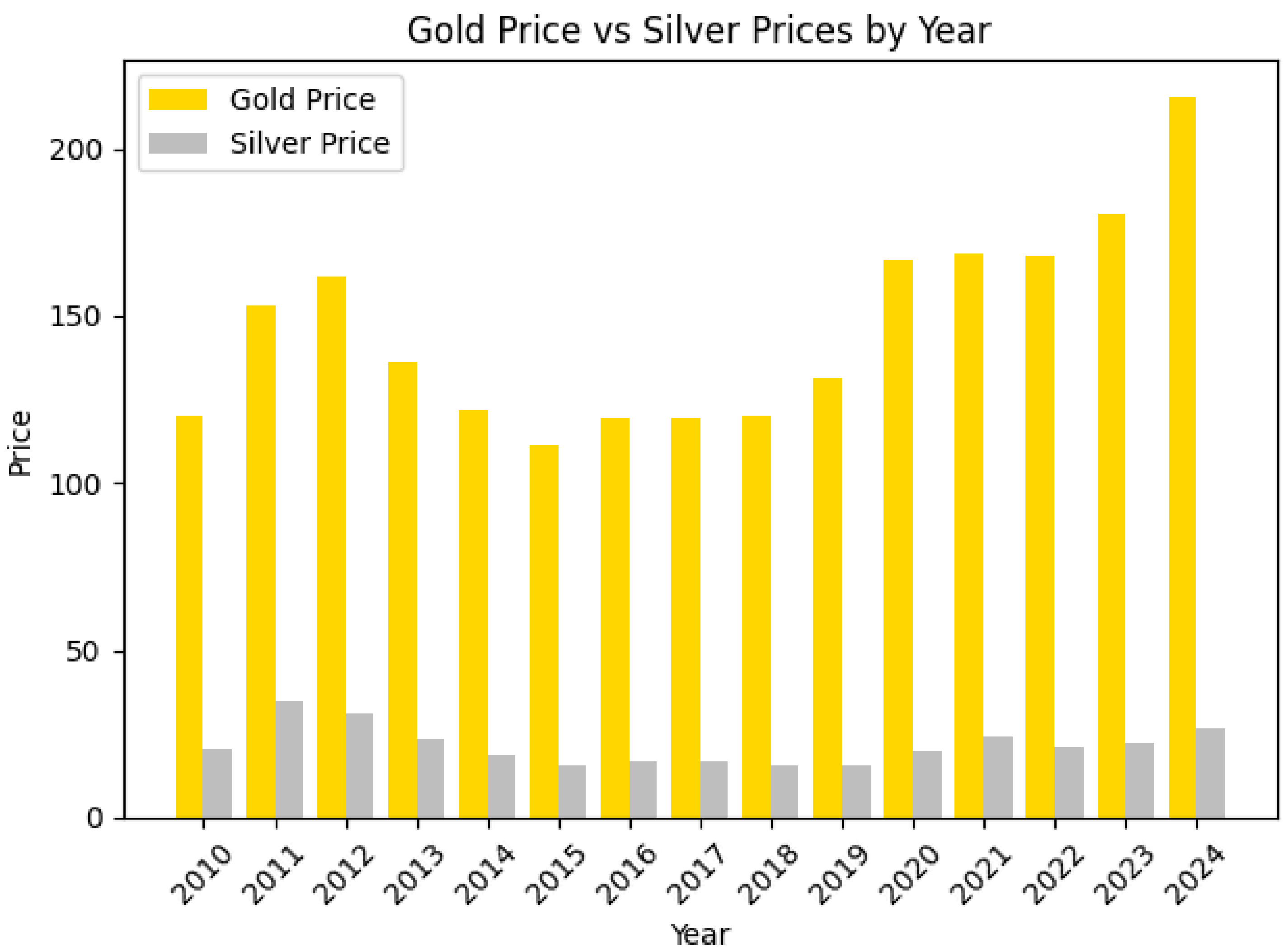
My dataset has different commodities such as gold, silver, palladium, platinum and oil. compared these commodities closing price for every year with gold price.

DATA ANALYSIS: Comparing the commodities with each other to find out the trend.

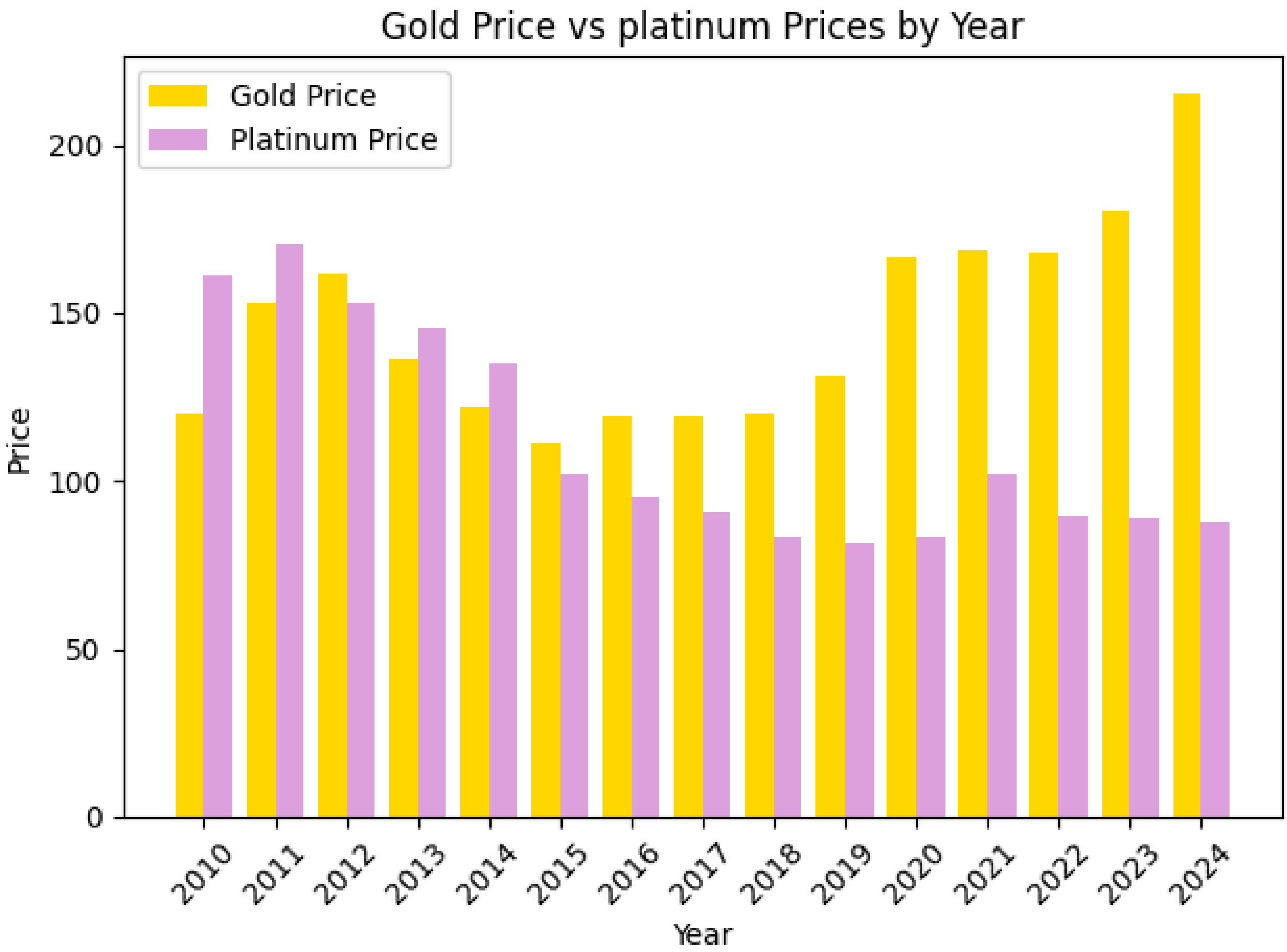
Comparing the gold with silver prices

```python
#ColumnChart of gold with silver prices
#graph1
data['year'] = pd.to_datetime(data['date']).dt.year

# Group data by year and calculate the mean for 'gold close' and 'silver close'
yearly_data = data.groupby('year')[['gold close', 'silver close']].mean().reset_index()

# Column Chart: Gold vs Silver Prices by Year
years = np.arange(len(yearly_data['year']))
width = 0.4

plt.bar(years - width / 2, yearly_data['gold close'], width, label='Gold Price', color='gold')
plt.bar(years + width / 2, yearly_data['silver close'], width, label='Silver Price', color='silver')

plt.title('Gold Price vs Silver Prices by Year')
plt.xlabel('Year')
plt.ylabel('Price')
plt.xticks(years, yearly_data['year'], rotation=45)
plt.legend()
plt.tight_layout()
plt.show()
```

