École Polytechnique Fédérale de Lausanne

Project

Financial Applications of Blockchains and Distributed Ledgers : Advanced Portfolio Construction

FIN-413, spring 2024

Yassin Al-Nuaimee, Mina Attia, Nils Delage, Marc El Khoury, Kamal Nour

April 7, 2025

Contents

Abstract

This report offers a detailed exploration of advanced portfolio construction methodologies that incorporate both cryptocurrency assets and traditional financial indices. It focuses on the dynamics of correlations between several major cryptocurrencies and between cryptocurrencies and traditional markets. The primary aim is to formulate effective diversification and risk management strategies that are adaptable to the increasingly complex global financial environment. The structure of the project is organized into three main sections: an initial review of the analytical methods employed, followed by an in-depth examination of the resultant data visualizations, and concluding with a resolution of assignment questions based on the thorough analysis conducted in each section.

0 Introduction

The rapid emergence of cryptocurrencies as investable assets has prompted a reevaluation of traditional portfolio management strategies. These digital assets, characterized by their high volatility and distinct correlation patterns with traditional markets, present both challenges and opportunities for asset allocation. This project aims to dissect these dynamics through a series of structured analyses and simulations, culminating in a set of optimized portfolio construction strategies.
All the code used to generate the analytics in this project can be found at
https://github.com/kamalnour/blockchain

0.1 Project Overview

The "Advanced Portfolio Construction" project is segmented into three primary analyses, each addressing specific facets of portfolio management in the context of both traditional and crypto markets:

1. **Exploratory Data Analysis (EDA)**: This foundational analysis will utilize linear and logarithmic returns to investigate the characteristics of asset distributions, identify outliers, and understand the implications of non-trading periods in traditional assets. Tools such as correlation heatmaps, rolling correlations, and density plots will be employed to provide insights into the data.
2. **Portfolio Construction Techniques**: Following the EDA, the project will explore various portfolio construction techniques:
   - Construction of equally weighted portfolios at key historical dates.
   - Application of eigenvalue clipping to clean covariance matrices and improve the stability of portfolio optimization under varying market conditions.
   - Use of Euler risk contributions to analyze and compare the impact of raw and cleaned covariance matrices on portfolio risk.
   - Development of risk-based portfolios using advanced methods like Minimum Variance, Equal Risk Contribution, and Hierarchical Risk Parity.

3. **Advanced Optimization Techniques**: The project will extend the Hierarchical Risk Parity model by incorporating alternative measures of dependency beyond traditional correlation metrics. This involves applying novel distance metrics to enhance the robustness and performance of the portfolios under different market conditions.

0.2 Detailed Plan

Each section of the report will address specific questions and sub-questions as outlined in the project brief:

- **Section 1 - Exploratory Data Analysis (EDA)**:
  - Discussion on the choice between linear and logarithmic returns, supported by statistical analysis of asset distributions.
  - Identification and treatment of outliers, and their impact on subsequent analyses.
  - Examination of the effects of market closure on correlation calculations and portfolio optimization, proposing methodologies to mitigate these issues.
- **Section 2 - Portfolio Construction Techniques**:
  - Description and justification of the data window size for covariance calculations, followed by eigenvalue clipping to enhance matrix stability.
  - Comparative analysis of risk contributions and portfolio diversification metrics across different dates and conditions.
  - Construction and comparison of various portfolio models, assessing their performance and suitability under defined market conditions.
- **Section 3 - Advanced Optimization Techniques**:
  - Development of an alternative Hierarchical Risk Parity model using new dependency metrics.
  - Evaluation and comparison of portfolio performance using traditional and modified models across significant market events.

0.3 Concluding Remarks

The outcomes of this analysis will contribute to a deeper understanding of the complexities involved in integrating cryptocurrencies into traditional portfolio management frameworks, offering insights that could be pivotal for investors looking to navigate this evolving market landscape.

1 Exploratory Data Analysis (EDA)

The analysis begins with initial data processing, including loading, summarizing, and checking for missing values, followed by a deeper dive into specific asset types and visualization techniques.

1.1 Initial Data Loading and Summary

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as stats
import statsmodels.api as sm

df = pd.read_csv('2024_03_25_-_epfl_fin413_crypto_portfolio_construction_-
    _project_dataset.csv', skiprows=1, index_col='time')
df.index = pd.to_datetime(df.index)
df.head()
df.describe()
df.isna().sum()
```

The dataset is loaded with a focus on the 'time' column as the index, which is converted to a DateTime object to facilitate time-series analysis. Initial exploratory functions like `head()`, `describe()`, and checking for missing values with `isna().sum()` provide an overview of the dataset's structure, completeness, and statistical summary.

1.2 Resampling and Separation of Asset Types

```
df_monthly = df.resample('1m').mean()
traditional_monthly = df_monthly[df_monthly.columns[-4:]]
crypto_monthly = df_monthly[df_monthly.columns[:-4]]
```

Data is resampled to a monthly frequency to smooth out daily fluctuations. Assets are divided into traditional indices and cryptocurrencies to analyze them separately, reflecting different market behaviors and risk profiles.

1.3 Visualizing Data Trends

```
plt.figure(figsize=(12,6))
sns.set_style("whitegrid")
sns.lineplot(data=crypto_monthly)
plt.show()

plt.figure(figsize=(12,6))
sns.lineplot(data=traditional_monthly)
plt.show()
```

Using line plots, we visualize monthly trends for both asset groups. The 'whitegrid' style enhances visual clarity, helping to identify trends and volatilities in the asset price movements.

1.3.1 Visualizing Cryptocurrency Trends Over Time

The visualization presented below reflects the monthly return fluctuations of various cryptocurrencies from the beginning of 2018 to early 2024. This result is a line plot made with the Seaborn library, designed to highlight trends and volatility in time-series data.

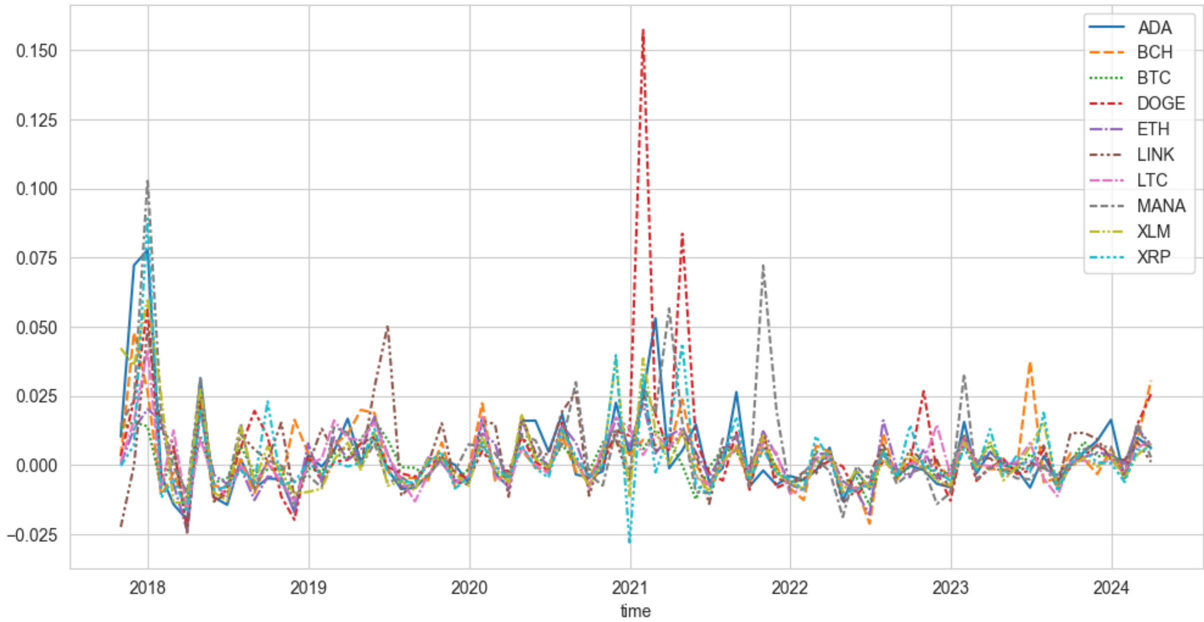

Figure 1: Monthly Return Fluctuations of Various Cryptocurrencies.

- The x-axis, a historical timeline that spans from 2018 to 2024, allows for the observation of long-term trends and cyclic behaviors within the cryptocurrency markets.
- The y-axis denotes the return values, indicative of percentage changes, normalized for comparative analysis.
- Noticeable volatility of returns is observed, particularly in the early part of 2018 and again in 2021 related to the big surge, followed by crashes that happened in the crypto market. Several events at these time affected the sentiment across the cryptocurrency spectrum. Most notably in 2021, the LUNA and UST crash, later followed by the FTX collapse.
- The similarities in the movement patterns across different cryptocurrencies suggest a market-wide influence, hinting at the presence of underlying common factors or systemic market drivers impacting the entire sector.
- Post-2021, the extremity of fluctuations appears to be moderate, implying a possible shift in market dynamics, such as increased regulatory clarity, maturation of the market, and changes in investor behavior and sentiment, particularly with the developments of institutional-grade investment tools, such as ETFs.

This visualization is not only helpfull in identifying periods of heightened market activity, it also serves as a basis for further statistical analysis, to discern the factors influencing these periods of pronounced volatility. Such insights can be used by both short-term traders and long-term investors in strategizing their market participation.

1.3.2 Assessing Volatility in Traditional Financial Indices

The line plot below represents the monthly returns of well established traditional financial indices over the same period.

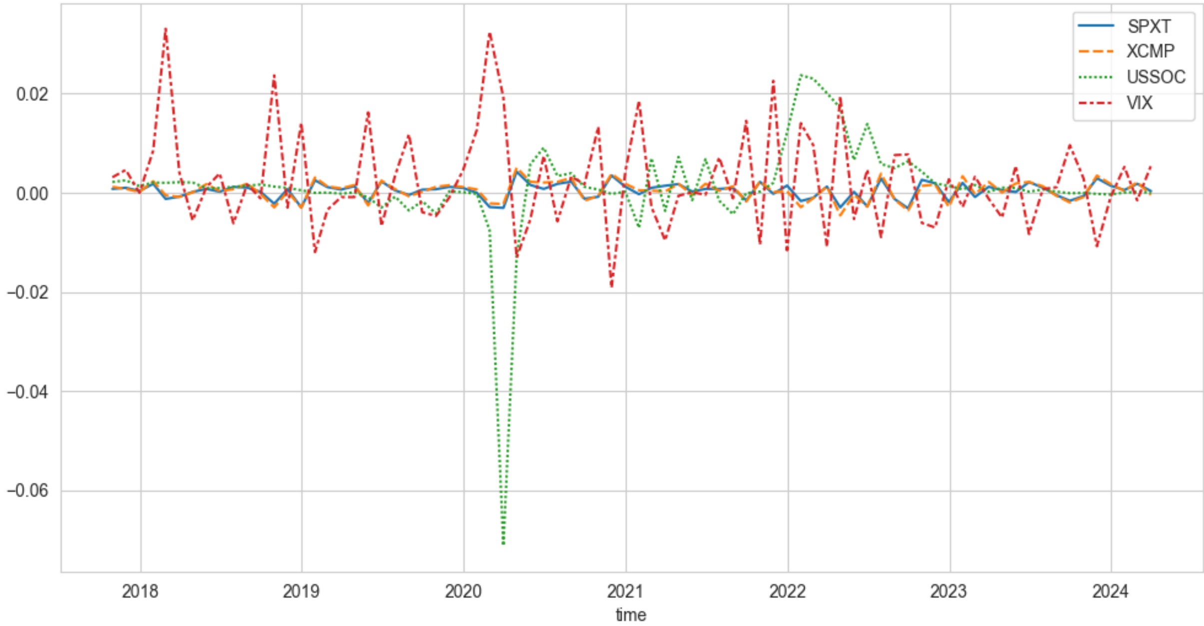

Figure 2: Monthly Return Fluctuations of Traditional Financial Indices.

Interpreting Traditional Market Dynamics
Key takeaways from the visualization include:

- The x-axis time span allows for the observation of how these indices have responded to various economic cycles and events.
- The returns on the y-axis, while showing fluctuations, remain within a relatively narrow band, especially when compared to the earlier analyzed cryptocurrency returns.
- The NASDAQ and SP 500 indices tend to follow a similar trend pattern, which may reflect the overlapping constituents or correlated market sectors.
- The USSOC index displays a distinct sharp decline at one point, hinting at a possible significant market or policy event affecting short-term interest rates.
- The VIX index's occasional spikes underscore moments of market turmoil or investor anxiety, offering a contrast to the general trend stability observed in the other indices.

This traditional index performance analysis provides a stark contrast to the crypto asset behavior, underscoring the diversification potential when combining different asset classes in a portfolio construction context.

1.4 Volatility and Return Analysis

