Inter IIT TechCamp
-Technical Board IITG

Team- QuantiGram

Pairs Trading and Cointegration Strategy

Quant PS Documentation

Abstract

Pairs Trading is a **market-neutral strategy** where we use **statistical techniques** to identify two stocks that are **historically highly correlated** with each other. When there is a deviation in the price relationship of these stocks, we expect this to mean **reverting** and buying the underperforming stock and simultaneously selling the outperforming one. If our mean-reversion assumption is valid then prices should **converge to long term average** and trade should benefit. However, if the price divergence is not temporary and it is due to other factors, then there is a high risk of losing the money.

This is a kind of **Statistical Arbitrage**. Statistical arbitrage is a class of short-term financial trading strategies that employ mean reversion models involving broadly diversified portfolios of securities (hundreds to thousands).These strategies are supported by substantial mathematical, computational, and trading platforms.

In the Sections that follow, we will start by talking about the steps of **Identification and Selection of Pair of companies (Chapter 1)** whose stocks are closely correlated using **Tests for Correlation**. We will be running these tests and programs on the dataset containing values of the stocks between June 2017 and June 2020. We will follow that up with the discussion of the **Optimal Trading Strategy** and the **Signal Generation Methods(Chapter 2).** This part concludes the training part of the strategy. Now comes the application of the strategy developed, but before that we are going to evaluate the Risks involved with the strategy developed and the **Risk Management techniques** and **methods(Chapter 3).**

Now we move on to the implementation part where we will be using our strategy on the new data (Backtesting) and the backtesting will be done on the data between July 2020 to July 2023 we will be looking **at the Signal generations on the new data** and how to make the **Position Sizing (Chapter 4)** that yields maximum profits. We will then show the **Portfolio PnL (Chapter 5)** to show how our strategy performs with the initial portfolio. Naturally we will follow this up with evaluating certain **Performance Indices** such as Cumulative Returns, the Annualised Sharpe ratio etc. continued in this chapter. We will conclude with certain statements about the practical application of the strategy in the real market.

Contents

Chapter 1

Pair Identification and Selection

This chapter focuses on our strategy of selecting a company from an intensive list of companies whose data has been inferred from Yahoo Finance API[1]. The strategy consists of certain tests and rejection algorithms to find out the best correlated companies which would be used for our future analysis.
The pair chosen so far would carry statistically significant cointegration relationship

But before we do any of that lets look at certain terms and their definitions which we are going to make extensive use of later in pair Identification.

1.1 INTRODUCTORY TERMS AND DEFINITIONS

➢ Cointegration/Correlation Coefficient

The Correlation coefficient is a statistical measure of the strength of the "linear relationship" between two variables. Its values can range from -1 to 1.

- A correlation coefficient of -1 describes a perfect negative, or inverse, correlation, with values in one series rising as those in the other decline, and vice versa.
- A coefficient of 1 shows a perfect positive correlation, or a direct relationship.
- A correlation coefficient of 0 means there is no linear relationship.

➢ Null Hypothesis and Alternative Hypothesis

[1] The Yahoo Finance API is a range of libraries/APIs/methods to obtain historical and real time data for a variety of financial markets and products

- A Null Hypothesis is a type of statistical hypothesis that proposes that no statistical significance exists in a set of given observations. Hypothesis testing is used to assess the credibility of a hypothesis by using sample data. Sometimes referred to simply as the "null", it is represented as H_0.
- The Alternative Hypothesis states exactly the opposite and proposes that statistical significance exists in a set of observations.

➢ Linear Regression[2]

Linear regression analysis is used to predict the value of a variable based on the value of another variable. The variable you want to predict is called the dependent variable. The variable you are using to predict the other variable's value is called the independent variable.

➢ Significance Level

To decide if we have "sufficient" evidence against the null hypothesis to reject it (in favour of the alternative hypothesis), we must first decide upon a *significance level*. The significance level is the probability of rejecting the null hypothesis when the null hypothesis is true and is denoted by α. The 5% significance level is a common choice for statistical tests.

➢ P-Value

The P value is defined as the probability under the assumption of no effect or no difference (null hypothesis), of obtaining a result equal to or more extreme than what was actually observed. The P stands for probability and measures how likely it is that any observed difference between groups is due to chance. Being a probability, P can take any value between 0 and 1.

- Values close to 0 indicate that the observed difference is unlikely to be due to chance,
- Whereas a P value close to 1 suggests no difference between the groups other than due to chance.

[2] Python's statsmodel module is used to implement the Ordinary Least Squares(OLS) method of linear regression.

➢ Stationarity

Stationarity in finance refers to the statistical property of a time series data set where its statistical properties, such as mean and variance, remain constant over time. This concept is crucial in financial analysis and modelling because it helps ensure that historical data can be used to make reasonable predictions about future trends and movements in financial markets.

➢ Spread

Measures of spread (also called measures of *dispersion*) describe how similar or varied the set of observed values are for a particular variable. There are several basic measures of spread used in statistics. Here we will be using the spread between the stock-price points of two companies and given by

$$\text{Spread} = \text{X - Y} \cdot \beta$$

Here

- X is the stock of say company 1 on a given day,
- Similarly Y is the stock price of the other company on the same day.
- β[3] is the Linear Regression slope coefficient for the entire data set over the three years.

➢ Augmented Dicky Fuller(ADF) Test

Statistical tests make strong assumptions about your data. They can only be used to inform the degree to which a null hypothesis can be rejected or fail to be rejected. The result must be interpreted for a given problem to be meaningful. However, they provide a quick check and confirmatory evidence that the time series is stationary or non-stationary. The *Augmented Dickey-Fuller* test is a type of statistical test called a **unit root test**.