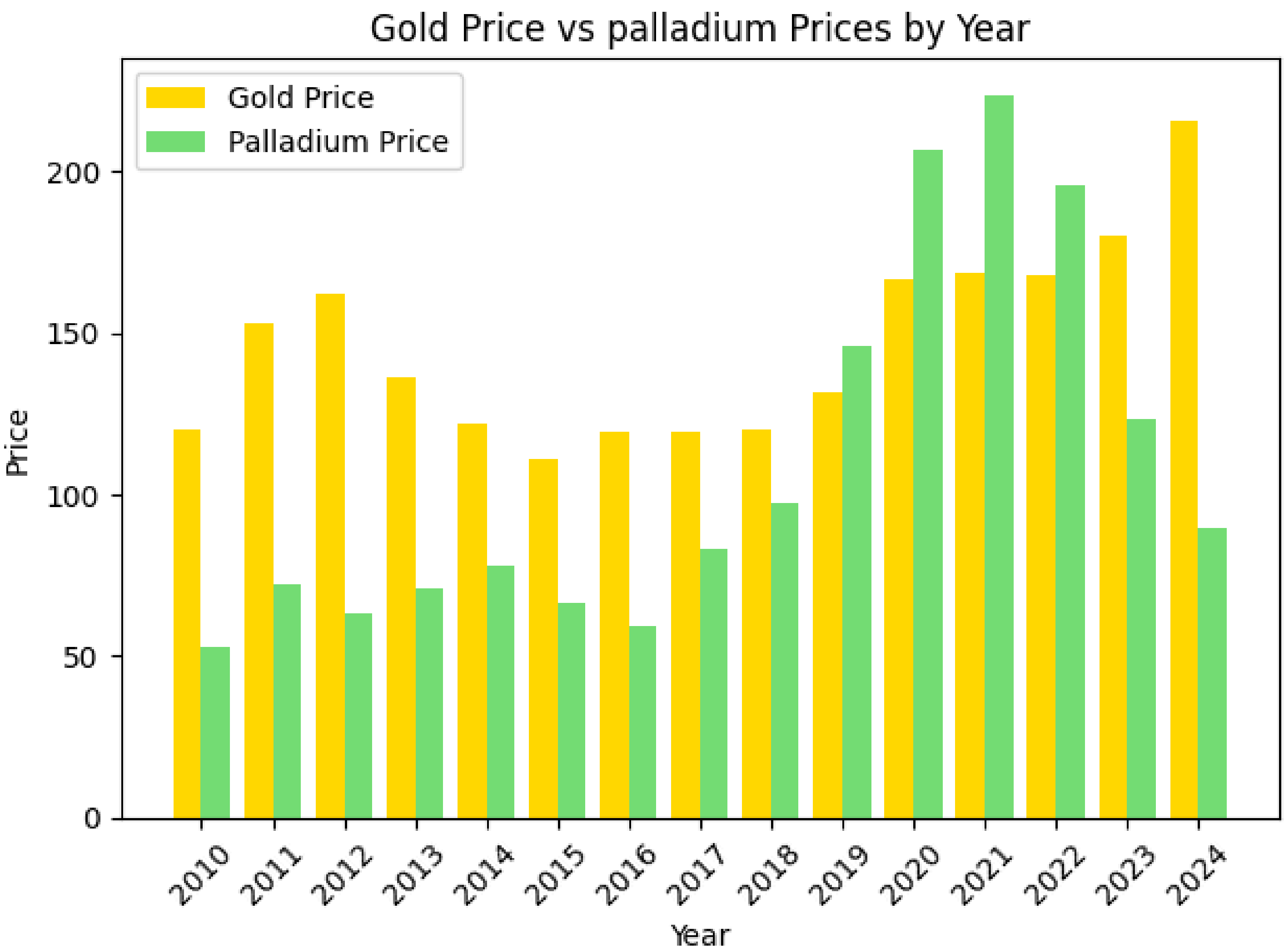
Graph1: Comparing the gold with silver prices.



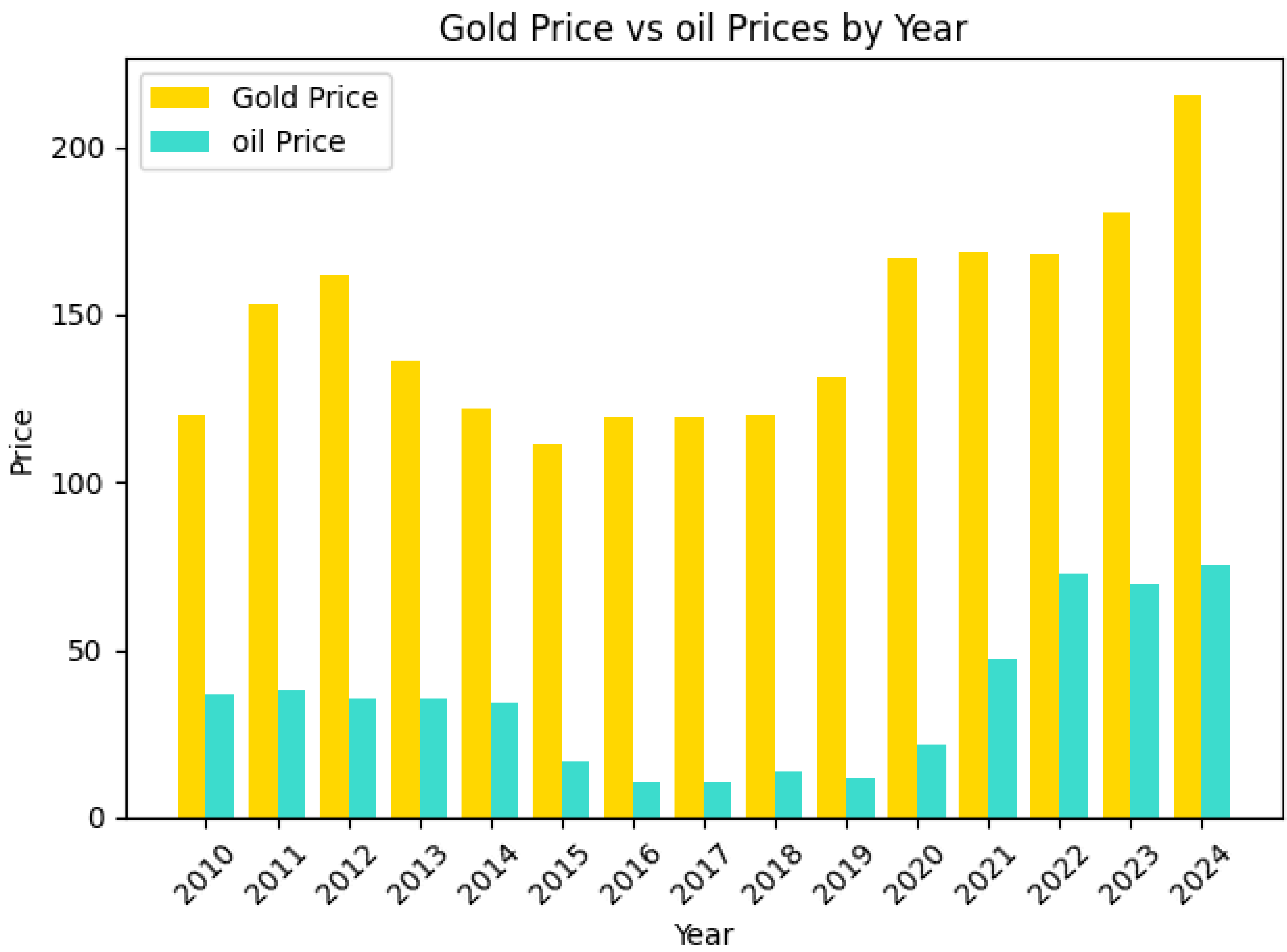
Graph2: Comparing the gold with platinum