```
crypto_daily = df[df.columns[:-4]]
crypto_desc = crypto_daily.describe()
crypto_desc.loc['std'].sort_values(ascending=False)*100

crypto_volatility = crypto_daily.groupby(crypto_daily.index.year).std()*100
crypto_volatility.mean(axis=1)

fig, axs = plt.subplots(2, 1, figsize=(12, 12))
sns.lineplot(data=crypto_daily[crypto_daily.index.year==2017], ax=axs[0])
axs[0].set_title('Most Volatile Year Crypto Returns')
sns.lineplot(data=crypto_daily[crypto_daily.index.year==2023], ax=axs[1])
axs[1].set_title('Least Volatile Year Crypto Returns')
plt.show()
```

This section details the volatility analysis across different cryptocurrencies and years. It highlights how market conditions can significantly affect asset performance, with visualizations for the most and least volatile years.

1.4.1 Volatility Analysis of Cryptocurrency Daily Returns

The standard deviation of daily returns provides insight into the volatility and risk profile of each cryptocurrency. The following table lists the calculated standard deviation for selected cryptocurrencies, sorted in descending order of volatility.

Cryptocurrency	Standard Deviation (%)
DOGE	10.65
MANA	9.80
BCH	7.85
ADA	7.10
LINK	6.83
XLM	6.59
XRP	6.42
LTC	5.37
ETH	4.77
BTC	3.80

Table 1: Standard Deviation of Daily Returns for Cryptocurrencies.

1.4.2 Implications of Volatility Measures

The results from the table can be interpreted as follows:

- **DOGE**, known for its high volatility, tops the list. This can be attributed to several factors, including speculative trading and sensitivity to social media influence. The concentration of DOGE's supply over the years, coupled with its high volatility, could also be a sign of market manipulation.

- **MANA**, a cryptocurrency associated with the Decentraland virtual world platform, also exhibits high volatility, possibly driven by the emerging nature of virtual real estate and NFT markets, very sensitive to social media aswell.
- **BTC**, being the most established cryptocurrency, shows the least volatility, indicating a level of market stability relative to other digital assets.
- **ETH**, despite its lower volatility compared to smaller altcoins, still shows significant price movements, which may reflect its ongoing development and a rapidly changing ecosystem.

Investors might utilize this volatility information to tailor their asset allocation according to their risk appetite, with potential strategies ranging from a focus on stability with assets like BTC and ETH, to pursuing higher returns—and accepting higher risk—with assets like DOGE and MANA.

1.4.3 Annual Volatility Trends in Cryptocurrency Markets

The standard deviation, as a measure of annual volatility, sheds light on the fluctuating nature of cryptocurrency markets. The table below summarizes the mean annual volatility percentages for the studied period.

Year	Average Volatility (%)
2017	13.67
2018	7.59
2019	5.00
2020	5.79
2021	8.94
2022	4.73
2023	3.69
2024	4.26

Table 2: Average Annual Volatility of Cryptocurrencies.

1.4.4 Contextualizing the Volatility Fluctuations

The annual volatility data can be contextualized as follows:

- The year **2017** was characterized by remarkable volatility, which aligns with the rapid price increases and the heightened public interest in cryptocurrencies during that period.
- A noticeable decline in volatility in **2018** corresponds with a cooling-off period following the late-2017 cryptocurrency boom.
- **2019** and **2020** demonstrate lower volatility levels, which could be indicative of a market finding its equilibrium after the extreme fluctuations of previous years.

- A resurgence in volatility during **2021** can be associated with the increased institutional interest in cryptocurrencies and the rise of DeFi and NFTs, bringing in new dynamics and market participants. The May 2020 halving of Bitcoin emissions, has probably been a catalysor of this effect.
- The decreasing trend in volatility from **2022** to **2024** may suggest growing market maturity and the establishment of clearer regulatory frameworks that reduce uncertainty and speculative trading. This helps to put into perspective the impact of the 2022 scandals, which thus appear to have had a moderate effect on annual volatility.

This trend analysis of annual volatility is crucial for investors who seek to understand the risk environment of the cryptocurrency market over time. Lower volatility in recent years could make cryptocurrencies more attractive to a broader base of investors, particularly those with a lower risk tolerance.

1.4.5 Comparative Analysis of Cryptocurrency Returns in High and Low Volatility Environments

The subplot visualizations provide a comparative view of the daily cryptocurrency returns during the most volatile year (2017) and the least volatile year (2023) observed in the dataset.

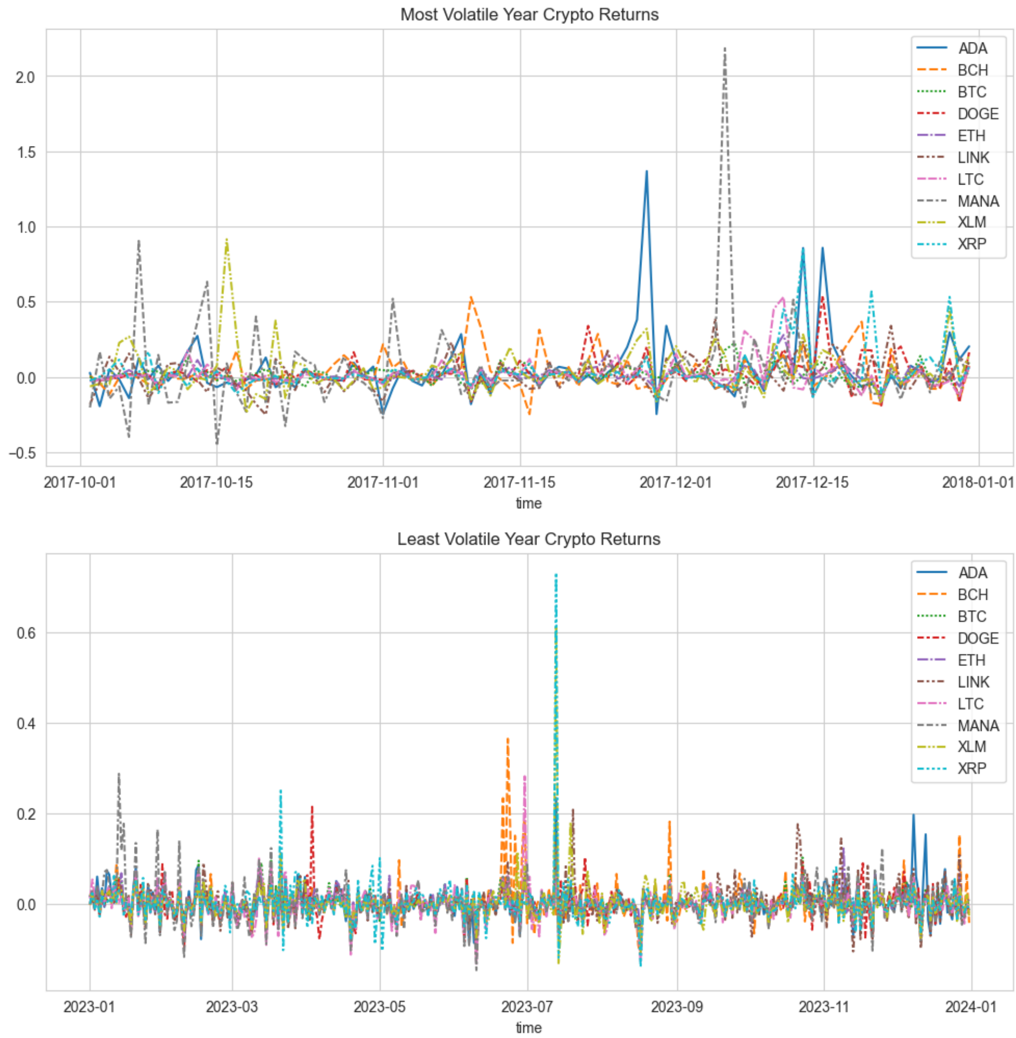

Figure 3: Comparison of Daily Crypto Returns for the Most and Least Volatile Years.

1.4.6 Year of High Volatility: 2017

The upper plot, titled "Most Volatile Year Crypto Returns," demonstrates significant fluctuations in daily returns within 2017. This is characterized by:

- Sharp spikes in returns, with certain days experiencing drastic changes in value, reflective of the high-risk, high-reward nature of the cryptocurrency market during that year.
- The presence of extreme values, particularly noticeable in currencies such as BCH (Bitcoin Cash), which usually correspond to specific market events or news that caused sudden investor reactions, but can also simply be the result of surging attention over a specific asset, which would provoke major price changes, especially in 2017 when the cryptocurrency market was much smaller and less liquid.

1.4.7 Year of Low Volatility: 2023

Conversely, the lower plot, titled "Least Volatile Year Crypto Returns," shows a subdued range of returns in 2023, with observations revealing:

- More constrained movements in daily returns, indicating a relatively calmer market environment with fewer instances of sharp price changes.
- The fluctuations are notably milder compared to 2017, suggesting increased market maturity, perhaps due to clearer regulations or a more established presence in investors' portfolios.

This visual analysis emphasizes the contrast between the two distinct market conditions and could serve as a crucial element in assessing the risk profile and investment strategies suitable for different market climates. The evident reduction in volatility in later years may also indicate the evolving nature of the cryptocurrency market, potentially attracting a different type of investor compared to the early, more turbulent years.

1.5 Logarithmic Returns and Detailed Asset Comparison