In probability theory and statistics, a unit root is a feature of some stochastic processes (such as random walks) that can cause problems in statistical inference involving time series models. In simple terms, the unit root *is* **Non-Stationary** but does not always have a trend component.

ADF test is conducted with the following assumptions:

- *Null Hypothesis (HO)*: Series is non-stationary, or series has a unit root.
- *Alternate Hypothesis(HA)*: Series is stationary, or series has no unit root.

[3] Calculated using OLS method

If the null hypothesis fails to be rejected, this test may provide evidence that the series is non-stationary.

Condition to Reject Null Hypothesis(HO)

- If Test statistic < Critical Value and p-value < 0.05 → *Reject Null Hypothesis(HO), i.e.,* time series does not have a unit root, implying it is stationary having no time dependent structure.

1.2 The Main Idea

How are we going to select?

Assuming mean reversion as discussed above, we assume that companies that are highly correlated, that is the stock prices move together for the companies to a very high degree, when deflects from their path are expected to converge back to the mean. The strategy expects underperforming securities to rebound and over performing securities to decrease in value.

Usually, deviation from the correlation is short-lived, and the stocks rapidly return to their previous correlation. At this point we tend to exploit this property

We need to note that there is thus a need to find the most correlated companies and the best measure to do so is the correlation coefficient. But only checking the correlation is not enough. We also have to look at the p-value to conclude whether there really is an exploitable correlation.

Hence we apply selection strategy onto 3 levels.

- **Correlation Coefficient**
- **P Value**
- **T Statistic**

The companies showing high value of Correlation Coefficient are expected to be more correlated to each other and those whose correlation values are low are immediately rejected. For those whose correlation coefficient is considerable (around 0.5-0.9) we run a further **P Value** test to choose the most reliably correlated pair of companies.

If the p-value is low enough we will be able to conclude that this correlation is not generated often or rather is generated only in rare cases thus adding weight to the correlation obtained.

Given the stock prices of all the companies we are going to find the correlation coefficient for every pair of these companies by looking at the data frames corresponding to the closing values of the stock prices in the selection period which would be then implemented in the testing period.

The time frame would be divided into 2 segments.

- 1st June 2017 to 1st June 2020 as the lookback period to select the companies.
- 1st July 2020 to 1st July 2023 as the backtesting period for the strategy/hypothesis.

A further measure that is important here and is essential to our analysis is the **Stationarity**. As discussed above, Stationarity is important so that we can be sure that the mean, standard deviation etc. doesn't deviate from their original values to a considerable value(taking significant deviation to be >5%).

The best metric that looks at this is the **Test Statistic**. The test statistic is calculated using the spread values passing it to the ADF function as a parameter. If the test statistic is less than the 1% critical value then we have an extremely low deviation of the local mean from the overall mean. Even If it is less that 5% critical value, still it is considerably good. If the test statistic is greater than 10% then the deviation of the local mean from the mean is too much and thus our purpose will not be solved hence rejected from consideration.

1.3 CODE IMPLEMENTATION

The Python code used to select a .....

- IMPORTING THE REQUIRED PYTHON LIBRARIES

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import datetime
import yfinance as yf
import statsmodels.api as sm
from statsmodels.regression.linear_model import OLS
from statsmodels.tsa.stattools import augmFull
from statsmodels.tsa.stattools import coint
from sklearn.model_selection import selec
```

- FETCHING DATA FROM YFINANCE LIBRARY

```
companies=pd.DataFrame()
stocks=["AAPL", "MSFT", "GOOGL", "AMZN", "TSLA", "JPM", "WMT", "JNJ",
"V", "PG", "KO", "NFLX", "DIS", "NVDA", "VZ", "T", "IBM",
"HD","BA","MA"]
for stock in stocks:
    stk=yf.Ticker(stock)
    data=stk.history(start='2017-06-01',end='2023-07-31')
    companies[stock]=data['Close']
companies.head()
```

- PLOTTING HEATMAP OF CORRELATION VALUES OF PAIRS.