Graph3: Comparing gold with Palladium Prices



Graph4: Comparing gold with Oil prices



Time series plot

Comparing gold price with S&P 500 and NASDAQ

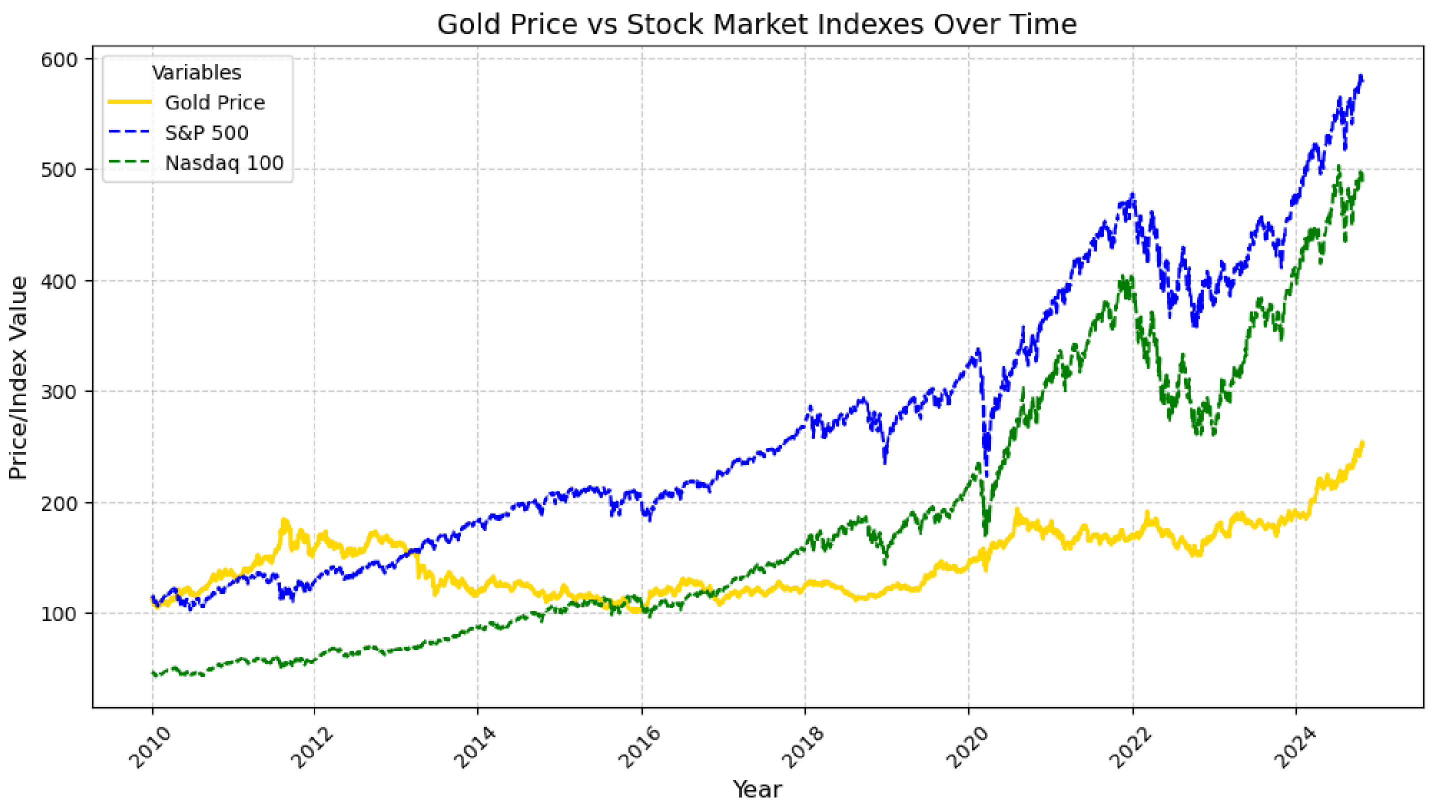
What is S&P 500 and Nasdaq?

The S&P 500 is a stock market index that tracks the performance of 500 large, publicly traded companies in the United States across various sectors. It is widely regarded as a benchmark for overall U.S. stock market health and economic trends.

The Nasdaq is a global electronic marketplace for trading securities and is also home to the Nasdaq Composite Index, which tracks over 3,000 technology and growth-oriented companies. It is known for its heavy weighting in tech giants like Apple, Amazon, and Microsoft.

```
#graph5
data['date'] = pd.to_datetime(data['date'])
# Extract the year from the 'date' column
data['year'] = data['date'].dt.year
# Time Series Plot: Gold Price vs S&P 500 and Nasdaq 100
plt.figure(figsize=(12, 6))
plt.plot(data['date'], data['gold close'], label='Gold Price', color='gold', linewidth=2)
plt.plot(data['date'], data['sp500 close'], label='S&P 500', color='blue', linestyle='--')
plt.plot(data['date'], data['nasdaq close'], label='Nasdaq 100', color='green', linestyle='--')
# Add title and labels
plt.title('Gold Price vs Stock Market Indexes Over Time', fontsize=14)
plt.xlabel('Year', fontsize=12)
plt.ylabel('Price/Index Value', fontsize=12)
# Customize x-axis to show alternate years
years = data['year'].unique()
alternate_years = [year for i, year in enumerate(years) if i % 2 == 0]
plt.xticks(ticks=[data[data['year'] == year]['date'].iloc[0] for year in alternate_years],
           labels=alternate_years, rotation=45)

# Add legend and grid
plt.legend(title="Variables", loc="upper left")
plt.grid(visible=True, linestyle='--', alpha=0.6)
plt.show()
```

Graph5: Timeseries Gold price with Stock market Indexes



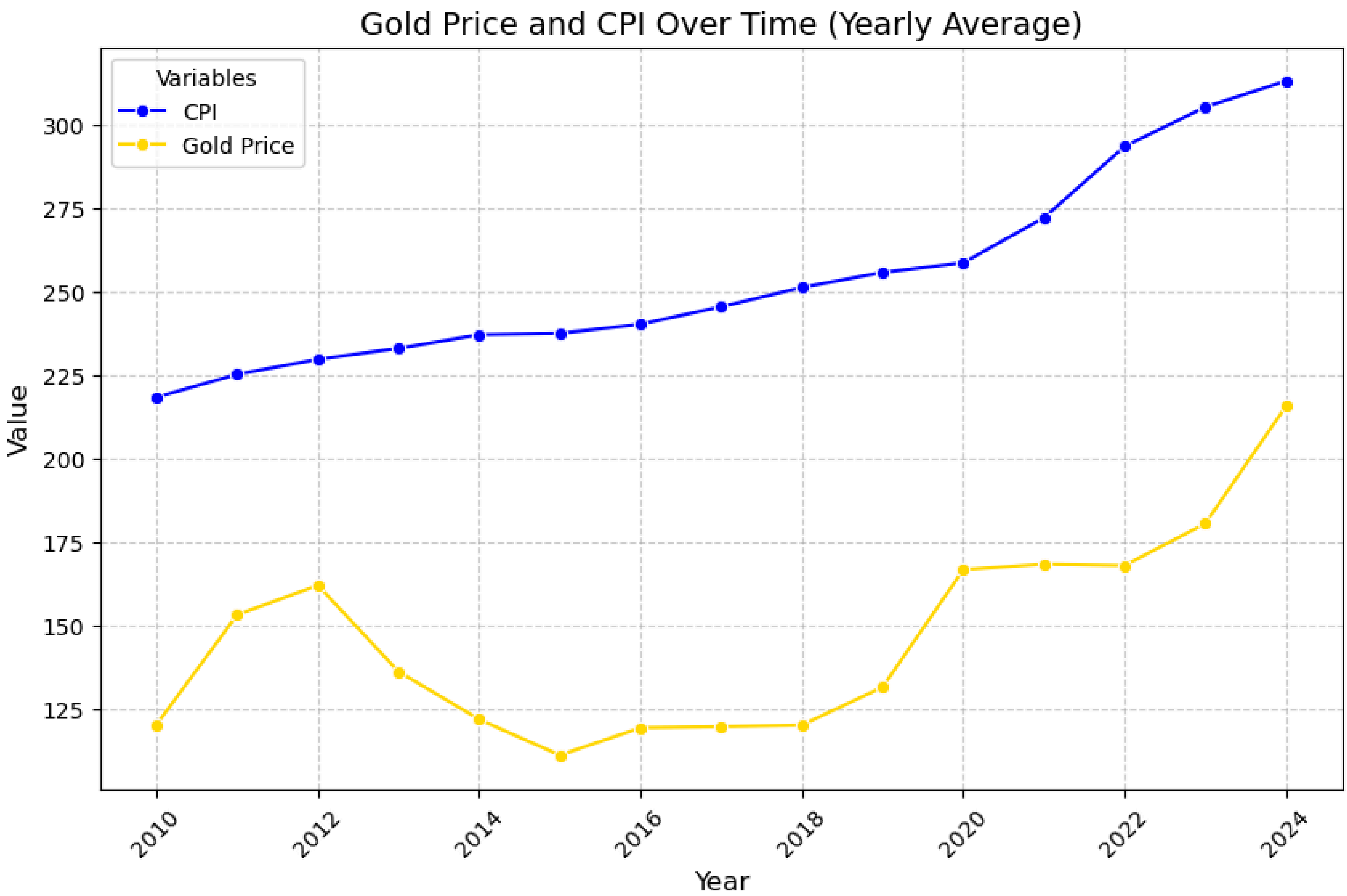
The graph suggests that the stock market has outperformed gold over the past decade. While gold might have served as a hedge against certain risks, its long-term growth has been more modest. Investors should consider their risk tolerance and investment objectives when deciding on the allocation between gold and stocks.