```
crypto_daily.apply(lambda x: np.log(x+1))

plt.figure(figsize=(12,6))
sns.lineplot(crypto_daily['DOGE'])
plt.title('DOGE fluctuations')
plt.show()

crypto_log = crypto_daily.apply(lambda x: np.log(x + 1))

fig,axs=plt.subplots(2,1,figsize=(12,12))
plt.figure(figsize=(12,6))
sns.lineplot(crypto_log[crypto_log.index.year==2017],ax=axs[0])
axs[0].set_title('Most Volatile Year Crypto Log Returns')
sns.lineplot(crypto_log[crypto_log.index.year==2023],ax=axs[1])
axs[1].set_title('Least Volatile Year Crypto Log Returns')
plt.show()

plt.figure(figsize=(12,6))
sns.lineplot(data=crypto_daily['DOGE'], label='Linear')
sns.lineplot(data=crypto_log['DOGE'], label='Log')
plt.title('DOGE fluctuations: Linear vs Logarithmic')
plt.show()

crypto_log.describe().loc['std'].sort_values(ascending=False)*100
```

Comparison of linear and logarithmic returns for Dogecoin (DOGE) is presented to illustrate how log transformations can stabilize return data, making trends and volatilities easier to analyze.

1.5.1 Dogecoin (DOGE) Daily Price Fluctuations

This visualization represents the daily fluctuations of Dogecoin (DOGE) over a multi-year period.

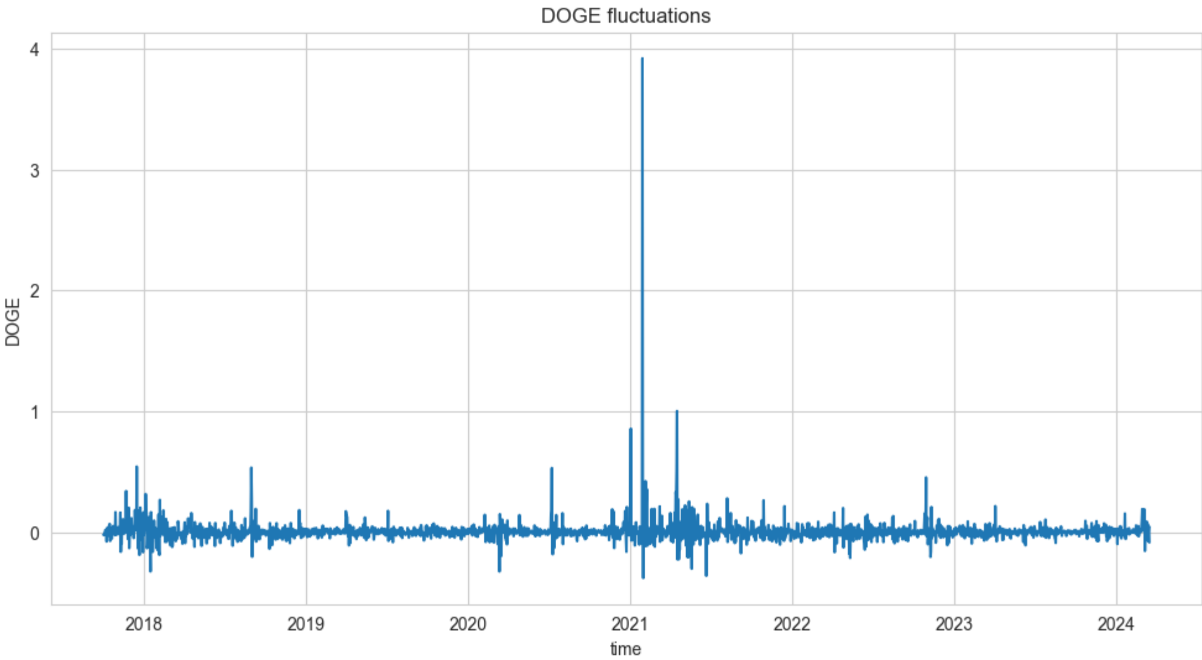

Figure 4: Daily Fluctuations of Dogecoin (DOGE).

Insights from the DOGE Plot The line plot elucidates the following points of interest:

- The x-axis, displaying a time frame from 2018 to 2024, captures the long-term trends of DOGE prices, providing insights into the cryptocurrency's behavior over time.
- The y-axis, representing the daily transformation of DOGE prices, shows values ranging from slight negatives to above 4.
- There are notable spikes observed, especially a significant one around the year 2021. This peak may correspond with real-world events that sparked a substantial increase in DOGE's price and public interest.
- Apart from the pronounced spike, the rest of the plot shows a series of smaller fluctuations which are characteristic of the cryptocurrency's volatility but do not reach the extremity of the 2021 event.

This graph serves as a clear illustration of DOGE's price volatility over the years and may be particularly useful in analyzing the market reactions to specific events. Such a detailed view helps investors and analysts alike to understand the magnitude of price changes and to infer the risk associated with investing in such assets.

1.5.2 Comparing Linear and Logarithmic Fluctuations of Dogecoin (DOGE)

The figure below juxtaposes the linear and logarithmic daily returns of Dogecoin (DOGE) to demonstrate how different transformations can affect the perception of data volatility.

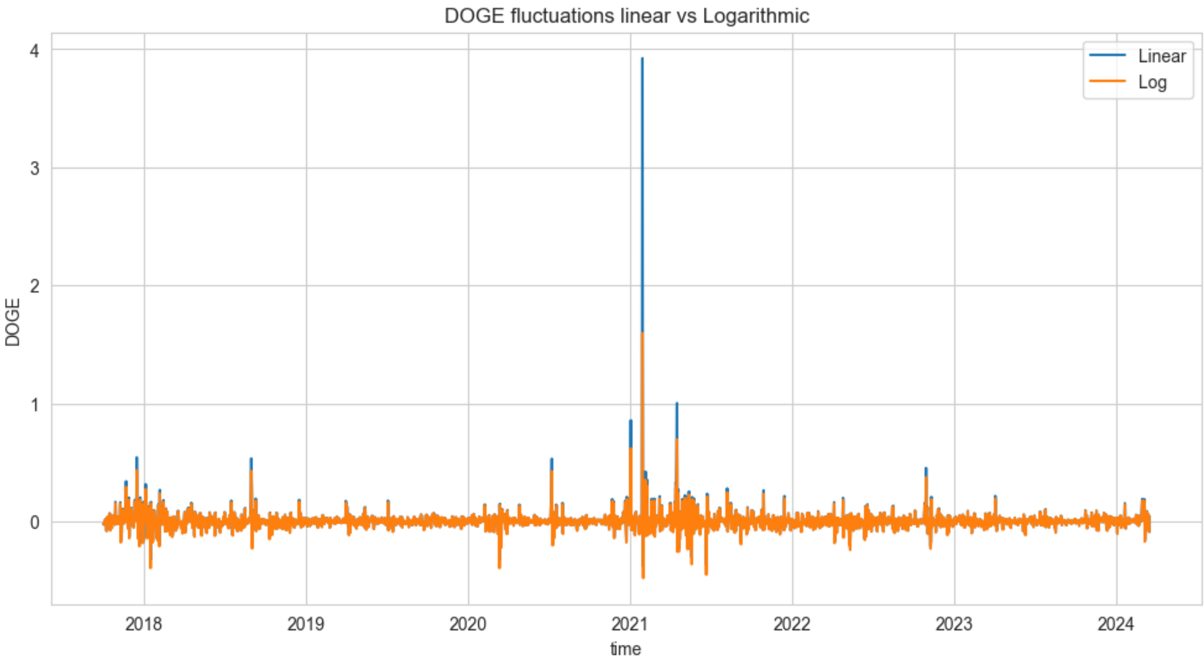

Figure 5: Dogecoin Fluctuations: Linear vs Logarithmic Scale.

Analytical Observations The line plot yields several insights:

- The linear plot, depicted in blue, shows the raw values of DOGE's daily returns. It highlights the actual scale of fluctuations, with some extreme spikes, particularly around 2021.
- The logarithmic plot, overlaid in orange, represents the log-transformed returns, which diminishes the impact of extreme values and can provide a more nuanced view of growth rates and relative changes.
- The logarithmic transformation is particularly useful for financial data like cryptocurrency returns, as it tends to normalize the percentage changes, making trends more discernible, especially when dealing with asymmetric volatility or skewness in the data.
- The standard deviation calculated on the logarithmic returns, when multiplied by 100, would provide the average percentage volatility, allowing for a standardized comparison across different time periods or assets.

This comparative visualization underscores the utility of log transformations in financial analysis by smoothing out the most extreme data points, which can be valuable for identifying underlying trends and reducing the visual impact of outliers. Such a transformation may also better align with certain financial models and risk assessments where proportional changes are more meaningful than absolute ones.

1.5.3 Comparative Analysis of Log Returns for Cryptocurrencies

This figure contrasts the logarithmic returns of cryptocurrencies during the most and least volatile years, showcasing the utility of logarithmic scaling in analyzing financial data.

Figure 6: Cryptocurrency Log Returns in the Most and Least Volatile Years.

Logarithmic Transformation Insights The log transformation stabilizes the variance in financial time series data and can better handle the asymmetric nature of positive and negative returns. Here's what we observe from the log-transformed data:

- The upper subplot, titled "Most Volatile Year Crypto Log Returns," indicates that even with the log transformation, 2017 experienced significant volatility. However, the extreme values are less pronounced compared to the linear returns, facilitating an easier comparison of relative changes over time.
- The lower subplot, titled "Least Volatile Year Crypto Log Returns," presents a more compact and less spread-out range of log returns for 2023, emphasizing the year's stability when compared to 2017's turbulent market.

- When compared with the earlier provided non-logarithmic figures, the log transformation in the current plots reduces the visual impact of outliers and extreme fluctuations. This is particularly useful in the context of financial data, where such transformations aid in identifying trends and potential periods of financial stress or growth.

Comparison with Non-Logarithmic Returns Contrasting these log-transformed plots with the previous non-logarithmic ones, we see that:

- The non-logarithmic plots highlight the absolute magnitude of return fluctuations, which can be useful for understanding the raw volatility and risk profile of an asset.
- The logarithmic plots provide a relative view of returns, often giving a clearer picture of the proportional changes and trends, which might be more relevant for multiplicative processes like asset growth.

This comparative analysis underscores the importance of choosing the right scale for financial analysis. Logarithmic returns tend to give a more normalized and interpretable view, particularly for asymmetric return distributions common in cryptocurrency markets.

1.6 Monthly Returns for Traditional Equity Indices

```
equity_monthly=df_monthly[['SPXT','XCMP']]
equity_monthly.head(12)
```

The table below lists the first 12 months of returns for two major equity indices, the S&P 500 Total Return Index (SPXT) and the NASDAQ Composite Index (XCMP), as extracted from the monthly dataset.

Time	SPXT Return	XCMP Return
2017-10-31	0.0772%	0.1195%
2017-11-30	0.1013%	0.0784%
2017-12-31	0.0361%	0.0163%
2018-01-31	0.1808%	0.2321%
2018-02-28	-0.1244%	-0.0524%
2018-03-31	-0.0778%	-0.0838%
2018-04-30	0.0168%	0.0083%
2018-05-31	0.0783%	0.1741%
2018-06-30	0.0214%	0.0348%
2018-07-31	0.1189%	0.0724%
2018-08-31	0.1042%	0.1847%
2018-09-30	0.0194%	-0.0222%

Table 3: Monthly Returns for SPXT and XCMP Indices.

1.6.1 Insights from the Monthly Returns

From the table, we can derive several insights:

- Most months show positive returns for both indices, which suggests a period of overall growth in the equity markets during this timeframe.
- The highest monthly return for the SPXT index within this period is in January 2018, while for the XCMP index, it's also in January 2018, aligning with a strong start to that year in the equity markets.
- There are instances of negative monthly returns, specifically in February and March 2018, reflecting short-term downturns or corrections in the market.
- The variation between the two indices in certain months may indicate sector-specific movements or different market dynamics affecting each index.

This analysis of traditional equity index returns is important as it provides a benchmark for assessing the performance of other asset classes, such as cryptocurrencies, within the same period. It also serves as a reference for the overall market climate influencing investor sentiment and portfolio strategy.

1.7 Daily Returns for Traditional Financial Market Indices