```
train_close, test_close = selec(companies, test_size=0.5,
shuffle=False)[4]
fig, ax = plt.subplots(figsize=(20,14))
sns.heatmap(train_close.pct_change().corr(method ='pearson'), ax=ax,
cmap='coolwarm', annot=True, fmt=".3f") #spearman
ax.set_title('Assets Correlation Matrix')
```

[4] Breaks the fetched data into two parts which we refer to as selection period and backtesting period.In this chapter we would deal with “train_close” referring to the selection period

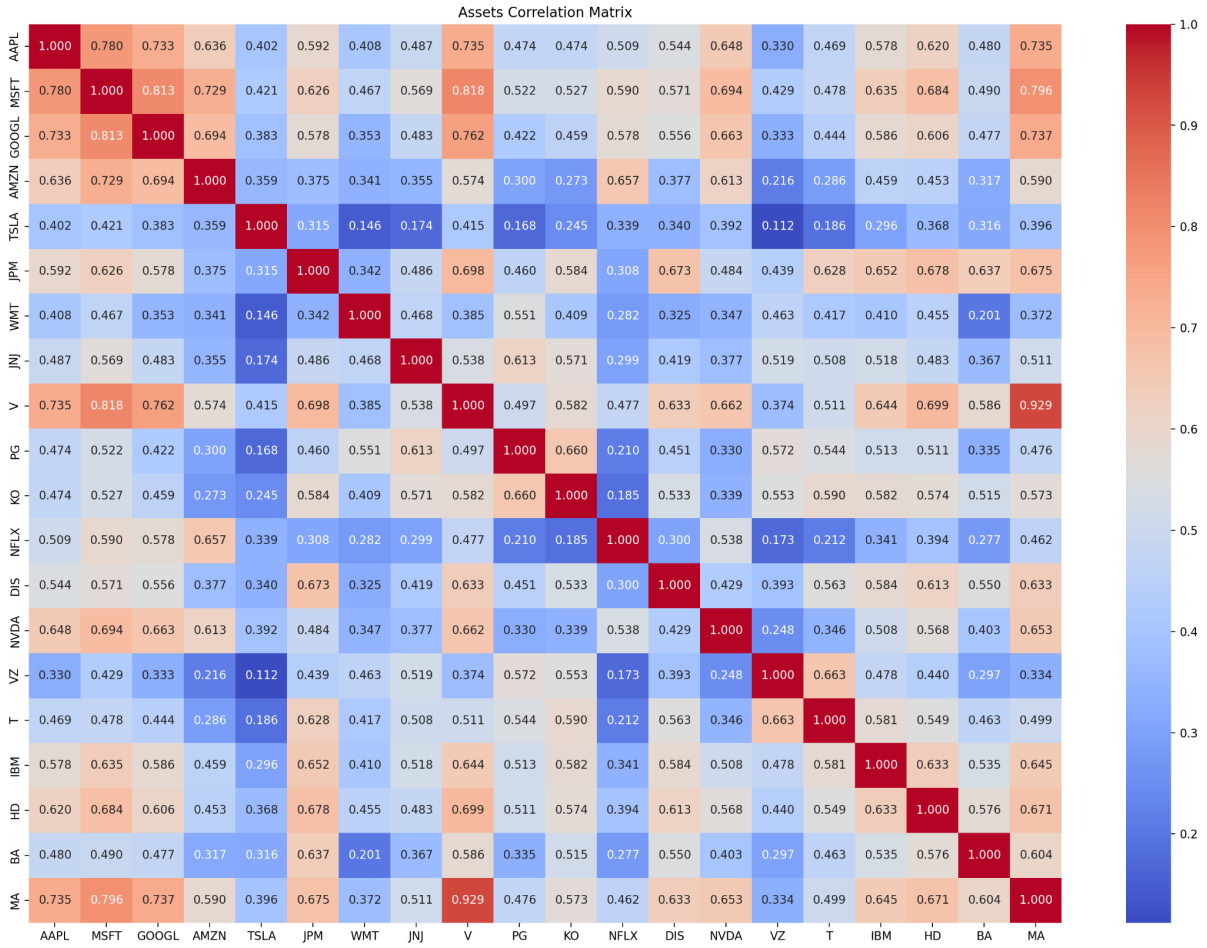


Fig.1 HeatMap representing correlation values

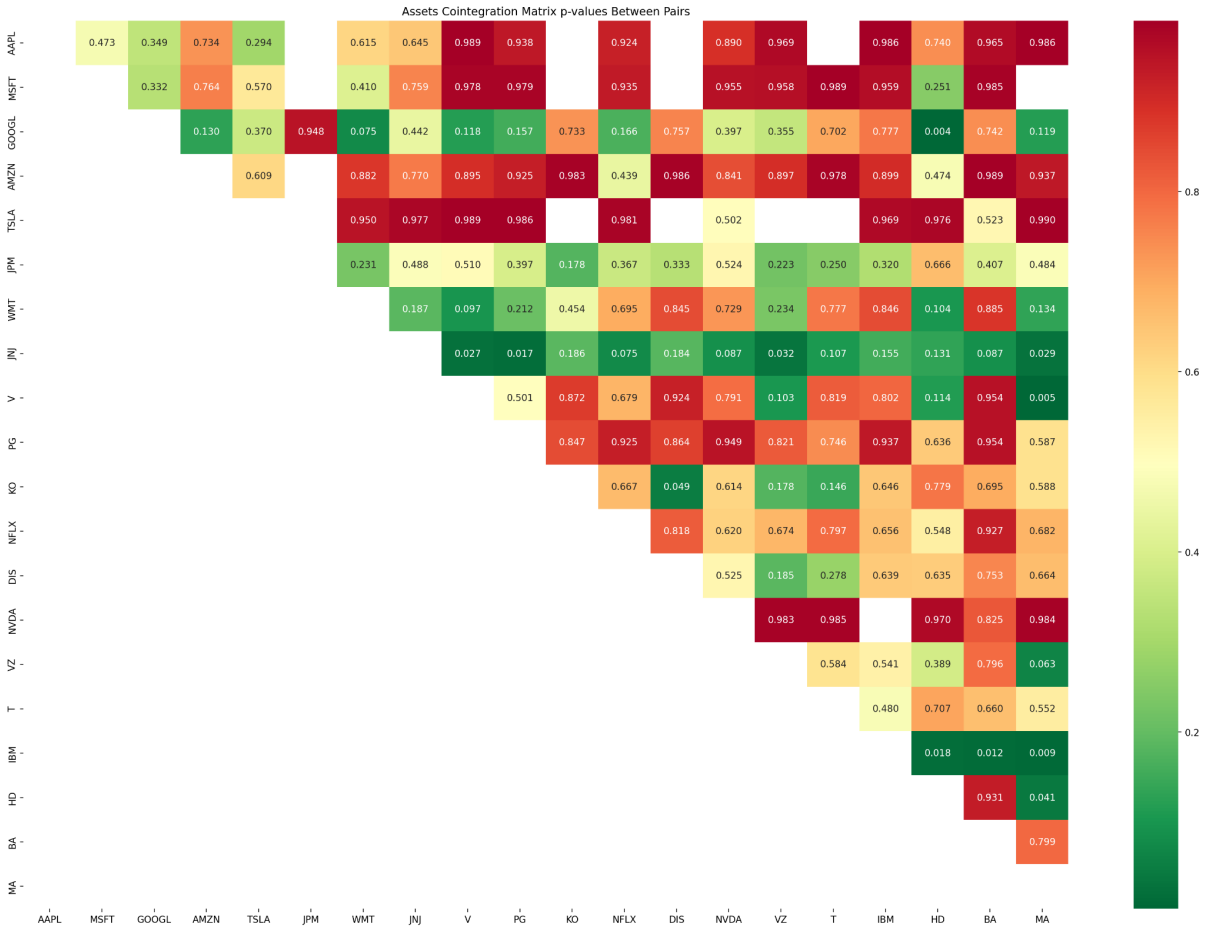
- CALCULATING P VALUE MATRIX AND ELIMINATING PAIRS.