CPI with Gold prices

The Consumer Price Index (CPI) measures the average change in prices over time that consumers pay for a basket of goods and services. It is a key indicator of inflation and reflects the cost of living in an economy.

```
#graph6
data['year'] = pd.to_datetime(data['date']).dt.year
# Aggregate yearly averages for numeric columns
yearly_data = data.groupby('year').mean(numeric_only=True).reset_index()
# Plot Gold Price and CPI
plt.figure(figsize=(10, 6))
# Plot CPI
sns.lineplot(data=yearly_data, x='year', y='CPI', label='CPI', marker='o', color='blue')
# Plot Gold Price
sns.lineplot(data=yearly_data, x='year', y='gold close', label='Gold Price', marker='o', color='gold')
# Customize the plot
plt.title('Gold Price and CPI Over Time (Yearly Average)', fontsize=14)
plt.xlabel('Year', fontsize=12)
plt.ylabel('Value', fontsize=12)
plt.xticks(yearly_data['year'][::2], rotation=45)  # Show alternate years
plt.legend(title="Variables")
plt.grid(True, linestyle='--', alpha=0.6)
plt.show()
```

Graph6: CPI vs Gold price

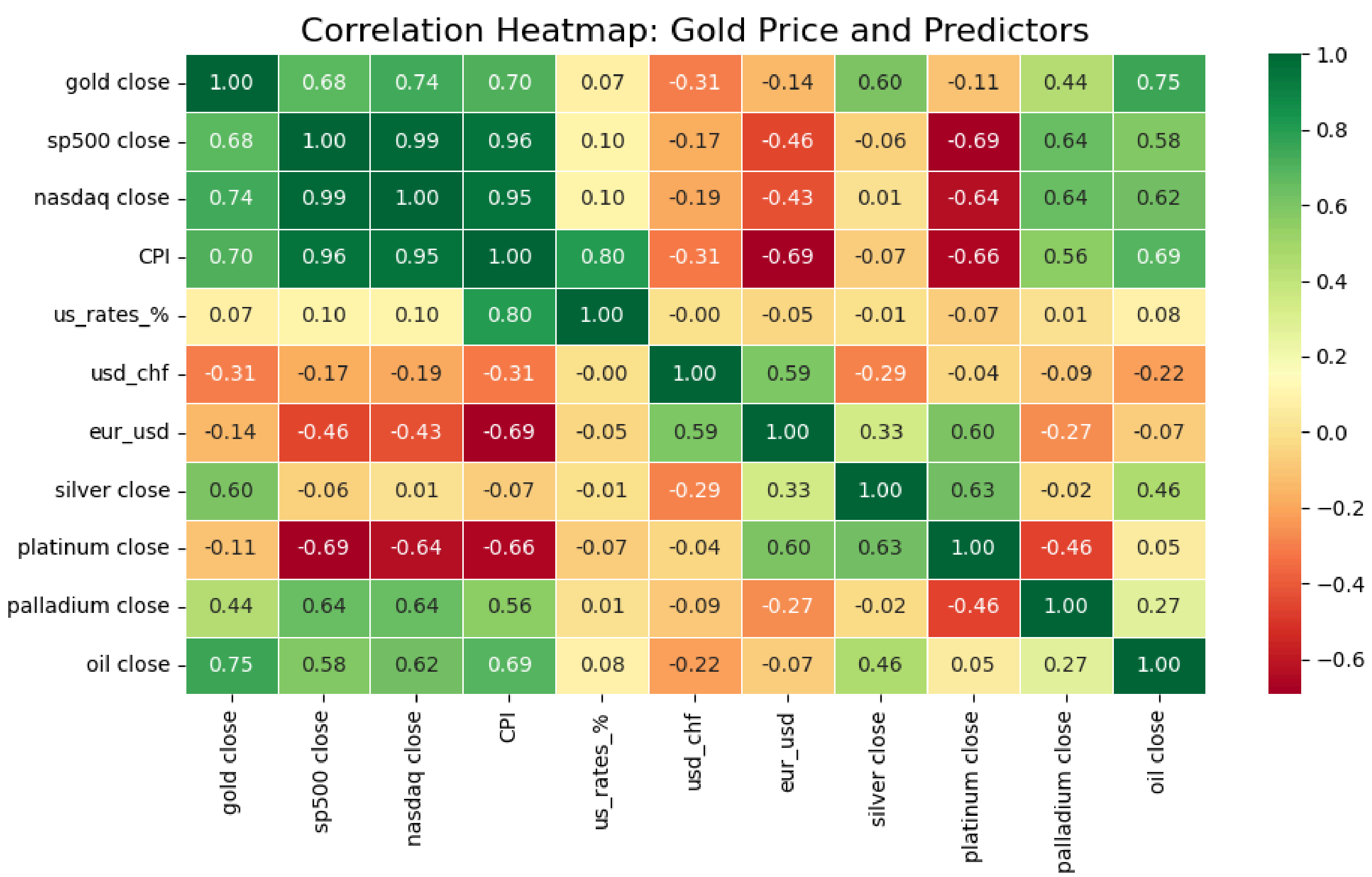


Correlation of Gold price and predictors

Correlation is a statistical measure that describes the strength and direction of a relationship between two variables. Positive correlation (green) indicates that as one variable increases, the other also tends to increase. Negative correlation (red) means that as one variable increases, the other tends to decrease. Values in between (yellow) signify a moderate relationship between the variables.

```python
#graph7

# Define the columns for pairwise correlation
predictors = ['gold close', 'sp500 close', 'nasdaq close', 'CPI', 'us_rates_%', 'usd_chf',
              'eur_usd', 'silver close', 'platinum close', 'palladium close', 'oil close']

# Calculate the correlation matrix
correlation_matrix = data[predictors].corr()

# Plot heatmap
plt.figure(figsize=(10, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='RdYlGn', fmt='.2f', linewidths=0.5)  # Red-Yellow-Green colormap
plt.title('Correlation Heatmap: Gold Price and Predictors', fontsize=16)
plt.tight_layout()
plt.show()
```

The heatmap below visualizes these relationships, with green representing strong positive correlations, red indicating no correlation, and yellow showcasing moderate correlations.



The heatmap reveals that gold prices are positively correlated with stock market indices, inflation, and oil prices. Conversely, they are negatively correlated with US interest rates and the strength of the US dollar. Understanding these relationships can help investors make informed decisions about investing in gold.

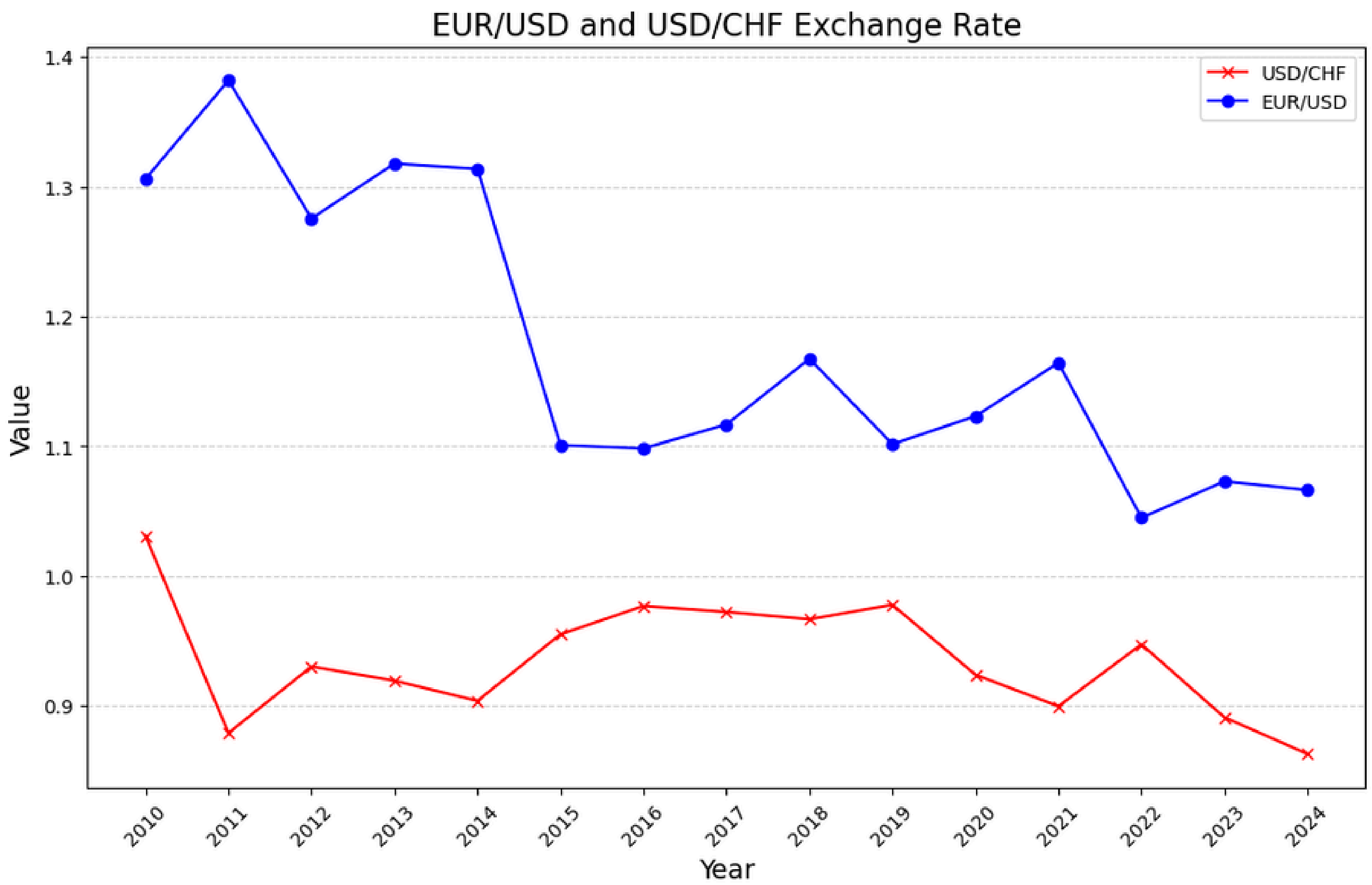
Foreign Exchange

For our gold price regression analysis project, integrating the **EUR/USD** and **USD/CHF** exchange rates could provide valuable insights into how global currency movements influence commodity trading volumes, particularly gold. Here's why they matter:

1. **EUR/USD (Euro to US Dollar):**
   This is one of the most traded currency pairs globally. A weaker USD (indicated by an increasing EUR/USD exchange rate) often leads to higher gold prices and volumes since gold is traded in USD. Investors might shift toward gold as the USD depreciates, viewing it as a hedge against currency devaluation.
2. **USD/CHF (US Dollar to Swiss Franc):**
   The Swiss Franc is considered a "safe haven" currency, much like gold. During periods of economic instability or geopolitical tensions, a stronger CHF (lower USD/CHF rate) often aligns with increased interest in gold. Both assets appeal to risk-averse investors, and changes in USD/CHF can reflect shifts in global investor sentiment toward safety.

```
#graph8
plt.figure(figsize=(12, 7))
# Bar plot for USD/CHF Exchange Rate
plt.plot(yearly_data['year'].astype(str), yearly_data['usd_chf'], color='red',marker='x', label='USD/CHF')
# Line plot for EUR/USD Exchange Rate
plt.plot(yearly_data['year'].astype(str), yearly_data['eur_usd'], color='blue', marker='o', label='EUR/USD')
# Customize the plot
plt.title('EUR/USD and USD/CHF Exchange Rate', fontsize=16)
plt.xlabel('Year', fontsize=14)
plt.ylabel('Value', fontsize=14)
plt.xticks(rotation=45)
# Add legend and grid
plt.legend()
plt.grid(axis='y', linestyle='--', alpha=0.6)
# Show the plot
plt.show()
```

Graph8: EUR/USD and USD/CHF Exchange Rate



Key Insights:

- The graph shows a general downward trend in the USD/CHF exchange rate and depreciation of the Euro against the US Dollar.
- This trend aligns with the negative correlation between gold prices and the strength of the US Dollar.
- Currency fluctuations, particularly in major pairs like USD/CHF and EUR/USD, can significantly influence gold prices.

Implications for Gold Price Regression Analysis:

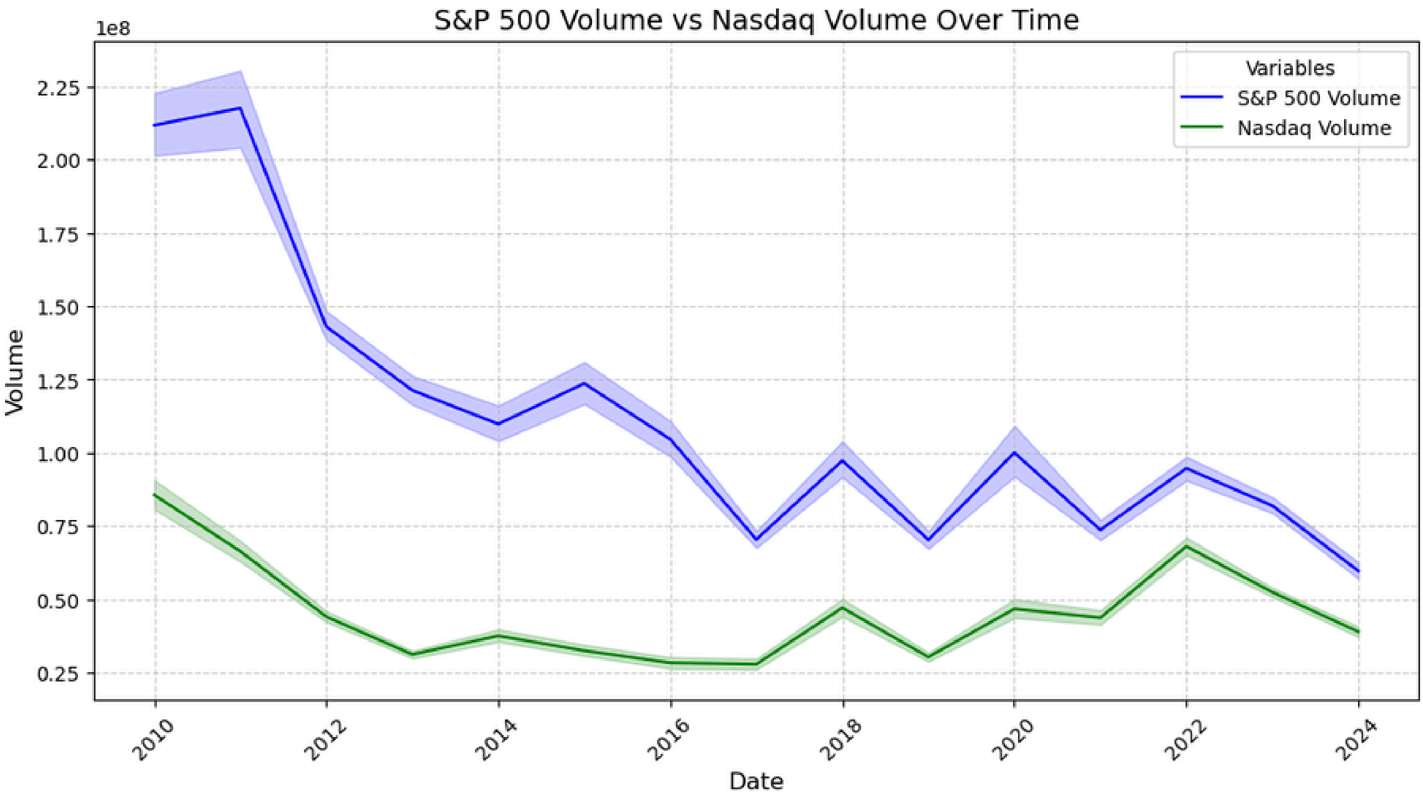
- Exchange rate data should be included as important predictors in gold price regression models.
- Time series analysis techniques can be used to model the dynamics of exchange rate movements and their impact on gold prices.
- Incorporating lagged values of exchange rates can help capture the time-dependent effects of currency fluctuations.

In essence, the graph emphasizes the crucial role of exchange rate dynamics in predicting gold prices. By considering currency movements alongside other relevant economic factors, analysts can improve the accuracy and robustness of their gold price regression models.

Comparing the Volumes of Stock market indices

```python
#graph9
# Line Plot: S&P 500 Volume vs Nasdaq Volume
plt.figure(figsize=(12, 6))
# Plot S&P 500 Volume
sns.lineplot(data=data, x='year', y='sp500 volume', label='S&P 500 Volume', color='blue')
# Plot Nasdaq Volume
sns.lineplot(data=data, x='year', y='nasdaq volume', label='Nasdaq Volume', color='green')
# Add titles and labels
plt.title('S&P 500 Volume vs Nasdaq Volume Over Time', fontsize=14)
plt.xlabel('Date', fontsize=12)
plt.ylabel('Volume', fontsize=12)
# Rotate x-axis labels
plt.xticks(rotation=45)
plt.legend(title="Variables")
plt.grid(True, linestyle='--', alpha=0.6)
# Show the plot
plt.show()
```

Graph9: Comparing the volumes of S&P500 and Nasadaq



Decreasing Trading Activity: The declining trend in both S&P 500 and Nasdaq 100 volumes could be attributed to various factors, such as changes in market structure, investor behavior, or regulatory changes.

Market Volatility: The higher volatility in S&P 500 volume suggests that it might be more sensitive to market events and investor sentiment compared to the Nasdaq 100.