```
traditional_daily=df[df.columns[-4:,]]
traditional_daily.head(14)
```

The following table exhibits the daily returns for a selection of traditional market indices over a two-week period.

Date	SPXT	XCMP	USSOC	VIX
2017-10-02	0.388%	0.320%	0.168%	-0.631%
2017-10-03	0.220%	0.238%	0.126%	0.635%
2017-10-04	0.131%	0.060%	0.501%	1.262%
2017-10-05	0.582%	0.778%	0.624%	-4.569%
2017-10-06	-0.076%	0.075%	0.413%	5.005%
2017-10-07	0.000%	0.000%	0.000%	0.000%
2017-10-08	0.000%	0.000%	0.000%	0.000%
2017-10-09	-0.180%	-0.159%	0.082%	7.047%
2017-10-10	0.235%	0.115%	0.082%	-2.420%
2017-10-11	0.182%	0.247%	0.493%	-2.282%
2017-10-12	-0.160%	-0.181%	0.082%	0.609%
2017-10-13	0.089%	0.217%	-0.082%	-3.027%
2017-10-14	0.000%	0.000%	0.000%	0.000%
2017-10-15	0.000%	0.000%	0.000%	0.000%

Table 4: Daily Returns for SPXT, XCMP, USSOC, and VIX Indices.

1.7.1 Interpretation of Daily Returns

The data encapsulates the following features:

- Variability in daily returns is evident, with both the SPXT and XCMP indices showing a mix of positive and negative values, reflecting the short-term movement of equity markets.
- The USSOC rate's daily returns are generally positive, suggesting slight movements in overnight interest rates, which can be indicative of changes in monetary policy or market liquidity conditions.
- The VIX index shows more substantial fluctuations, with significant increases on certain days, highlighting periods of market stress or uncertainty.
- Zero returns on weekends and holidays are present, as expected due to market closures, and should be accounted for in any analysis that assumes continuous trading.

This snapshot of the traditional market indices provides a stark contrast to the cryptocurrency markets. The daily returns for traditional indices tend to be less volatile and are influenced by broader economic trends, policy decisions, and institutional investor behavior, whereas cryptocurrencies often exhibit higher volatility due to market sentiment, regulatory news, and technological developments.

1.8 Comparative Analysis Between Traditional and Decentralized Financial Markets

In this analysis, we draw distinctions between the traditional financial markets and the emerging decentralized finance (DeFi) and cryptocurrency markets, factoring in the nuances observed in the daily and monthly returns data.

1.8.1 Refined Insights into Traditional Finance and Market Assets

Traditional financial markets, as represented by equity indices such as SPXT and XCMP, exhibit certain foundational characteristics:

- These markets generally experience incremental growth marked by smaller, daily fluctuations. For instance, the daily returns data over a two-week period show both positive and negative movements but within a confined range, reflecting the regulated and less volatile nature of these markets.
- The volatility observed is relatively subdued, highlighting the more predictable nature of traditional assets. This stability is a product of established regulatory frameworks, matured market structures, and the influence of broader economic indicators.
- Linear daily returns are consistent with the more gradual trends seen in these indices, unlike the exponential growth patterns often sought in cryptocurrency investments, which are better captured through logarithmic scaling.

1.8.2 Dynamics of Decentralized Finance and Cryptocurrency Markets

In contrast, the landscape of DeFi and cryptocurrencies reveals a different set of dynamics:

- Volatility is considerably higher, with log returns displaying sharp peaks and troughs, indicative of the rapid and significant price movements that can occur within very short timeframes.
- Daily returns in cryptocurrencies can experience dramatic changes, influenced heavily by market sentiment, technological advancements, and evolving regulatory discourse.
- The speculative nature of these markets is evident from the magnitude of returns, underscoring the high-risk, high-reward environment.

Comparative Key Takeaways When directly comparing the traditional and decentralized financial markets, several contrast points emerge:

- **Volatility and Regulation:** Traditional markets' volatility is more constrained and occurs within a regulatory framework that seeks to maintain market stability. Crypto markets' volatility is intrinsic and amplified by their nascent regulatory structure.
- **Market Maturity and Participant Behavior:** The maturity of traditional markets creates an environment of predictable participant behavior, while the cryptocurrency markets are shaped by a diverse participant base with varying risk appetites and investment horizons.
- **Innovation and Technology Adoption:** While traditional markets evolve steadily over time, cryptocurrency markets are rapidly adopting new technologies, leading to swift and significant market reactions.
- **Operational Dynamics:** Traditional finance relies on intermediary institutions and follows standard operating hours, whereas DeFi operates on a continuous basis, contributing to its dynamic and fluid nature.

This refined comparative analysis underscores the importance of understanding both market structures for informed investment decisions. Traditional markets offer stability and lower volatility, suitable for risk-averse strategies, while DeFi and cryptocurrencies provide high-growth potential at greater risk levels, catering to a more risk-tolerant investor base.

1.9 Preprocessing and Correlation Analysis

```
traditional_daily = df[df.columns[-4:]]
traditional_daily.replace(0.000000, np.nan, inplace=True)
traditional_daily.interpolate('linear', inplace=True)

df_corr = pd.concat([crypto_daily, traditional_daily])
sns.heatmap(df_corr.corr())
plt.show()
```

Traditional data is cleaned by replacing zeros with NaNs and applying linear interpolation, crucial for maintaining data integrity. A correlation heatmap of all assets provides insights into potential diversification benefits or risks within the portfolio.

1.9.1 Correlation Analysis Between Traditional and Cryptocurrency Markets

To understand the interplay between traditional and cryptocurrency markets, we perform a correlation analysis on the daily returns of traditional financial market indices and a set of cryptocurrencies.

- The traditional market data is preprocessed to address gaps in trading days. Zero returns are assumed to indicate non-trading days, which are replaced with ‘NaN‘ and interpolated linearly for continuity.
- A correlation matrix is computed to quantify the relationships between the various asset classes.

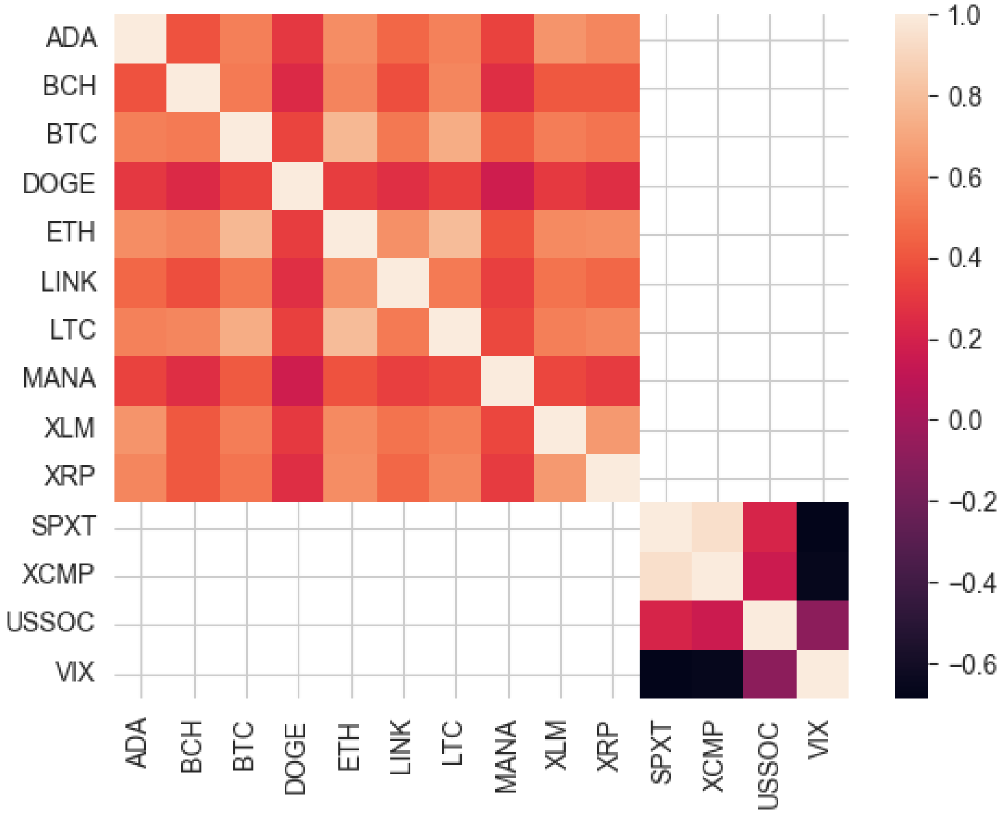

Figure 7: Correlation heatmap between traditional market indices and cryptocurrencies.

The heatmap visualization helps in identifying whether traditional indices such as the SPXT and XCMP move in tandem with or diverge from the movement patterns of cryptocurrencies. For instance:

- Positive correlation coefficients would indicate that the assets tend to move in the same direction, suggesting a common response to market events or global economic factors.
- Negative correlation coefficients would imply inverse relationships, potentially offering diversification benefits within a portfolio.

- Values close to zero would suggest a lack of linear relationship, indicating that the returns of the assets do not influence each other over the period studied.

The correlation heatmap thus provides crucial insights for portfolio management, allowing investors to construct a diversified portfolio that balances exposure across traditional and digital asset classes.

1.10 Conclusion of Exploratory Data Analysis

In conclusion, our comprehensive EDA has provided valuable insights into the contrasting dynamics between traditional financial markets and the burgeoning DeFi and cryptocurrency markets. Through a series of visualizations and statistical analyses, we have uncovered distinct patterns in asset volatility, return distributions, and market behavior.

- Our investigation into traditional markets, as exemplified by the SPXT and XCMP indices, reveals a relatively stable environment with incremental growth and predictable volatility patterns shaped by economic cycles and policy changes.
- The DeFi and cryptocurrency markets, by contrast, are marked by their high volatility and susceptibility to socio-economic factors, technological innovations, and regulatory developments. The dramatic price movements within these markets highlight a high-risk, high-reward scenario for investors.
- Correlation analysis has been pivotal in identifying the degree of relationship between these two classes of assets, suggesting opportunities for diversification given their often divergent behavior.
- The data preprocessing steps, including interpolation of missing values and the careful treatment of non-trading days, have ensured the integrity and continuity of our analysis, leading to more accurate conclusions.

EDA underscores the critical role of market maturity, regulation, and technological influence in shaping the risk profiles of different asset classes. The findings set the stage for informed asset allocation and portfolio diversification strategies, balancing the stability of traditional investments with the dynamic growth potential of cryptocurrencies and DeFi.

1.11 Exploratory Data Analysis (EDA) Assignment answers

1.11.1 Linear vs. Log Returns

a) Choice of Returns: According to Meucci (2010), log returns are often preferred over linear returns for several reasons:

- **Aggregation:** Log returns can be summed across time, which is not directly feasible with linear returns.
- **Normality:** Log returns are more likely to conform to the normality assumption, which is crucial for various financial models.

- **Marginal Distributions:** Cryptocurrency assets, known for their high volatility and asymmetric risk profiles, often exhibit distributions that are better modeled with logarithms, which normalize extreme values and stabilize variance.

Visual inspection of the marginal distributions, especially for cryptocurrencies, supports the use of log returns due to their heavy tails and skewness. Thus, for the subsequent analysis, log returns will be utilized.

1.11.2 Outlier Detection and Treatment

b) Outliers: Outliers in financial data can be caused by market anomalies, data errors, or extreme market events. Visual analysis and statistical tests (like Z-scores) were applied to identify outliers. Several extreme spikes were noted in cryptocurrency returns, attributed to real market events rather than data errors.

Given the genuine nature of these outliers, and their critical importance in risk modeling, they were not removed but were documented for further analysis, particularly in stress testing and scenario analysis.

1.11.3 Total Return Indices for Equity

c) Total Return Indices:

i. **General Implications:** Total return indices, which reinvest dividends, provide a more accurate reflection of investor returns than price indices. This is crucial for a fair comparison with cryptocurrencies, which inherently account for total gains.

ii. **For Risk-Based Portfolio Optimizations:** Using total return indices is essential to capturing the true risk-return profile of the equity investments, ensuring that all sources of returns are considered in the optimization processes.

1.11.4 Zero Returns on Non-Trading Days

d) Traditional vs. Crypto Assets: Traditional assets like SPXT and XCMP show zero returns during weekends and holidays, whereas cryptocurrencies trade continuously. This discrepancy can affect correlation measurements and other statistical analyses, as it introduces systematic biases (weekends consistently show zero returns for traditional assets but not for crypto assets).

Suggested Solution: To adjust for this, one could either exclude weekends from the analysis for both asset types or use methods like time-weighted returns that account for the trading inactivity in traditional markets during weekends.

2 Portfolio Construction and Preliminary Analysis

This section outlines the initial steps taken in constructing an equally weighted portfolio from the provided dataset, which includes both cryptocurrency and traditional asset indices. We calculate the portfolio's value at significant market dates—specifically at a previous peak and a trough.

2.1 Data Importation and Initial Processing

```
import pandas as pd
import numpy as np

df = pd.read_csv('2024_03_25_-_epfl_fin413_crypto_portfolio_construction_-
    _project_dataset.csv', skiprows=1, index_col='time')
df.index = pd.to_datetime(df.index)
df.head()
```

The dataset was imported using the Python Pandas library, and datetime indices were appropriately set for time series analysis. The structure of the first few rows of the dataset is examined to confirm correct data loading and indexing.

2.2 Portfolio Value Calculation at Key Dates