```
def find_cointegrated_pairs(data):
    n = data.shape[1]
    pvalue_matrix = np.ones((n, n))
    keys = data.keys()
    pairs = []
    for i in range(n):
        for j in range(i+1, n):
            result = coint(data[keys[i]], data[keys[j]])
            pvalue_matrix[i, j] = result[1]
            if result[1] < 0.05:
                pairs.append((keys[i], keys[j]))
    return pvalue_matrix[5], pairs
pvalues, pairs = find_cointegrated_pairs(train_close)
```

[5] Returns an upper triangular matrix with some masked cells.

- PLOTTING HEAT MAP OF P VALUES BETWEEN PAIRS

```
fig, ax = plt.subplots(figsize=(20,14))
sns.heatmap(pvalues, xticklabels = train_close.columns,
                    yticklabels = train_close.columns, cmap = 'RdYlGn_r',
annot = True, fmt=".3f",
                    mask = (pvalues >= 0.99))
ax.set_title('Assets Cointegration Matrix p-values Between Pairs')
plt.tight_layout()
```

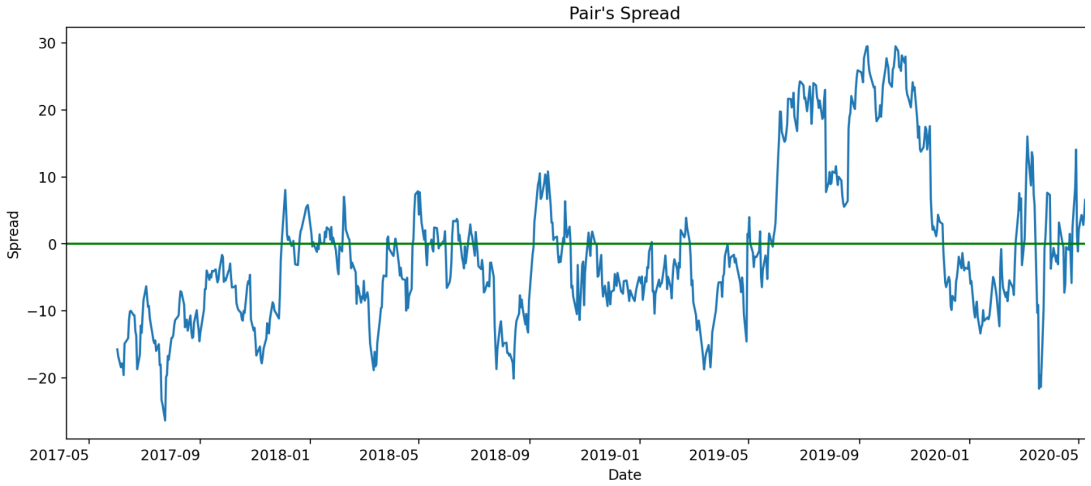


- SELECTING ONE OF THE PAIRS AND PROCEEDING FURTHER

```
asset1,asset2='GOOGL','HD'
train = pd.DataFrame()
train['asset1'] = train_close[asset1]
train['asset2'] = train_close[asset2]
ax = train[['asset1','asset2']].plot(figsize=(12, 6), title = 'Daily
Closing Prices for {} and {}'.format(asset1,asset2))
ax.set_ylabel("Closing Price")
ax.grid(True);
```

- APPLYING LINEAR REGRESSION AND PLOTTING SPREAD.

```
model=sm.OLS(train.asset2, train.asset1).fit()
spread = train.asset2 - model.params[0] * train.asset1
ax = spread.plot(figsize=(12, 6), title = "Pair's Spread")
ax.set_ylabel("Spread")
ax.grid(True);
```



- USING AUGMENTED DICKEY FULLER TEST TO CHECK STATIONARITY

```
adf = augmFull(spread, maxlag = 1)
print('Critical Value = ', adf[0])
print(adf[4])
```

OUTPUT