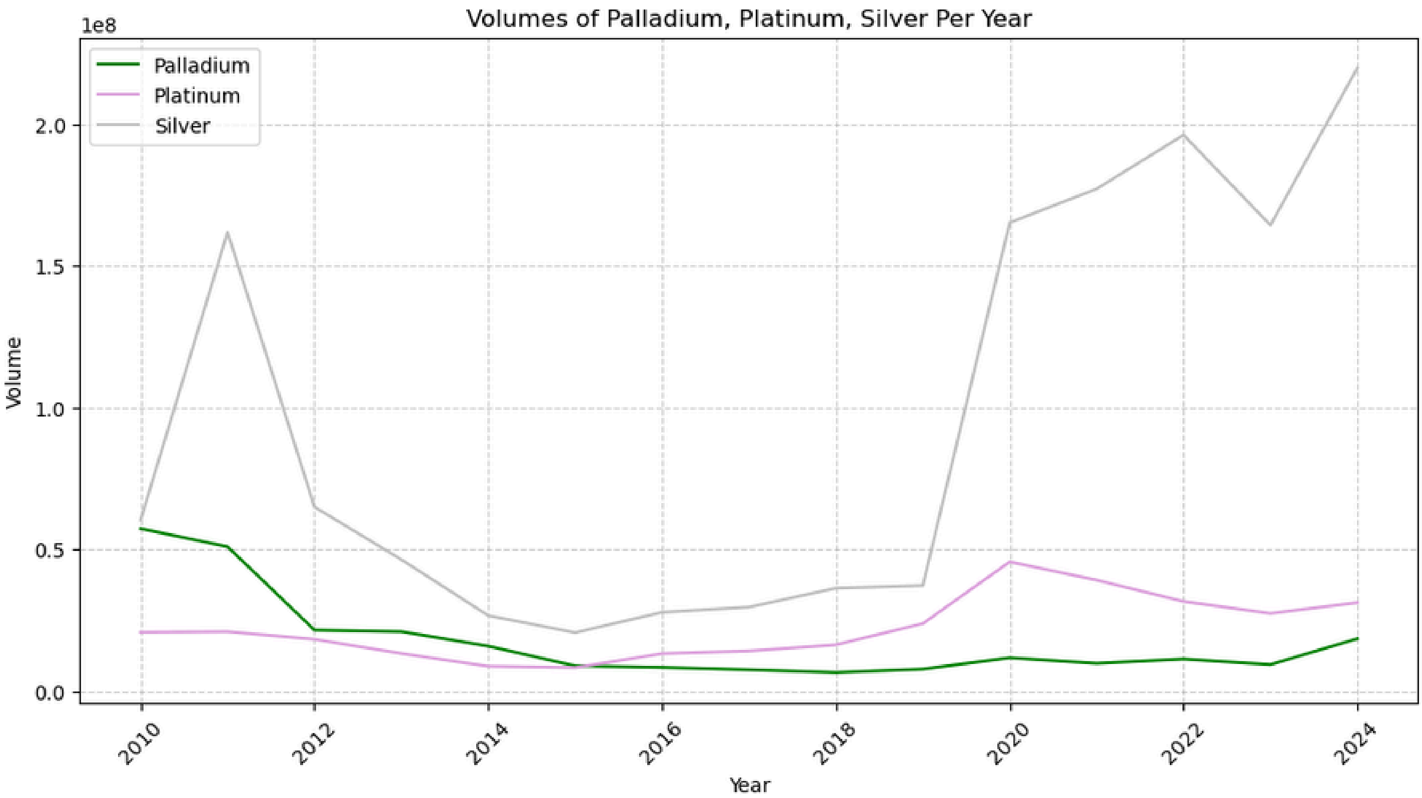
Sectoral Differences: The consistent higher volume in the S&P 500 reflects its broader representation of the US market, encompassing various sectors, while the Nasdaq 100 is heavily weighted towards technology and growth stocks.

The graph provides valuable insights into the evolution of trading activity in the S&P 500 and Nasdaq 100. The declining trend in both volumes and the higher volatility in the S&P 500 warrant further investigation to understand the underlying factors driving these dynamics.

Comparing the volume predictors with gold

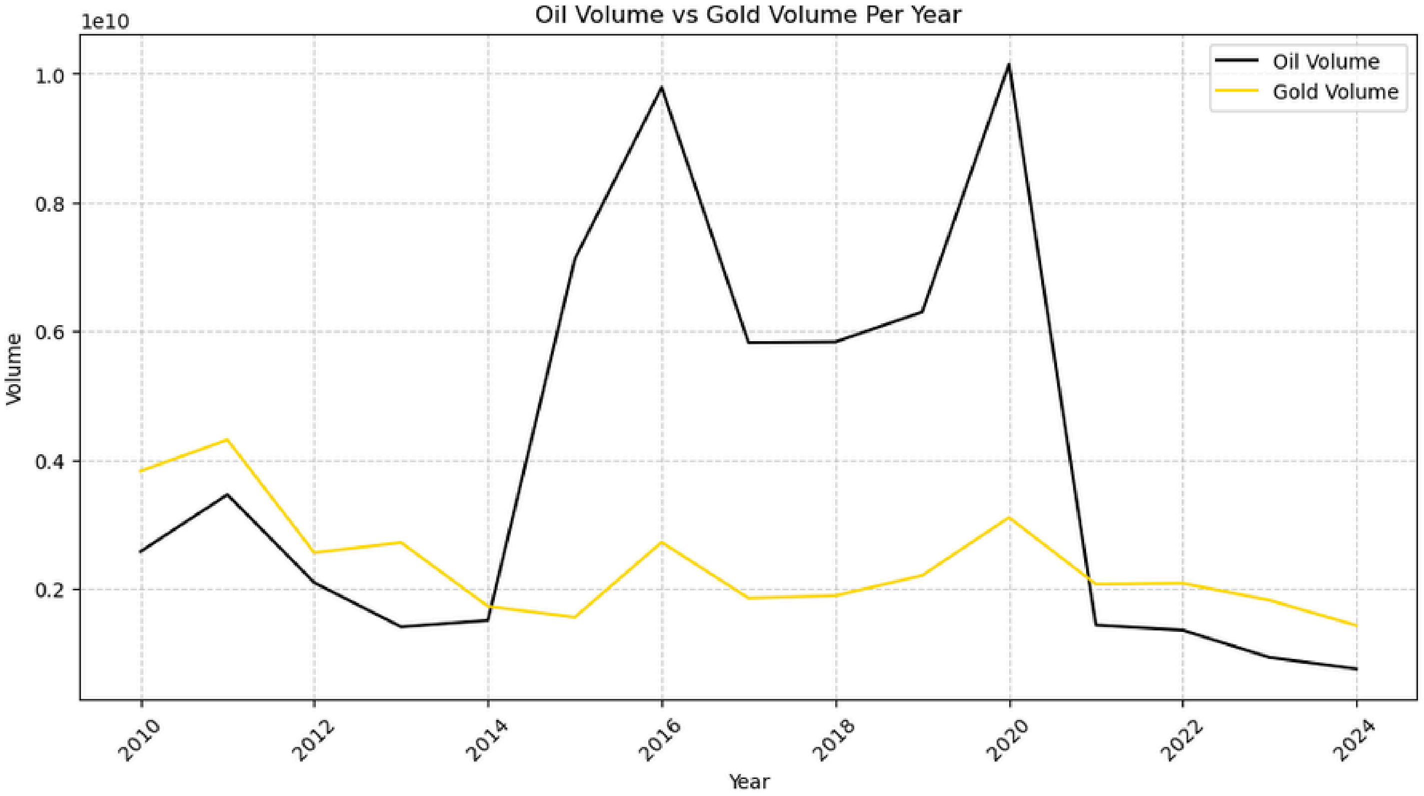
Graph10: comparing the volumes of Predictors such as palladium, platinum and silver by year.

```
#graph10
# Aggregate volumes of palladium, platinum, silver per year
yearly_volumes = data.groupby('year')[['palladium volume', 'platinum volume', 'silver volume']].sum().reset_index()

# Line plot for volumes of Palladium, Platinum, Silver per year
plt.figure(figsize=(12, 6))
sns.lineplot(data=yearly_volumes, x='year', y='palladium volume', label='Palladium', color='green')
sns.lineplot(data=yearly_volumes, x='year', y='platinum volume', label='Platinum', color='plum')
sns.lineplot(data=yearly_volumes, x='year', y='silver volume', label='Silver', color='silver')
plt.title('Volumes of Palladium, Platinum, Silver Per Year')
plt.xlabel('Year')
plt.ylabel('Volume')
plt.xticks(rotation=45)
plt.grid(True, linestyle='--', alpha=0.6)
plt.legend()

# Show plot
plt.show()
```



Graph11: comparing the oil volume with gold volume

```
#graph11
# Aggregate oil and gold volumes per year
yearly_volumes = data.groupby('year')[['oil volume', 'gold volume']].sum().reset_index()

# Line plot for Oil and Gold volumes per year
plt.figure(figsize=(12, 6))
sns.lineplot(data=yearly_volumes, x='year', y='oil volume', label='Oil Volume', color='black')
sns.lineplot(data=yearly_volumes, x='year', y='gold volume', label='Gold Volume', color='gold')

# Titles and labels
plt.title('Oil Volume vs Gold Volume Per Year')
plt.xlabel('Year')
plt.ylabel('Volume')
plt.xticks(rotation=45)
plt.grid(True, linestyle='--', alpha=0.6)
plt.legend()

# Show plot
plt.show()
```



Oil Volume Dominance: Throughout the period, the trading volume of oil significantly exceeds that of gold. This indicates that oil is a more actively traded commodity compared to gold.