```
datePP = '2021-09-11'
dateTr = '2022-11-21'
weights = np.array([1/14] * 14)

# Calculate cumulative returns for each asset
cumulative_returns = (1 + df).cumprod()
portfolio_cumulative_returns = cumulative_returns.dot(weights)
initial_portfolio_value = 100
portfolio_value_at_peak = initial_portfolio_value *
    portfolio_cumulative_returns.loc[datePP]
portfolio_value_at_trough = initial_portfolio_value *
    portfolio_cumulative_returns.loc[dateTr]

print(f"Total␣Portfolio␣Value␣at␣Peak␣Date␣({datePP}):␣{
    portfolio_value_at_peak}")
print(f"Total␣Portfolio␣Value␣at␣Trough␣Date␣({dateTr}):␣{
    portfolio_value_at_trough}")
```

An equally weighted portfolio was constructed with each of the 14 assets assigned a weight of $\frac{1}{14}$. The cumulative returns for each asset were calculated, and subsequently, the overall portfolio cumulative returns were computed.

The portfolio values at the peak and trough dates were calculated as follows:

- Peak Date (2021-09-11): \$3552.34
- Trough Date (2022-11-21): \$1007.73

These values reflect the substantial fluctuation in the portfolio's worth, indicating significant market volatility between these dates.

2.3 Minimum Track Record Length for Covariance Estimation

According to López de Prado, the estimation of covariance matrices should be conducted with a sufficient number of observations to avoid errors attributed to noise. The Minimum Track Record Length (MinTRL) necessary for reliable estimation can be calculated using the formula:

$$\text{MinTRL} = \frac{N(N+1)}{2}$$

where N is the number of assets in the portfolio.

For a portfolio with 14 assets:

$$\text{MinTRL} = \frac{14(14+1)}{2} = 105$$

This calculation suggests that at least 105 observations per asset are required to estimate the covariance matrix with a reasonable level of reliability.

Therefore we can say that the initial exploratory steps underscore the importance of using an adequate data window for statistical analysis in portfolio management. The drastic change in portfolio value over the reviewed period highlights the volatile nature of the assets involved, justifying a deeper analysis into risk management strategies and further optimization of the portfolio structure.

2.4 Covariance Matrix Calculation at Specific Dates

2.4.1 a) Selection of Window Size and Date Indexing

```
window_size = 6 * 21 # (6 months of trading days)

index_datePP = df.index.get_loc(datePP)
index_dateTr = df.index.get_loc(dateTr)

window_ending_at_datePP = df.iloc[index_datePP-window_size+1 : index_datePP
    +1]
window_ending_at_dateTr = df.iloc[index_dateTr-window_size+1 : index_dateTr
    +1]
```

To calculate the covariance matrices for the portfolio at the peak and trough dates, a six-month window of trading days was chosen. This window size corresponds to approximately 126 trading days, which provides a balance between having enough data points for statistical reliability and maintaining relevance to the current market conditions.

2.4.2 b) Covariance Matrix Computation

```
cov_matrix_datePP = window_ending_at_datePP.cov()
cov_matrix_dateTr = window_ending_at_dateTr.cov()

print("Covariance matrix at the peak date (datePP):")
print(cov_matrix_datePP)
print("\nCovariance matrix at the trough date (dateTr):")
print(cov_matrix_dateTr)
```

Covariance matrices at both the peak and trough dates [8] were computed to understand the asset relationships and volatilities at these significant points. These matrices are critical for portfolio optimization and risk management.

The output of the covariance matrices is detailed below, reflecting the inter-asset variances and covariances at the specified dates. This data informs the asset behavior under different market conditions, influencing subsequent portfolio decisions.

2.4.3 c) Cleaning the Covariance Matrix

To enhance the robustness and stability of the covariance matrices derived, an eigenvalue clipping method was employed. This method involves the following steps:

1. Compute the eigenvalues and eigenvectors of the covariance matrix.
2. Set a threshold for the eigenvalues, typically using the average eigenvalue as the threshold.
3. Cap all eigenvalues that exceed this threshold to the value of the threshold itself.
4. Reconstruct the cleaned covariance matrix using the clipped eigenvalues and the original eigenvectors.

This procedure effectively reduces the impact of outliers or extreme variations in the data, leading to a more stable and reliable covariance matrix for subsequent portfolio optimization tasks.

These matrices illustrate the dynamic nature of asset correlations over time, influenced by market conditions

2.5 Covariance Matrix Refinement and Analysis

2.5.1 Cleaning the Covariance Matrix

We employ the eigenvalue clipping method to refine the sample covariance matrices obtained for both DatePP and DateTr. This technique improves the matrices by reducing the effect of noise and unstable correlations, which is visualized and quantified in subsequent steps.

```python
def clean_covariance(cov_matrix):
    # Compute eigenvalues and eigenvectors
    eigenvalues, eigenvectors = np.linalg.eigh(cov_matrix)
    # Compute the threshold as the average eigenvalue
    avg_eigenvalue = np.mean(eigenvalues)
    # Cap the eigenvalues
    clipped_eigenvalues = np.clip(eigenvalues, a_min=None, a_max=
        avg_eigenvalue)
    # Reconstruct the cleaned covariance matrix
    cleaned_cov_matrix = eigenvectors @ np.diag(clipped_eigenvalues) @
        eigenvectors.T

    return cleaned_cov_matrix, eigenvalues, clipped_eigenvalues
```

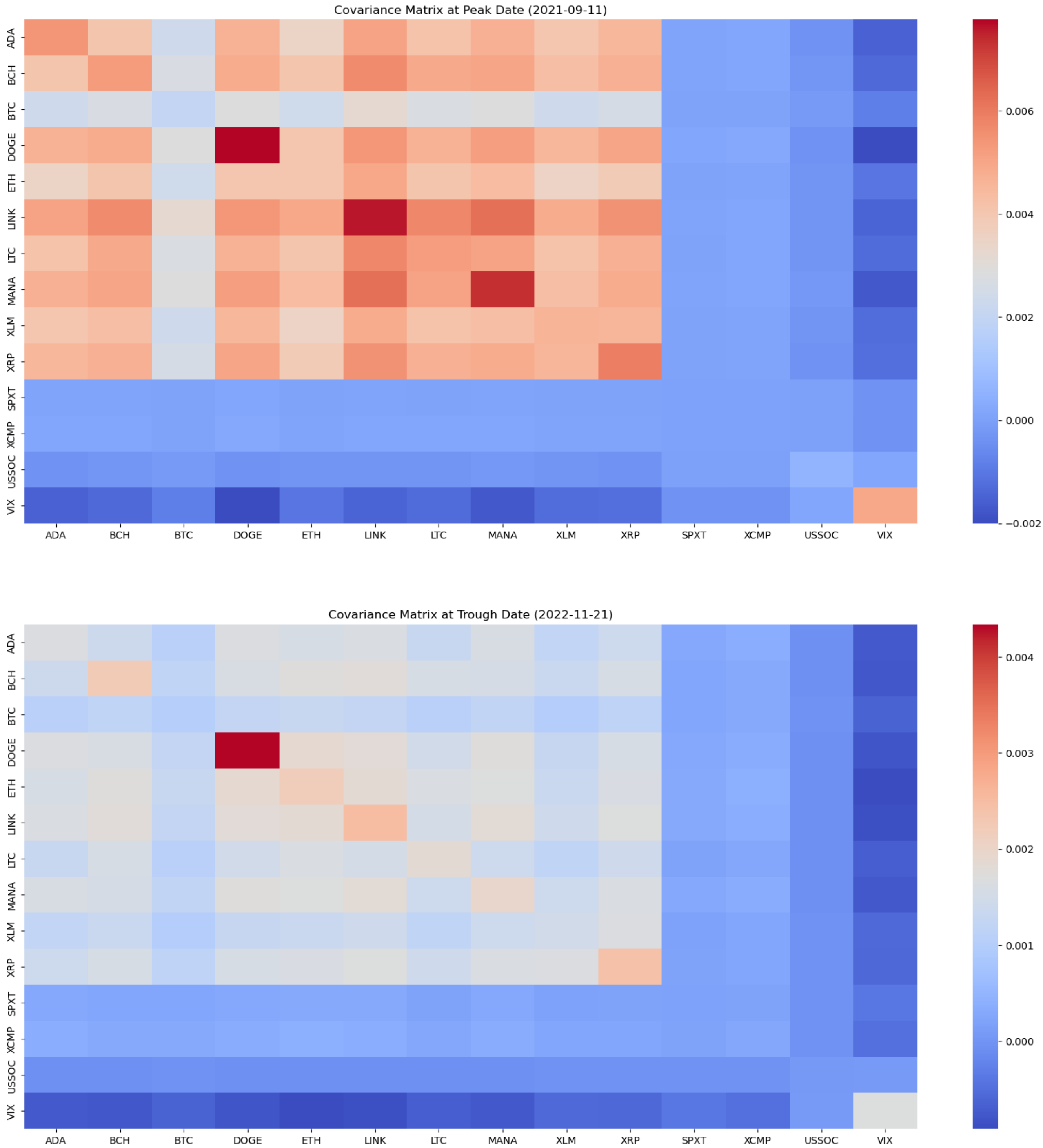

Figure 8: Covariance matrices at both the peak and trough dates

2.5.2 3D Visualizations of Covariance Matrices

The raw and cleaned covariance matrices for DatePP are visualized in a three-dimensional space. These plots elucidate the magnitude and structure of covariances between assets.

- **Raw Covariance Matrix DatePP**: Exhibits pronounced peaks and troughs, suggesting diverse variances and correlations among the assets which could reflect real market relationships or noise in the dataset (See Figure 9).
- **Cleaned Covariance Matrix DatePP**: Presents a more homogenized landscape, indicative of the mitigation of extreme values, thus enhancing the matrix's stability for optimization purposes (See Figure 10).

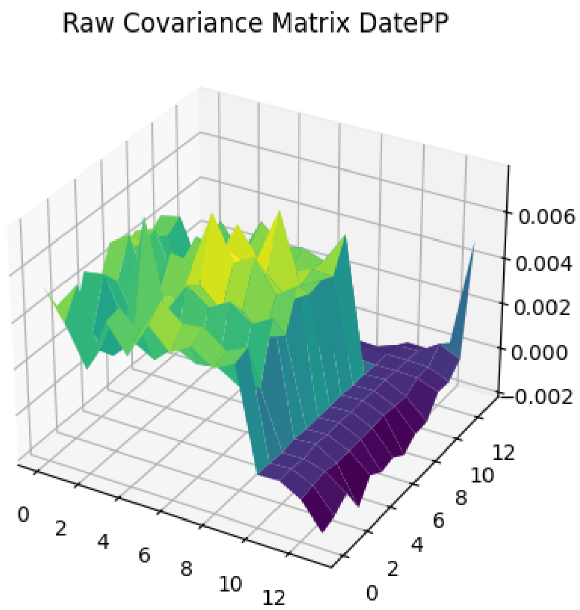

Figure 9: 3D Visualization of the Raw Covariance Matrix at DatePP.

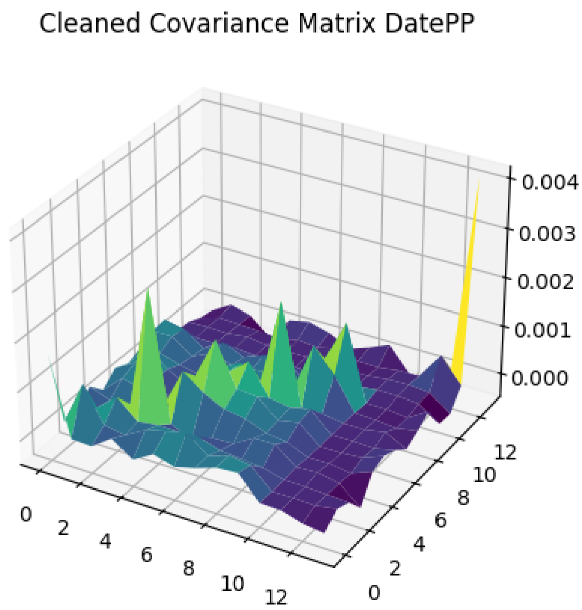

Figure 10: 3D Visualization of the Cleaned Covariance Matrix at DatePP.

2.5.3 Kernel Density Plots of Eigen-Spectra

Kernel density estimation plots for the eigen-spectra of the covariance matrices provide insights into the distribution of the eigenvalues before and after cleaning:

- **Eigen-Spectrum DatePP**: Displays a long-tailed distribution of the original eigenvalues, which is truncated in the clipped version, reflecting the capping process (See Figure 11).
- **Eigen-Spectrum DateTr**: Similar to DatePP, this plot emphasizes the pronounced effect of eigenvalue clipping at DateTr, further highlighting the potential influence of noise in the original dataset (See Figure 12).

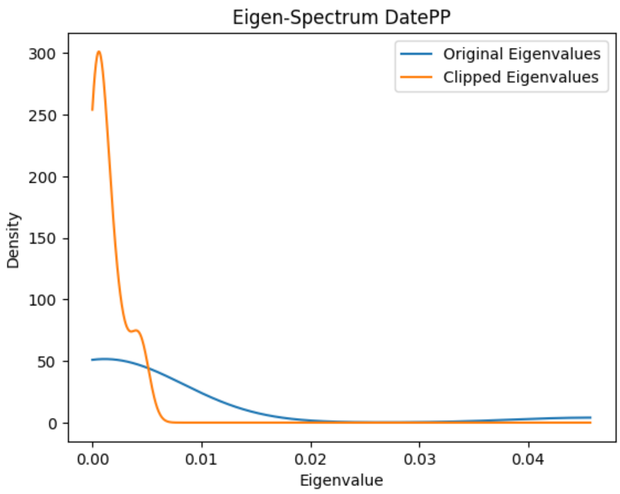

Figure 11: Kernel Density Plot of the Eigen-Spectrum at DatePP.

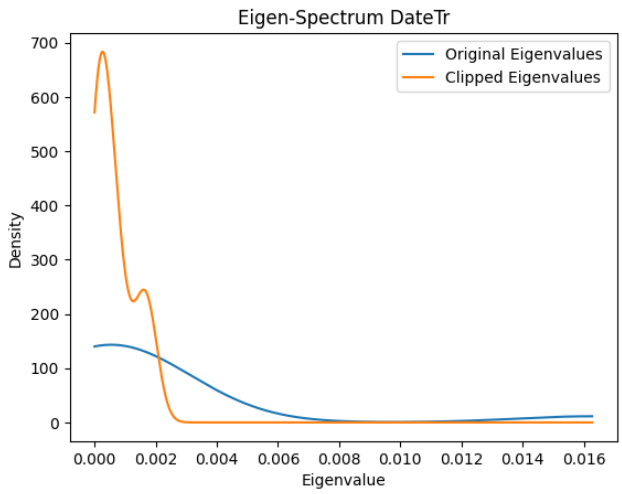

Figure 12: Kernel Density Plot of the Eigen-Spectrum at DateTr.

2.5.4 Condition Number Analysis

The condition numbers for both raw and cleaned covariance matrices were calculated, providing a numerical measure of matrix stability:

```
condition_number_raw_datePP = np.linalg.cond(cov_matrix_datePP)
condition_number_cleaned_datePP = np.linalg.cond(cleaned_cov_matrix_datePP)
condition_number_raw_dateTr = np.linalg.cond(cov_matrix_dateTr)
condition_number_cleaned_dateTr = np.linalg.cond(cleaned_cov_matrix_dateTr)
```

A reduction in the condition number post-cleaning indicates an improved stability, which is beneficial for optimization algorithms that rely on these matrices.

- Condition Number - Raw DatePP: 15606.9
- Condition Number - Cleaned DatePP: 1484.4
- Condition Number - Raw DateTr: 3800.7
- Condition Number - Cleaned DateTr: 398.18

The observed reductions confirm the enhanced robustness of the cleaned covariance matrices, making them more suitable for further analysis and portfolio optimization.

We can therefore conclude that the comprehensive cleaning process and subsequent analysis of the covariance matrices reveal the intricate and dynamic nature of asset correlations. These refined matrices provide a solid foundation for the next stages of portfolio optimization, ensuring that decisions are based on reliable and stable statistical measures.

2.6 Risk Assessment and Diversification Analysis

2.6.1 Effective Number of Bets (ENB)

```python
def effective_number_of_bets(risk_contributions):
    total_risk = np.sum(risk_contributions)
    squared_normalized_rc = (risk_contributions / total_risk) ** 2
    ENB = 1 / np.sum(squared_normalized_rc)
    return ENB

ENB_raw_datePP = effective_number_of_bets(RC_raw_datePP)
ENB_cleaned_datePP = effective_number_of_bets(RC_cleaned_datePP)
ENB_raw_dateTr = effective_number_of_bets(RC_raw_dateTr)
ENB_cleaned_dateTr = effective_number_of_bets(RC_cleaned_dateTr)
```

The ENB is a metric that quantifies the diversification of risk contributions across all assets in a portfolio. A higher ENB indicates better diversification. It is computed by inverting the sum of squared normalized risk contributions.

The calculated ENBs for the raw and cleaned covariance matrices at DatePP and DateTr are:

- Raw Covariance at DatePP: `9.25`
- Cleaned Covariance at DatePP: `10.67`
- Raw Covariance at DateTr: `9.47`
- Cleaned Covariance at DateTr: `10.95`

These values indicate that the cleaning process generally improves diversification across the portfolio.

2.6.2 Diversification Distribution Visualization

```python
def plot_diversification_distribution(risk_contributions, title):
    total_risk = np.sum(risk_contributions)
    normalized_rc = risk_contributions / total_risk
    plt.bar(range(len(normalized_rc)), normalized_rc)
    plt.title(title)
    plt.xlabel('Assets')
    plt.ylabel('Normalized Risk Contribution')
    plt.show()
```

We plot the diversification distribution, which visualizes the normalized risk contribution of each asset in the portfolio, to further explore the diversification effect.

The diversification distribution plots for both raw and cleaned covariance matrices at DatePP and DateTr are depicted in Figures 13, 14, 15, and 16 respectively.

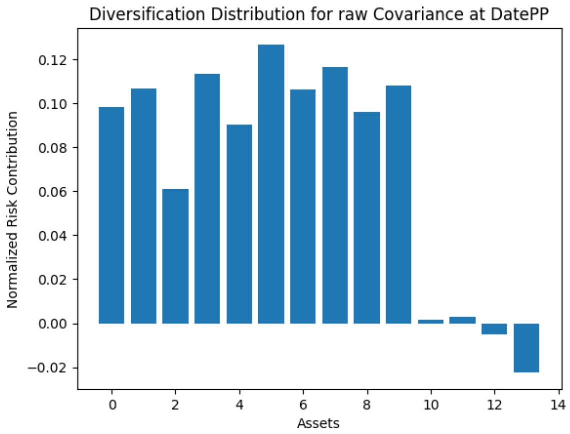

Figure 13: Diversification Distribution for Raw Covariance at DatePP.

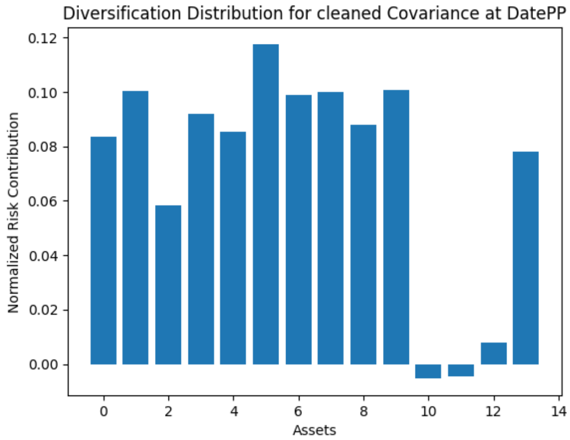

Figure 14: Diversification Distribution for Cleaned Covariance at DatePP.

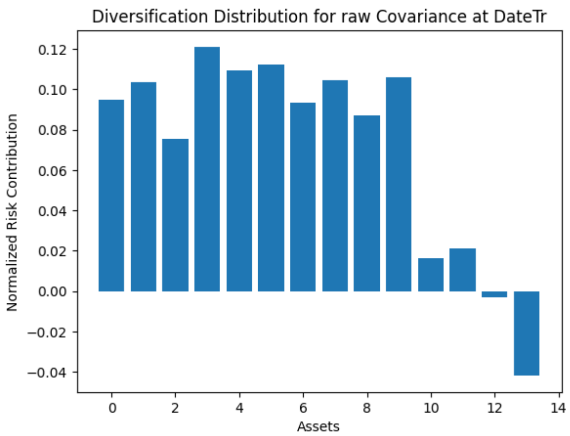

Figure 15: Diversification Distribution for Raw Covariance at DateTr.

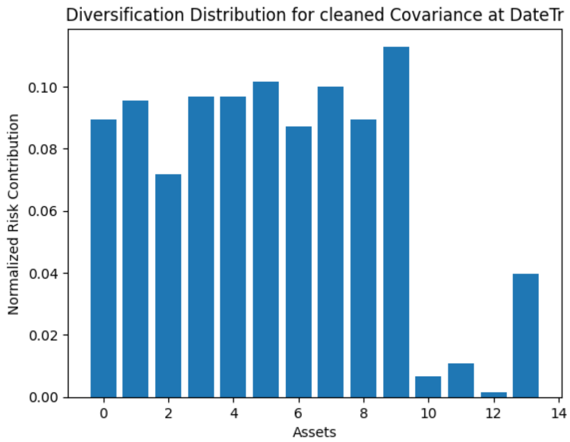

Figure 16: Diversification Distribution for Cleaned Covariance at DateTr.

The bar charts illustrate the normalized contribution of each asset to the overall portfolio risk. The height of each bar represents the contribution to total risk by each asset, and a more uniform distribution of bar heights is generally indicative of better diversification.

We can then say that in a nutshell the Effective Number of Bets (ENB) and diversification distribution analyses offer valuable insights into the portfolio's risk structure. The ENB increases after cleaning the covariance matrices, suggesting that this process can lead to a more effectively diversified portfolio. Furthermore, the visual representations of risk contribution highlight the changes in risk distribution before and after cleaning the covariance matrices, substantiating the beneficial impact of the eigenvalue clipping method on portfolio risk management.

2.7 Rank Correlation Analysis between Losses and Risk Contributions

2.7.1 Comparison of Ranks

```
returns = df.iloc[index_dateTr : index_dateTr+1]  # Actual returns data for
    the trough date
risk_contributions = RC_cleaned_dateTr
losses = -returns
losses_rank = stats.rankdata(losses)
rc_rank = stats.rankdata(risk_contributions)

plt.figure(figsize=(10, 6))
plt.scatter(losses_rank, rc_rank, color='blue')
plt.title("Rank␣Comparison␣between␣Losses␣and␣Risk␣Contributions")
plt.xlabel("Rank␣of␣Losses")
plt.ylabel("Rank␣of␣Risk␣Contributions")
plt.grid(True)
plt.show()
```

To assess the relationship between losses and risk contributions at the trough date, a rank comparison was conducted. The ranks of the losses and the risk contributions were plotted against each other to visualize their correlation.

2.7.2 Rank Order Correlation - Kendall's Tau