```
T-STATISTIC = -3.439626649280881
CRITICAL VALUES = {'1%': -3.4388268991356936,'5%':-2.8652813916285518,
'10%': -2.5687622857867782}
```

Since the T-STATISTIC is less than the 1% critical value, we are ready to choose this pair of companies.

Chapter 2

Trading Strategy and Signal Generation Methods

Herein we would be discussing

- How to develop strategies to invest into the selected companies,
- How to decide the entry and exit positions and timings,
- Signal Generation at the passage points,
- What are going to be the criterias for them that would yield the maximum profit.

We will try to answer these questions in this chapter. But we start with the necessary terms and definitions.

2.1 INTRODUCTORY TERMS AND DEFINITIONS

➢ Z-Score

A Z-Score[6] is a statistical measurement of a score's relationship to the mean in a group of scores. It is measured in terms of standard deviations from the mean.

- If a Z-score is 0, it indicates that the data point's score is identical to the mean score.
- A Z-score of 1 indicates that the data point's score is one standard deviation away from the overall mean.

Z-scores are measures of an instrument's variability and are used to determine volatility.

[6] It can be calculated directly or maybe calculated using moving averages which would be discussed in subsequent sections

➢ Moving Average

Moving averages are calculated to identify the trend direction of a stock or to determine its support and resistance levels. It is a trend-following or lagging indicator because it is based on past prices.

- A rising moving average indicates that the security is in an uptrend, while a declining moving average indicates a downtrend.

➢ Z-Tables

A Z-Table, also known as the standard normal table, provides the area under the curve to the left of a z-score in a normal distribution. This area represents the probability that z-values will fall within a region of the standard normal distribution. We can use a z-table to find probabilities corresponding to ranges of z-scores and to find p-values for z-tests.

2.2 OUR TRADING STRATEGY

We are going to find the 6-day window moving average and the 10-day moving average at every point of the ratio values.These ratio values are calculated simply by dividing the first asset by the second asset. After calculating these values at every point we calculate the moving average based z-score values for each of the points and thus these values are more indicative of the trend in the near past and help in recognizing the correct time to enter and exit the market.

Entry Strategy

Whenever this Z-score calculated using the moving average exceeds 1 standard deviation from the mean value we enter the market. If it goes in the positive direction then we **Short ASSET1** and **Long ASSET2**. If instead it goes below the mean by 1 standard deviation,we long at ASSET1 and short at ASSET2. Here the standard deviation and the mean we are using are calculated based on the moving average z score values. More Formally,

$$\epsilon > \mu + \sigma \Rightarrow \text{Short Asset1 and Long Asset2}$$
$$\epsilon < \mu - \sigma \Rightarrow \text{Short Asset2 and Long Asset1}$$

These positions generate signals indicating the right time and position to invest or sell stocks.These have been generated and shown visually in the code implementation and graphs that follow.

Exit Strategy:

We implement a stop loss order when the exceeds 2 times the standard deviation from the mean of the z score values.in either directions.More formally
Let ε be the moving average z score values,Then if

$$\epsilon > \mu + 2\sigma$$

We place the stop loss order.At the other end of the spectrum we will place a take-profit order if

$$\epsilon < \mu - 2\sigma$$

These Conditions are appropriately fine-tuned to yield to maximum yield in a Low-Frequency Trading setting.The amount of variation that we allow before raising any signals changes from company pair to company pair,but these values are fine tuned for the Companies that we chose.These values change for different types of companies,the stages they are in,and since pairs trading strategy is neutral to the market, these values don't depend on time frame.

2.3 CODE IMPLEMENTATION

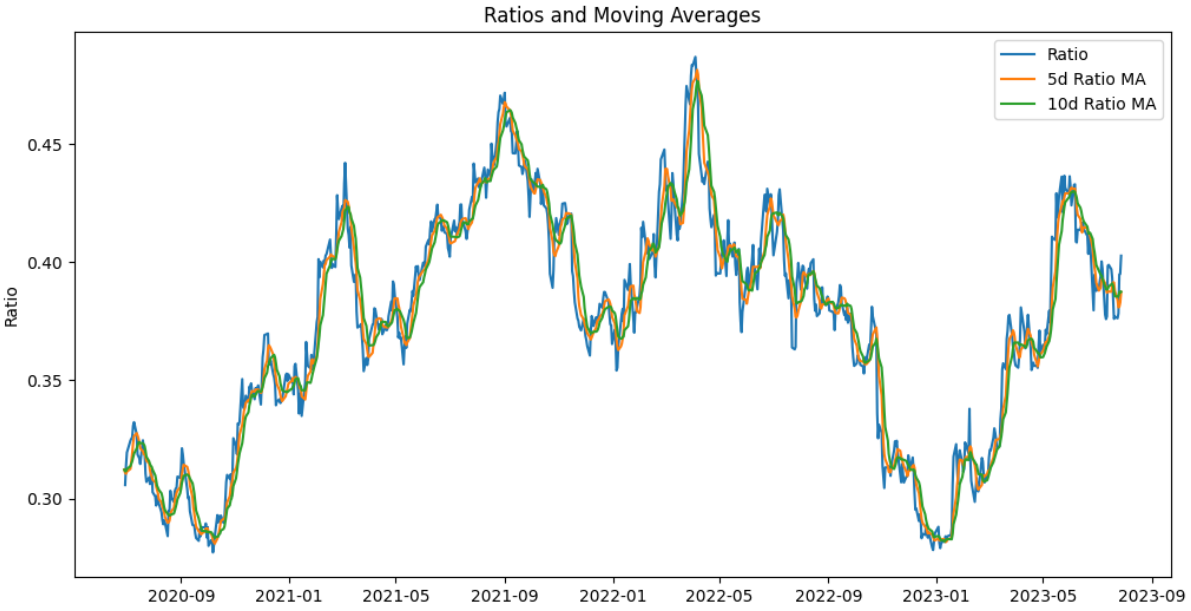
The Python code used to select a .....

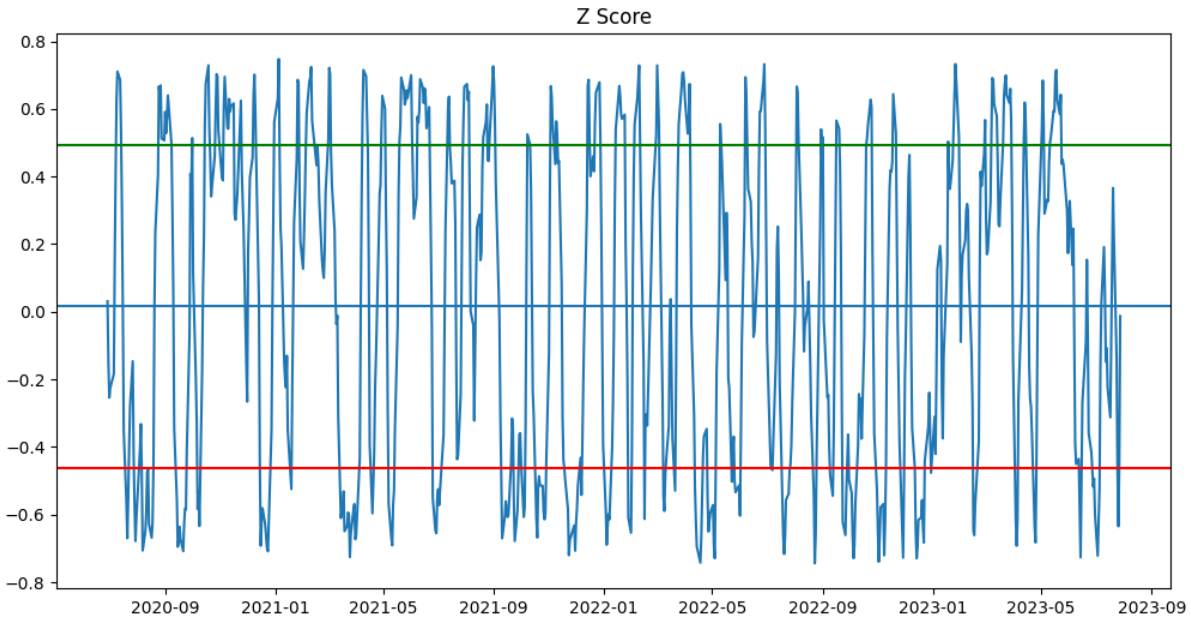
- CALCULATING Z-SCORE USING MOVING AVERAGES OVER A WINDOW.