Volatility: Both oil and gold volumes exhibit significant volatility over the years, with periods of high and low trading activity.

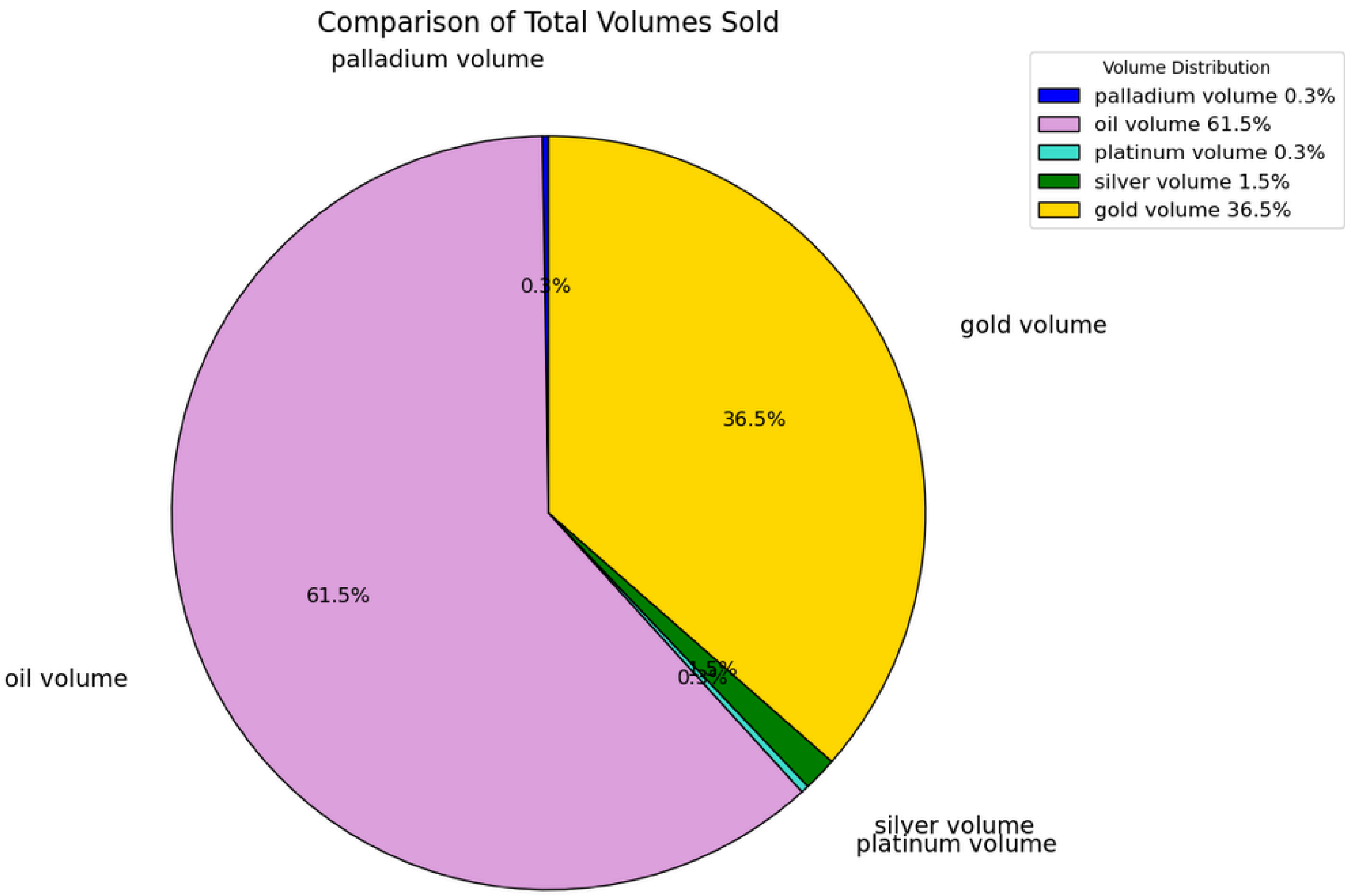
Oil Volume Peaks: Oil volume peaks in 2014 and 2020, suggesting periods of heightened trading activity and interest in the oil market.

Gold Volume Peaks: Gold volume peaks in 2011 and 2020, indicating periods of increased investor interest and trading in the gold market.

The graph provides valuable insights into the relative trading volumes of oil and gold. While oil consistently dominates in terms of trading activity, both commodities exhibit significant volatility and are influenced by various market factors.

Graph12: Piechart of Predictors

```
total_volumes = data[['palladium volume', 'oil volume', 'platinum volume', 'silver volume', 'gold volume']].sum()
# Create a pie chart
plt.figure(figsize=(10, 10))  # Adjusted figure size for better clarity
wedges, texts, autotexts = plt.pie(
    total_volumes,
    labels=total_volumes.index,
    autopct='%1.1f%%',  # Show percentages with one decimal place
    startangle=90,
    colors=['blue', 'plum', 'turquoise', 'green', 'gold'],
    textprops={'fontsize': 14},  # Increase font size for clarity
    labeldistance=1.2,  # Move labels outside the pie
    wedgeprops={'linewidth': 1, 'edgecolor': 'black'}  # Add edge separation for clarity
)
# Customize percentage texts to look smaller inside the pie and outside the pie
for autotext in autotexts:
    autotext.set_fontsize(12)  # Smaller font for the percentage inside
# Create the legend box to show percentages with matching colors
labels = [f'{label} {autotext.get_text()}' for label, autotext in zip(total_volumes.index, autotexts)]
plt.legend(
    labels,
    loc='upper left',
    fontsize=12,
    title="Volume Distribution",
    bbox_to_anchor=(1, 1)  # Position the legend box outside the chart
)
plt.title('Comparison of Total Volumes Sold', fontsize=16)
plt.show()
```



The pie chart provides a clear visual representation of the relative trading volumes of oil, gold, silver, platinum, and palladium. The dominance of oil and the significant share of gold highlight their importance in the commodity markets. Further analysis of historical data and market factors can provide deeper insights into the drivers of these trading volumes.

Graph13: Distribution Fit

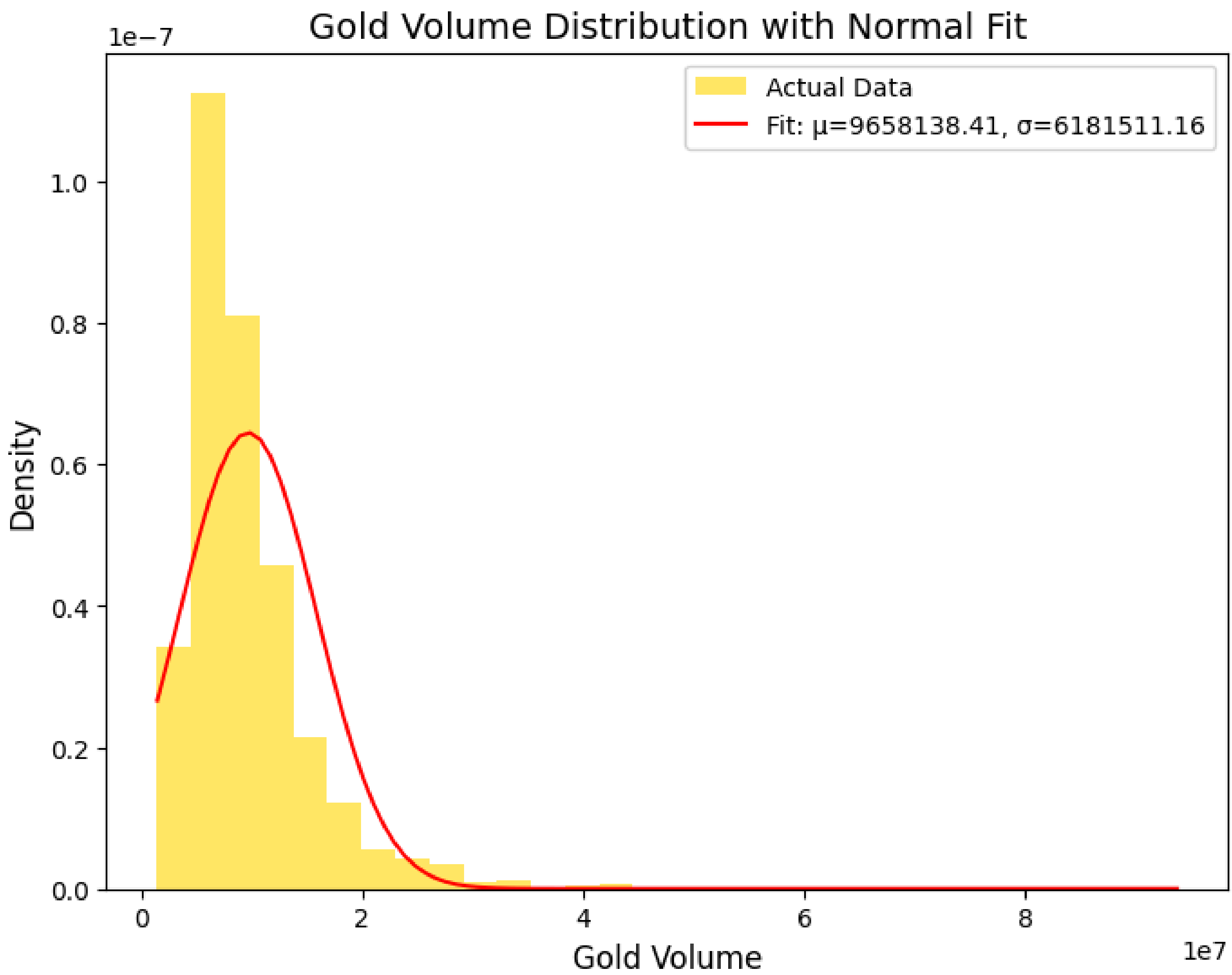
Distribution fit refers to the process of finding the most appropriate probability distribution that matches the characteristics of a given dataset. In other words, it's about identifying which theoretical statistical distribution (e.g., Normal, Exponential, Poisson) best represents the patterns or behavior of the data you're analyzing.

```python
#graph13
from scipy.stats import norm

# Fit a normal distribution to gold volume
mu, std = norm.fit(data['gold volume'])

# Plot histogram and PDF
plt.figure(figsize=(8, 6))
plt.hist(data['gold volume'], bins=30, density=True, alpha=0.6, color='gold', label='Actual Data')
x = np.linspace(data['gold volume'].min(), data['gold volume'].max(), 100)
pdf = norm.pdf(x, mu, std)
plt.plot(x, pdf, 'r-', label=f'Fit: μ={mu:.2f}, σ={std:.2f}')
plt.title('Gold Volume Distribution with Normal Fit', fontsize=14)
plt.xlabel('Gold Volume', fontsize=12)
plt.ylabel('Density', fontsize=12)
plt.legend()
plt.show()
```