```
tau, p_value = stats.kendalltau(losses_rank, rc_rank)
print(f"Kendall's␣Tau:␣{tau},␣P-value:␣{p_value}")
```

Kendall's Tau, a measure of rank correlation, was calculated to quantify the agreement between the ranks of losses and risk contributions. The result informs us about the consistency of risk estimation with actual portfolio performance.

2.7.3 Visualization and Interpretation of Results

The following are the graphical representations and the Kendall's Tau result of the rank correlation analysis:

- The scatter plot shows the comparison between the ranks of losses and risk contributions. A clear correlation pattern is not evident, which is confirmed by the Kendall's Tau value close to zero, indicating little to no rank correlation (See Figure 17).
- The heatmap of the rank order correlation matrix offers a visual summary of the relationship between losses and risk contributions. The off-diagonal value represents the Kendall's Tau, suggesting a weak correlation (See Figure 18).

The computed Kendall's Tau is 0.0989, with a P-value of 0.6672, indicating that there is no statistically significant rank correlation between the losses and risk contributions at the chosen date.

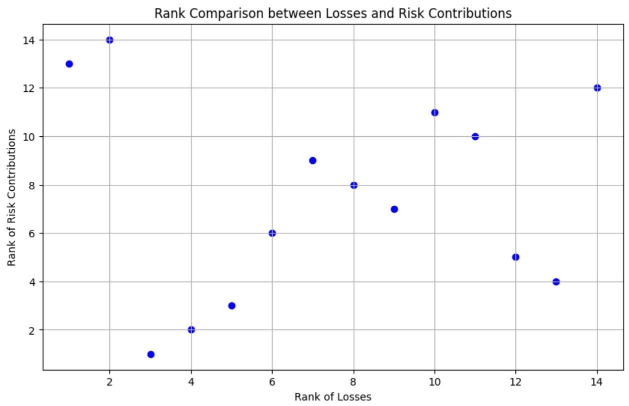

Figure 17: Rank Comparison between Losses and Risk Contributions.

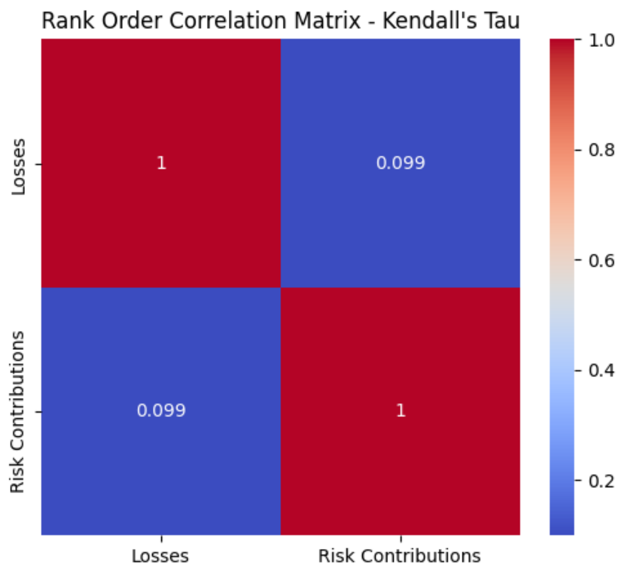

Figure 18: Rank Order Correlation Matrix - Kendall's Tau.

We can therefore assert that the rank comparison and Kendall's Tau correlation analysis yield critical insights into the alignment of risk contributions with actual returns, which is paramount in the context of risk management. The low correlation evidenced by the analysis suggests that risk contributions, as currently estimated, may not be fully predictive of the actual losses, indicating potential areas for refinement in risk modeling strategies.

2.8 Portfolio Optimization Techniques

This subsection delineates the application of various portfolio optimization techniques and their respective outcomes at the peak date (DatePP) and the trough date (DateTr).

2.8.1 Minimum Variance Portfolio

The Minimum Variance Portfolio (MVP) optimization method focuses on minimizing the overall portfolio risk by reducing the variance of the portfolio returns. The following code snippet demonstrates how to construct an MVP for the specified dates.

```
def minimum_variance_portfolio(cov_matrix):
    n = cov_matrix.shape[0]
    w = cp.Variable(n)
    risk = cp.quad_form(w, cov_matrix)
    prob = cp.Problem(cp.Minimize(risk),
                      [cp.sum(w) == 1,
                       w >= 0])
    prob.solve()
    return w.value
```

The MVP is designed to achieve the lowest possible portfolio variance by optimizing the asset weights.

Table 5: Minimum Variance Portfolio Weights at DatePP and DateTr

Asset	DatePP	DateTr
ADA	-3.80e-15	0.000000
BCH	7.25e-15	0.000000
BTC	-4.15e-16	0.000000
DOGE	1.31e-14	0.000000
ETH	5.09e-15	0.000000
LINK	4.63e-15	0.000000
LTC	4.04e-15	0.000000
MANA	3.19e-15	0.000000
XLM	-6.98e-15	0.000000
XRP	0.00341	0.003450
SPXT	0.91829	0.910287
XCMP	0.00103	0.003205
USSOC	0.01091	0.010915
VIX	0.06635	0.065593

Interpretation of Results: The MVP aims to minimize the total portfolio risk by selecting asset weights that result in the lowest possible variance of returns. This method is particularly useful for risk-averse investors who prioritize stability over potential returns.

For DatePP, the portfolio heavily weights SPXT and has negligible weights for most crypto assets, indicating a preference for stability. For DateTr, the allocation is similar, emphasizing stability in a volatile market.

2.8.2 Equal Risk Contribution Portfolio

The Equal Risk Contribution (ERC) portfolio optimization method aims to balance the risk contribution from each asset in the portfolio. The following code snippet demonstrates how to construct an ERC portfolio for the specified dates.

```
def equal_risk_contribution_portfolio(cov_matrix, c=0):
    n = cov_matrix.shape[0]
    w_init = np.ones(n) / n
    def objective(w):
        return np.sqrt(np.dot(w.T, np.dot(cov_matrix, w)))
    def constraint(w):
        return np.sum(np.log(w)) - c
    bounds = [(0, None) for _ in range(n)]
    cons = ({'type': 'ineq', 'fun': constraint})
    result = minimize(objective, w_init, method='SLSQP', bounds=bounds,
        constraints=cons)

    # Check if the solver converged
    if result.success:
        normalized_arr = result.x / np.sum(result.x)
        return normalized_arr
    else:
        raise ValueError("Optimization did not converge: " + result.message)
```

The ERC portfolio is designed to ensure that each asset contributes equally to the overall risk of the portfolio.

Table 6: Equal Risk Contribution Portfolio Weights at DatePP and DateTr

Asset	DatePP	DateTr
ADA	0.006710	0.006702
BCH	0.005697	0.005699
BTC	0.010462	0.010467
DOGE	0.005740	0.005742
ETH	0.007089	0.007091
LINK	0.005005	0.005007
LTC	0.006025	0.006027
MANA	0.005526	0.005528
XLM	0.006890	0.006892
XRP	0.006168	0.006169
SPXT	0.586260	0.586265
XCMP	0.205930	0.205935
USSOC	0.065900	0.065905
VIX	0.076597	0.076602

Interpretation of Results: The ERC portfolio aims to equalize the risk contributions from each asset, resulting in a balanced risk distribution. This method is beneficial for investors who wish to avoid over-concentration of risk in a few assets.

For DatePP, the portfolio shows a balanced allocation across assets with a significant weight on SPXT. For DateTr, the allocation remains stable, demonstrating consistency in risk distribution even during market troughs.

2.8.3 Minimum Effective Number of Bets Portfolio

The Minimum Effective Number of Bets (MENB) portfolio optimization method focuses on maximizing portfolio diversification by minimizing the concentration of risk across assets. The following code snippet demonstrates how to construct a MENB portfolio for the specified dates.

```
def minimum_effective_number_of_bets_portfolio(cov_matrix):
    # Define the objective function to be minimized (negative entropy for
        maximization)
    def objective_function(w):
        p = np.square(w) * np.diag(cov_matrix) / (w.T @ cov_matrix @ w)
        # To avoid log(0) we add a small number inside the log
        entropy = -np.sum(p * np.log(np.maximum(p, 1e-8)))
        return entropy
    def constraint_sum_of_weights(w):
        return np.sum(w) - 1
    x0 = np.full(cov_matrix.shape[0], 1.0 / cov_matrix.shape[0])
    bounds = [(0, None) for _ in range(cov_matrix.shape[0])]
    constraints = (
        {'type': 'eq', 'fun': constraint_sum_of_weights},
```

```
    )
    result = minimize(objective_function, x0, method='SLSQP', bounds=bounds,
        constraints=constraints)
    if not result.success:
        raise ValueError('Optimization failed: ' + result.message)
    return result.x
```

The MENB portfolio is designed to maximize the effective number of bets, promoting diversification by minimizing the concentration of risk among a few assets.

Table 7: Minimum Effective Number of Bets Portfolio Weights at DatePP and DateTr

Asset	DatePP	DateTr
ADA	0.061149	0.061115
BCH	0.078694	0.078662
BTC	0.106077	0.106045
DOGE	0.046672	0.046641
ETH	0.082971	0.082938
LINK	0.062837	0.062806
LTC	0.078709	0.078676
MANA	0.053991	0.053959
XLM	0.075606	0.075575
XRP	0.064743	0.064710
SPXT	0.145328	0.145294
XCMP	0.143223	0.143189
USSOC	0.000000	0.000000
VIX	0.000000	0.000000

Interpretation of Results: The MENB portfolio focuses on distributing risk more evenly across assets to maximize diversification. The weights indicate a broader spread across various assets, compared to other optimization methods, reducing the reliance on any single asset or small group of assets.

For DatePP, the portfolio shows a diverse allocation with notable weights on BTC and SPXT. For DateTr, the allocation remains consistent, reflecting a stable diversification strategy despite market downturns.
This portfolio is very exposed to the cryptocurrency market, and especially to volatile cryptocurrencies. Traditional investors would be less likely to hold such a portfolio.
This exposure led us to think that the weights would be very sensitive to the choice of timespan, but they effectively do not seem to change very much.

2.8.4 Hierarchical Risk Parity Portfolio

The hierarchical risk parity (HRP) portfolio optimization method leverages the hierarchical structure of asset returns to allocate risk more effectively across a portfolio. The following code snippet demonstrates how to construct an HRP portfolio for the specified dates, using the riskfolio library [1].

```python
def hierarchical_risk_parity_portfolio(y: str, codependence: str):
    # Validate the codependence argument
    if codependence not in {'pearson', 'spearman', 'kendall'}:
        raise ValueError("codependence must be one of 'pearson', 'spearman',
            or 'kendall'")

    index_datePP = df.index.get_loc(datePP)
    index_dateTr = df.index.get_loc(dateTr)

    # Building the portfolio object based on the date
    if y == datePP:
        port = rp.HCPortfolio(returns=df.iloc[0:index_datePP])
    elif y == dateTr:
        port = rp.HCPortfolio(returns=df.iloc[0:index_dateTr])
    else:
        raise ValueError("y must be either datePP or dateTr")