```
win=10
test=pd.concat([train_close[train_close.shape[0]-win-1:],test_close],
sort=False)
signs = pd.DataFrame()
signs['asset1'] = test[asset1]
signs['asset2'] = test[asset2]
ratios = signs.asset1 / signs.asset2
ratios_mavg5 = ratios.rolling(window=6, center=False).mean()
ratios_mavg5=ratios_mavg5[win:]
ratios_mavg10 = ratios.rolling(window=win, center=False).mean()
ratios_mavg10=ratios_mavg10[win:]
std_10 = ratios.rolling(window=win, center=False).std()
std_10=std_10[win:]
zscore_10_5 = (ratios_mavg5 - ratios_mavg10)/std_10
ratios=ratios[win:]
zscore_10_5=zscore_10_5.dropna()
```

- PLOTTING RATIOS AND MOVING AVERAGES OVER THE WINDOWS AND Z SCORES.

```
plt.figure(figsize=(12, 6))
plt.plot(ratios.index, ratios.values)
plt.plot(ratios_mavg5.index, ratios_mavg5.values)
plt.plot(ratios_mavg10.index, ratios_mavg10.values)
plt.legend(['Ratio', '5d Ratio MA', '10d Ratio MA'])
plt.ylabel('Ratio')
plt.show()
plt.figure(figsize=(12, 6))
plt.plot(zscore_10_5)
plt.axhline(zscore_10_5.mean())
plt.axhline(zscore_10_5.std()+zscore_10_5.mean(),c='g')
plt.axhline(zscore_10_5.mean()-zscore_10_5.std(),c='r')
```


Ratios and Moving Averages
Ratio
5d Ratio MA
10d Ratio MA
Ratio
0.45
0.40
0.35
0.30
2020-09
2021-01
2021-05
2021-09
2022-01
2022-05
2022-09
2023-01
2023-05
2023-09

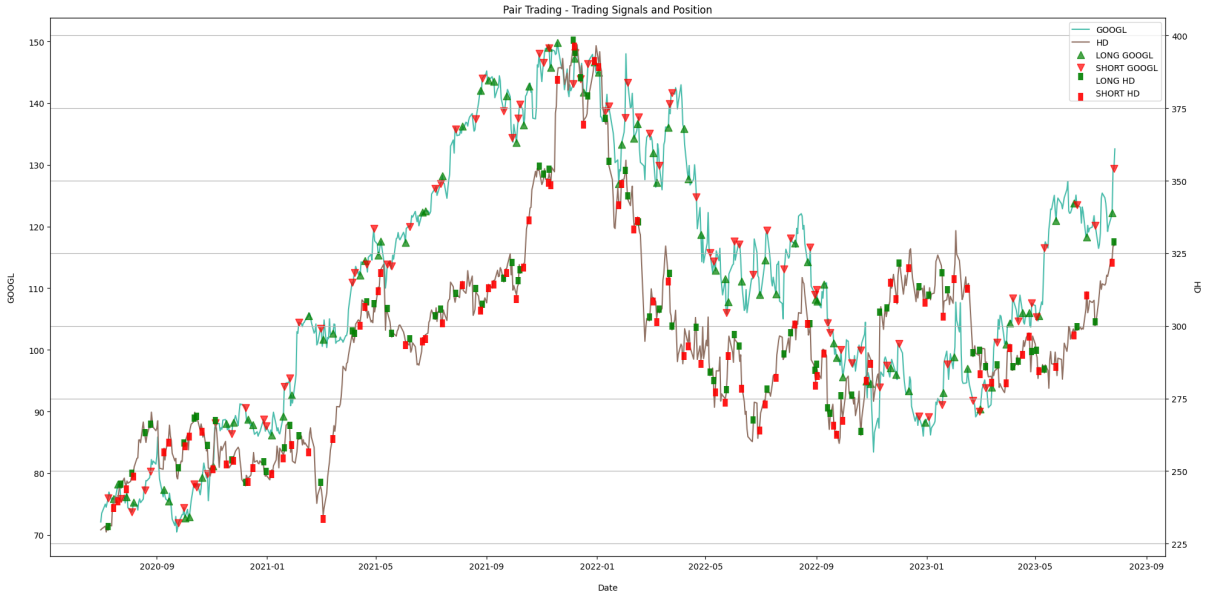

Z Score
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
2020-09
2021-01
2021-05
2021-09
2022-01
2022-05
2022-09
2023-01
2023-05
2023-09

- SIGNAL GENERATION AND PLOTTING SIGNALS ON TIME AXIS WITH STOCKS