Non-Normal Distribution: The actual distribution of gold volumes appears to be skewed to the right. This means there are more instances of lower trading volumes compared to higher volumes. A perfectly normal distribution would be symmetrical.

Normal Fit: The fitted normal curve does not perfectly overlap with the actual data. This indicates that the distribution of gold volumes is not perfectly normal.

Mean and Standard Deviation: The fitted normal distribution provides estimates of the mean (μ) and standard deviation (σ). These values can be used to characterize the central tendency and variability of the gold volume data.



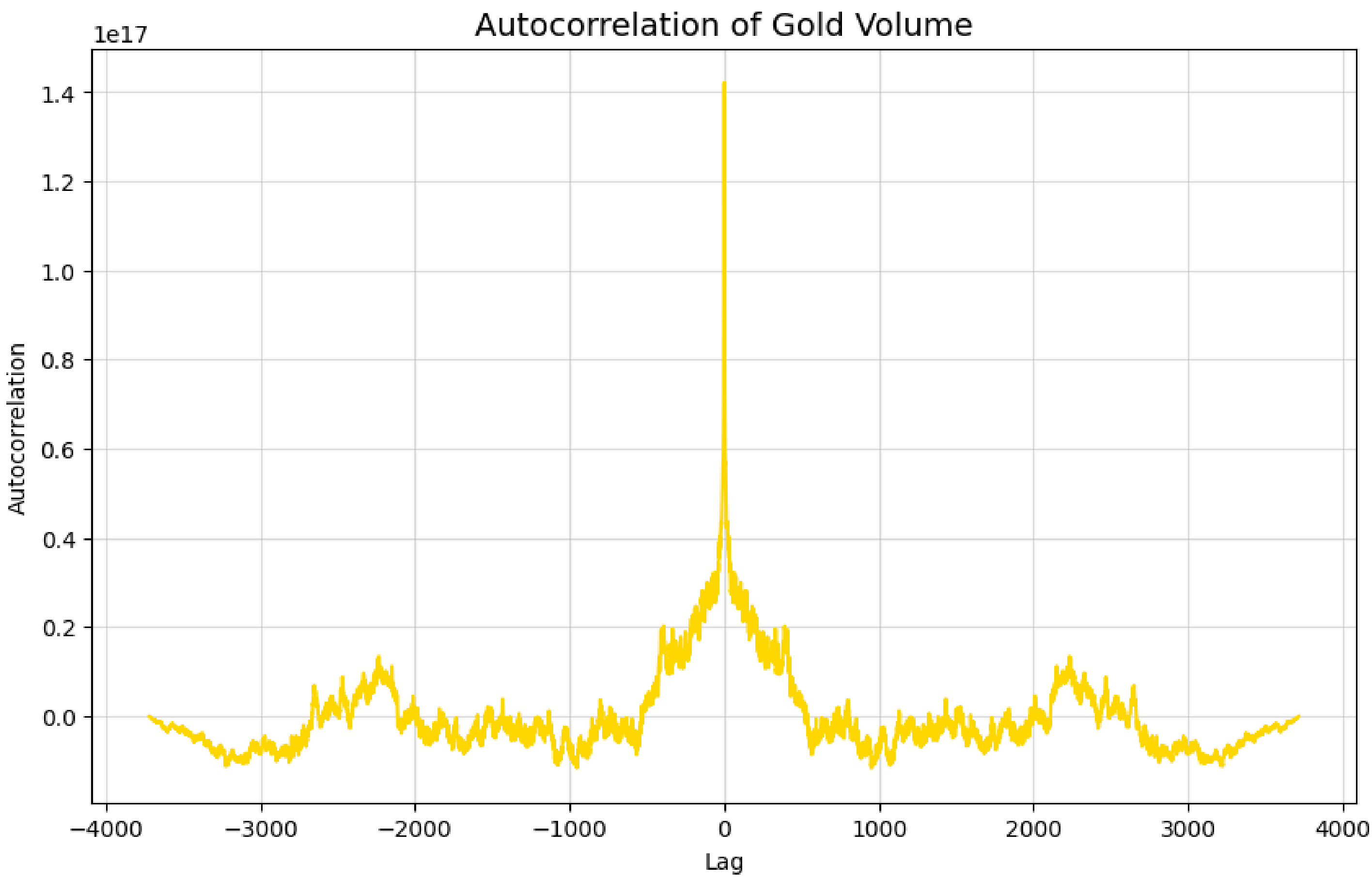
Insights:

1. **Gold Volume Distribution Characteristics**:
    - The skewed nature of the data suggests that extreme events (high gold trading volumes) occur infrequently but significantly impact the overall trend.
2. **Potential Misfit of Normal Distribution**:
    - Gold volumes are better represented by non-Normal distributions that can account for the skewness and heavy tails.
    - A Log-Normal distribution could provide a better fit, as gold trading volumes are multiplicative and grow asymmetrically over time.
3. **Gold Market Dynamics**:
    - High gold volumes may correspond to significant market events, such as financial crises or geopolitical instability, during which gold is viewed as a safe-haven asset.

Graph14: Auto Correlation Plot

An **autocorrelation plot** is a tool used in time-series analysis to measure and visualize the correlation of a dataset with its lagged (previous) versions over different time intervals. It helps to determine whether current values of a variable are influenced by past values and how far back that influence extends.

```
from scipy.signal import correlate

# Example: Autocorrelation of Gold volume
gold_volume = data['gold volume'] - data['gold volume'].mean()
autocorr = correlate(gold_volume, gold_volume, mode='full')

# Plot autocorrelation
lags = np.arange(-len(gold_volume) + 1, len(gold_volume))
plt.figure(figsize=(10, 6))
plt.plot(lags, autocorr, color='gold')
plt.title('Autocorrelation of Gold Volume', fontsize=14)
plt.xlabel('Lag')
plt.ylabel('Autocorrelation')
plt.grid(alpha=0.4)
plt.show()
```



Lagged Dependencies:

- The slight non-zero correlations beyond lag 0 suggest that past gold volumes have some influence on future values. This could be explored for forecasting purposes.

Possible Seasonal Trends:

- The periodic fluctuations hint at seasonal or cyclical behaviors in the gold volume. These could be linked to external factors like economic cycles or trading patterns.

Short-term Memory:

- The steep decline in autocorrelation as lag increases indicates that gold volume data may exhibit short-term dependencies rather than long-term memory.

Model Implications:

- Time-series models like ARIMA might capture these lagged dependencies, making them suitable for forecasting gold volumes.

The autocorrelation plot provides valuable insights into the temporal dependencies and patterns within the gold volume data. It highlights strong self-correlation at lag 0, with diminishing influence as the lag increases. Subtle periodic fluctuations suggest underlying seasonal patterns that could be further analyzed. This analysis is crucial for identifying time-series models that can effectively capture the dependencies and forecast gold trading volumes.