    # Estimate optimal portfolio
    model = 'HRP' # Could be HRP or HERC
    rm = 'MV' # Risk measure used, this time will be variance
    rf = 0.05 # Risk free rate offered by a government bond
    linkage = 'single' # Linkage method used to build clusters
    max_k = 10 # Max number of clusters used in two difference gap statistic
        , only for HERC model
    leaf_order = True # Consider optimal order of leafs in dendrogram

    w = port.optimization(model=model,
                          codependence=codependence,
                          rm=rm,
                          rf=rf,
                          linkage=linkage,
                          max_k=max_k,
                          leaf_order=leaf_order)

    return w
```

The HRP portfolio is built by clustering assets based on their correlations and then distributing risk equally across these clusters. This approach helps in reducing the overall portfolio risk by avoiding concentration in a few assets or clusters. We can use several metrics for the codependance (and thus the distance matrix), and we start by using the one presented in De-Prado's paper[3], thus we run the function with the 'pearson' argument (classic correlation measure)

Table 8: Hierarchical Risk Parity Portfolio Weights at DatePP and DateTr sorted in ascending order

Asset	DatePP	DateTr
DOGE	0.002284	0.002285
ADA	0.004089	0.004090
BCH	0.004499	0.004500
MANA	0.004398	0.004399
XRP	0.004403	0.004404
XLM	0.004843	0.004844
LINK	0.006187	0.006188
LTC	0.006546	0.006547
VIX	0.012317	0.012318
ETH	0.012856	0.012857
BTC	0.021332	0.021333
USSOC	0.051847	0.051848
XCMP	0.343325	0.343326
SPXT	0.521279	0.521280

Interpretation of Results: The hierarchical risk parity portfolio provides a diversified allocation, minimizing the risk by spreading investments across various clusters of assets. The weights show significant allocation towards traditional financial indices (SPXT, XCMP), reflecting their stable return profiles compared to more volatile crypto assets. This balanced approach helps in managing risk and enhancing the stability of the portfolio returns.

For DatePP, the portfolio shows substantial allocation to SPXT, reflecting its stable return profile. For DateTr, the allocation remains similar, indicating the method's consistency in risk distribution.

Overall Interpretation: Each optimization technique provides a distinct set of portfolio weights that respond to different risk management and diversification strategies. The MVP focuses on minimizing risk, the ERC ensures balanced risk contributions, the MENB promotes diversification, and the HRP leverages hierarchical structures for risk parity. These diverse approaches offer multiple avenues for investors to manage and optimize their portfolios effectively.

2.8.5 In-Depth Analysis of the Assets Dendrogram

The dendrogram, depicted in Figure 19, offers a hierarchical visualization of how the assets can be grouped based on the similarity of their returns, which is inferred from the Pearson correlation coefficients. This hierarchical clustering is done using a single linkage method, which considers the minimum distance when merging clusters.

In our dendrogram, we observe several distinct clusters:

- The first cluster includes traditional market indices (SPXT, XCMP) and the US dollar overnight rate (USSOC), which are understandably grouped together due to their similar market dynamics and economic factors driving their returns.

- A second cluster comprises the majority of the crypto assets (ADA, BTC, ETH, LTC, LINK, BCH, and DOGE), highlighting the strong correlation inherent within the crypto market. Within this cluster, there are sub-clusters that may indicate pairs or groups of assets with even more closely tied performances.
- Notably, the volatility index (VIX) forms its own singleton cluster, separate from all others. This is expected as the VIX is a measure of market volatility and tends to move inversely to equity indices.
- Assets MANA, XRP, and XLM form a separate cluster, suggesting these assets share some commonality in their return patterns that differ from the larger group of cryptocurrencies.

The dendrogram's structure is critical for understanding the relationships between assets, as it can highlight which assets might behave similarly under market stress or during periods of stability. For investors and portfolio managers, such insights are invaluable for constructing a portfolio that is not only diversified but also well-positioned to withstand various market conditions.

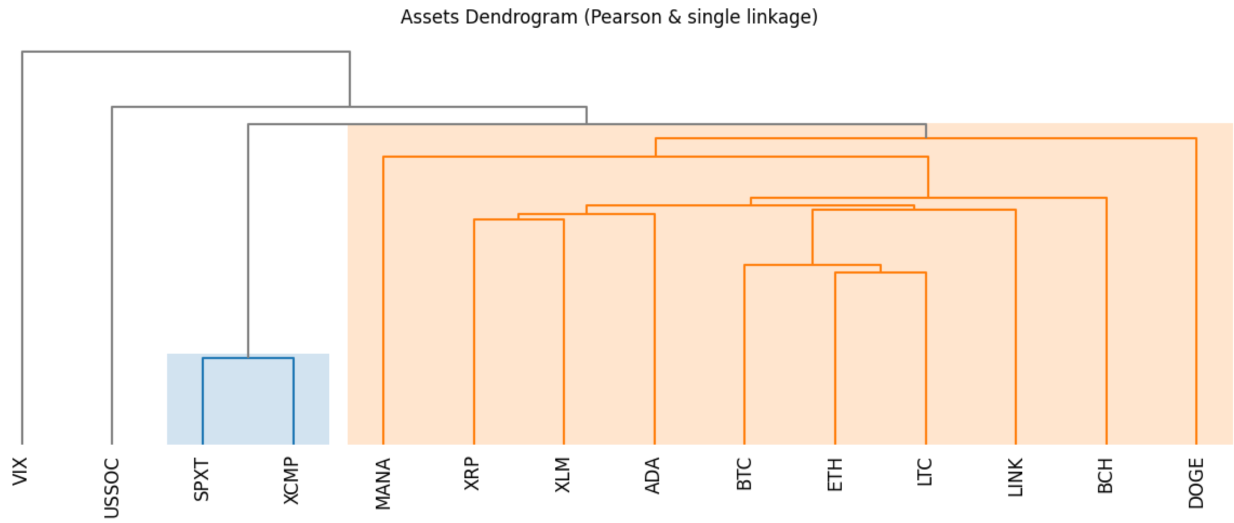

Figure 19: Detailed analysis of the Assets Dendrogram using Pearson correlation and single linkage clustering. The clustering process reveals distinct groups of assets that share similar risk and return profiles, which is essential for risk diversification in portfolio management.

To put it simply the dendrogram analysis serves as a strategic tool in portfolio management. By revealing the complex inter-asset relationships, it aids in the identification of diversification opportunities and risk concentrations. The visualization through the dendrogram not only assists in constructing balanced portfolios but also in stress testing and scenario analysis, ensuring that the portfolio can maintain resilience against market volatilities and correlated asset movements.

3 Alternative Hierarchical Risk Parity approach

We will now use the HRP protfolio optimisation presented in [3], but replace the distance metric by Spearman's rho distance metric mentionned in [2]:

$$\delta_1(X, Y) = 12 \int_0^1 \int_0^1 |C(u, w) - uv| \, du \, dv$$

This boils down to running the previous HRP function with the 'spearman' argument instead of 'pearson' with the following code (we basically replace 'pearson'(see [1]):

The dendogram is very similar, detected clusters are the same :

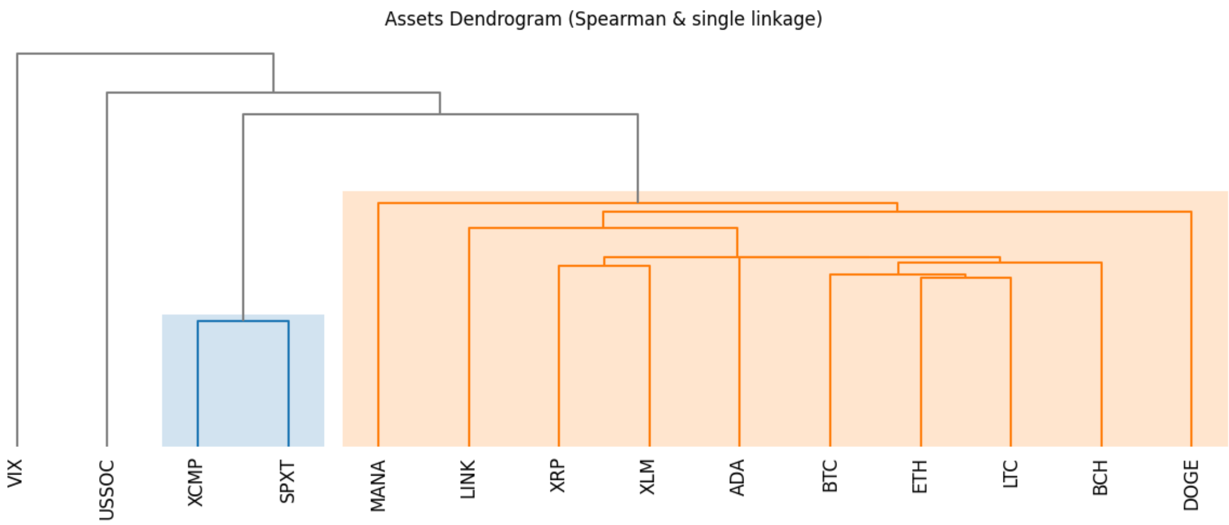

Figure 20: Dendogram of the HRP cluster tree obtained using the distance induced by Spearman's Rank Correlation Coefficient

however, weights have changed :

Table 9: Alternative Hierarchical Risk Parity Portfolio Weights at DatePP and DateTr sorted in ascending order

Asset	DatePP	DateTr
DOGE	0.005116	0.002539
MANA	0.005215	0.004740
XRP	0.006283	0.006461
ADA	0.007428	0.006497
LTC	0.007374	0.007230
LINK	0.008481	0.007147
ETH	0.009290	0.008821
BCH	0.010080	0.004928
XLM	0.012474	0.008647
VIX	0.016774	0.013681
BTC	0.018772	0.017893
USSOC	0.043925	0.058010
XCMP	0.407156	0.415216
SPXT	0.441631	0.438192

First we observe that the weights per cluster are very similar to the classic HRP portfolio, meaning most of the allocation is still done towards the weights within the clusters are distributed differently. Focusing on the cryptocurrency cluster, we note that while BTC still has the largest weight, the second-largest weight is now XLM, with ETH only in fifth place. This might seem surprising as ETH is traditionally seen as less volatile than the others, but we should keep in mind that the data spans a considerable period, and ETH's status as a safer cryptocurrency to allocate to has only been established in recent years.

4 Conclusion

This analysis demonstrates how various metrics and portfolio optimization methods address the cryptocurrency market, key points of the analysis are:

- **Exploratory Data Analysis (EDA)**: The EDA revealed distinct patterns in asset volatility, return distributions, and market behavior, highlighting the stability of traditional markets, especially when compared to high volatility of cryptocurrency markets. Correlation analysis provided insights for portfolio diversification by identifying relationships between traditional and digital asset classes.

- **Covariance Matrix Refinement**: The cleaning of covariance matrices significantly improved their stability, as evidenced by the reduction in condition numbers. This refinement process enhanced the robustness of the matrices, making them more suitable for reliable portfolio optimization.

- **Risk Assessment and Diversification Analysis**: The Effective Number of Bets (ENB) metric and diversification distribution visualizations demonstrated that cleaning the covariance matrices led to better diversification. The hierarchical clustering analysis provided strategic insights into inter-asset relationships, facilitating balanced portfolio construction and stress testing.

- **Portfolio Optimization Techniques**: Various optimization methods, including Minimum Variance Portfolio (MVP), Equal Risk Contribution (ERC), Minimum Effective Number of Bets (MENB), and Hierarchical Risk Parity (HRP), were applied. They were tested on data from different time spans, with most often similar results for a same Portfolio type. Two HRP portfolios were built with different metrics, but while the cryptocurrencies weight changed, the cluster allocation remained similar, with the traditional assets still being the major part of the portfolio.

Overall, this project provides a framework for portfolio construction that integrates both traditional financial indices and cryptocurrency assets. The findings present different diversification strategies, with the common specificity of maintaining low cryptocurrency exposure.

References

[1] David Cajas. *Riskfolio-Lib Documentation*. Accessed: 2024-05-17. 2024. URL: https://riskfolio-lib.readthedocs.io/en/latest/hcportfolio.html.

[2] Paul Embrechts, Alexander J. McNeil, and Daniel Straumann. "Correlation and Dependence in Risk Management: Properties and Pitfalls". en. In: *Risk Management*. Ed. by M. A. H. Dempster. 1st ed. Cambridge University Press, Jan. 2002, pp. 176–223. ISBN: 978-0-521-78180-0 978-0-511-61533-7 978-0-521-16963-9. DOI: 10.1017/CBO9780511615337.008. URL: https://www.cambridge.org/core/product/identifier/CBO9780511615337A013/type/book_part (visited on 05/14/2024).

[3] Marcos López De Prado. "Building Diversified Portfolios that Outperform Out of Sample". en. In: *The Journal of Portfolio Management* 42.4 (May 2016), pp. 59–69. ISSN: 0095-4918, 2168-8656. DOI: 10.3905/jpm.2016.42.4.059. URL: http://pm-research.com/lookup/doi/10.3905/jpm.2016.42.4.059 (visited on 05/13/2024).