```
signals = pd.DataFrame()
signals['asset1'] = test_close[asset1]
signals['asset2'] = test_close[asset2]
ratios = signals.asset1 / signals.asset2
signals['z'] = zscore_10_5
signals['z upper limit'] = np.mean(signals['z']) +
np.std(signals['z'])
signals['z lower limit'] = np.mean(signals['z']) -
np.std(signals['z'])
signals['signals1'] = 0
signals['signals1'] = np.select([signals['z'] > signals['z upper
limit'], signals['z'] < signals['z lower limit']], [-1, 1], default=0)
signals['positions1'] = signals['signals1'].diff()
signals['signals2'] = -signals['signals1']
signals['positions2'] = signals['signals2'].diff()
signals=signals.dropna()
fig=plt.figure(figsize=(20,10))
bx = fig.add_subplot(111)
bx2 = bx.twinx()
l1, = bx.plot(signals['asset1'], c='#4abdac')
l2, = bx2.plot(signals['asset2'], c='#907163')
u1, = bx.plot(signals['asset1'][signals['positions1'] == 1], lw=0,
marker='^', markersize=8, c='g',alpha=0.7)
d1, = bx.plot(signals['asset1'][signals['positions1'] == -1],
lw=0,marker='v',markersize=8, c='r',alpha=0.7)
u2, = bx2.plot(signals['asset2'][signals['positions2'] == 1],
lw=0,marker=2,markersize=9, c='g',alpha=0.9, markeredgewidth=6)
d2, = bx2.plot(signals['asset2'][signals['positions2'] == -1],
lw=0,marker=3,markersize=9, c='r',alpha=0.9,markeredgewidth=6)

bx.set_ylabel(asset1,)
bx2.set_ylabel(asset2, rotation=270)
bx.yaxis.labelpad=15
bx2.yaxis.labelpad=15
bx.set_xlabel('Date')
bx.xaxis.labelpad=15
plt.legend([l1,l2,u1,d1,u2,d2], [asset1, asset2,'LONG
{}'.format(asset1),'SHORT {}'.format(asset1),'LONG
{}'.format(asset2),'SHORT {}'.format(asset2)], loc ='best')

plt.title('Pair Trading - Trading Signals and Position')
plt.xlabel('Date')
plt.grid(True)
plt.tight_layout()
```


Pair Trading - Trading Signals and Position
GOOGL
HD
LONG GOOGL
SHORT GOOGL
LONG HD
SHORT HD
GOOGL
HD
Date
2020-09
2021-01
2021-05
2021-09
2022-01
2022-05
2022-09
2023-01
2023-05
2023-09

Chapter 3

Risk Management Measures

It seems that the Hedging position that the pairs trading strategy gives a trader a lot of advantage at the same time there are multiple disadvantages to this type of trading that come to a trader in the form of risks. Some of which are listed below.

- **Reliance of The High Statistical Correlation:**
  - Pairs trading relies on securities with a high statistical correlation. Most traders require a correlation of at least 0.80, which is challenging to recognize.
- **Price Filling:**
  - The generation of profits in pairs trading involves relying on the margins that are too less, and the transactions are made in large quantities, which shows that the risk of falling stock orders at the desired price when positions are open in a pair trading is high. Even a small difference in the security's purchase price or sale price can prove significant as the volume of transactions is high
- **High Commission:**
  - Some traders highly discourage Pairs trading because of its higher commission charges. Sometimes even a single Pairs trade requires a Pair trader to pay a commission that is just double the normal commission required in the standard trade.

What we have done to overcome these are as follows:
We have used stop loss and take profit orders so that when it seems that we are now going start making losses then we simply exit using one of these strategies and then so that we don't incur loss that automatically improves our gains.

Chapter 4

Trading Signals generated and Position Sizing

4.1 Trading Signals Generated

As per the criteria we had set as to when to enter the market and trade and when to exit is called multiple times throughout the course of the backtesting period.
These signals have been indicated in the graphs that follow in the form

- Green and Red Arrow Markers for **Asset1**
- Green and Red Rectangle markers for **Asset2**

The signals for trading were generated **88 times** in the course of the backtesting period. This indicates that we have successfully executed a **low frequency trade** (A typical low frequency trade has a frequency of a few days to week) with 88 trading days, we stand at an average of about 12-13 days per trade.

4.2 Position Sizing

It refers to the technique of determining the size of your trade. The size of a trade could be in terms of the amount of capital to be used in one trade or the quantity i.e. the number of shares to buy or sell in a trade. In both cases, position sizing helps by:

- Optimising profit potential
- Preventing big losses

There are two major ways to do position sizing:

4.2.1 Capital Based

This is a very straightforward method where the capital is equally distributed in each trade.

For example, if your capital is ₹5 lakh, you may want to allocate 10% (or ₹50,000) to each trade. It means you can execute 10 trades instead of putting the entire capital in one trade. Alternatively, a conservative trader could apply a fixed percentage (say 10%) to a reducing capital balance.This method is conservative because you could execute more than 10 trades (compared to the previous method) and thus distribute risk. In the above examples, we have considered a 10% allocation to each trade. However, you could choose a number that strikes the right balance for you between diversification and risk tolerance.

4.2.2 Risk Based

Here you risk a small percentage of your total capital on each trade and decide the position size based on the risk amount. A simple way to calculate risk is entry price minus stop loss.

Let's say you decide to risk 1% per trade. So based on your capital of ₹5 lakh, 1% comes to ₹5,000. The position size will be calculated as: Position size = Risk per trade / Risk per share.

Our Position Sizing

4.2.3 Past Statistic Based

We have decided to divide the initial amount provided to us that is INR 100000,in the ratio of ß:1,where ß is the linear regression slope constant. We have done this because basically the movements in Asset1(if it is larger) would generate larger amounts of profit and would increase the risk as well. The other way around reduces the risk but at the same time reduces the profits that we make as well. This ratio came out to us when we tried to find the most optimal ratio that generated maximum profit in the training dataset from June 2017 to June 2020.

Chapter 5

Portfolio PnL and Performance Metrics

This chapter deals with the **Portfolio Profit and Losses** throughout the backtesting period.
As a ritual let us first define some important terms and then we are going to see how our portfolio has performed based on some performance metrics using the strategy discussed in Chapter 2.

5.1 INTRODUCTORY TERMS AND DEFINITIONS

➢ CAGR

Compound annual growth rate (CAGR) is a way to measure how an investment or business has grown over a specific period of time. It takes into account the effect of compounding, which means that the growth builds upon itself.

$$\text{CAGR} = \left(\left(\frac{EV}{BV}\right)^{\frac{1}{n}} - 1\right) \times 100$$

Where,

- EV is Ending Value
- BV is Beginning Value
- n is number of years

➢ Cumulative Returns

The Total return produced by an investment over a predetermined period is known as the cumulative return. It is the total gain or loss on investment throughout time, regardless of the time involved. This kind of return takes profits and losses from the period into account and bases the final total on

the change in value from the start to the end of the same period. A cumulative return is typically shown as a percentage rather than a monetary value.

➢ Annualised Sharpe Ratio

The Annualised Sharpe Ratio is computed by dividing the annualised mean monthly excess return by the annualised monthly standard deviation of excess return. Equivalently, the annualised Sharpe Ratio equals the monthly Sharpe Ratio times the square root of 12.

The Sharpe ratio compares the return of an investment with its risk. It's a mathematical expression of the insight that excess returns over a period of time may signify more volatility and risk, rather than investing skill.

➢ Maximum Drawdown

A maximum drawdown (MDD) is the maximum observed loss from a peak to a trough of a portfolio, before a new peak is attained. Maximum drawdown is an indicator of downside risk over a specified time period.

It can be used both as a stand-alone measure or as an input into other metrics such as "Return over Maximum Drawdown" and the Calmar Ratio. Maximum Drawdown is expressed in percentage terms.

$$\text{MDD} = \frac{\text{Trough Value} - \text{Peak Value}}{\text{Peak Value}}$$

➢ Trading Frequency

Trading Frequency is basically the number of trades executed in a specific time interval.

5.2 CODE IMPLEMENTATION

The Python code used to select a .....

- DIVIDING INITIAL CAPITAL

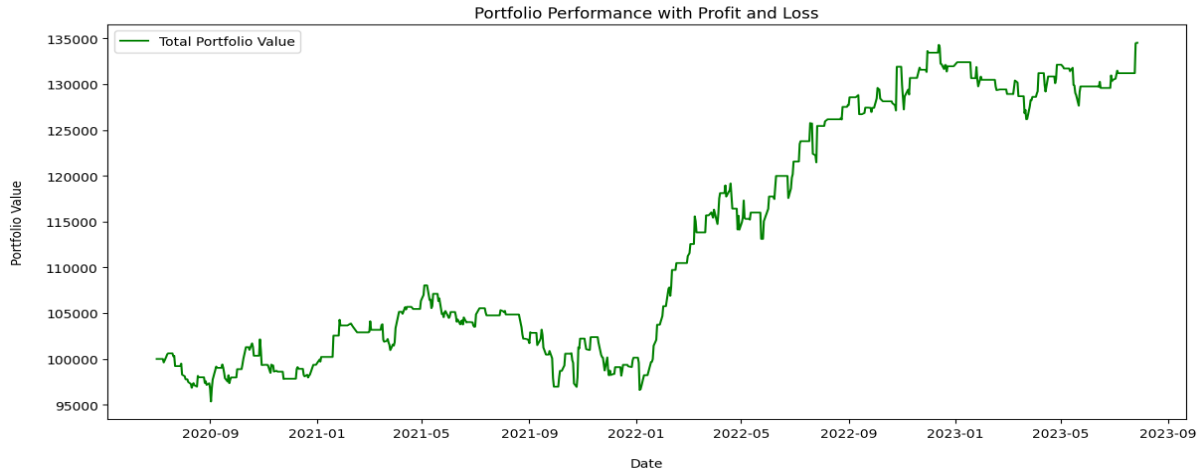
```
initial_capital = 100000
initial_capital2=initial_capital/(1+model.params[0])
initial_capital1=model.params[0]*initial_capital/(1+model.params[0])
positions1 = initial_capital1// max(signals['asset1'])
positions2 = initial_capital2// max(signals['asset2'])
```

- MAINTAINING A PORTFOLIO DATAFRAME

```
portfolio = pd.DataFrame()
portfolio['asset1'] = signals['asset1']
portfolio['holdings1'] = signals['positions1'].cumsum() *
signals['asset1'] * positions1
portfolio['cash1'] = initial_capital1 - (signals['positions1'] *
signals['asset1'] * positions1).cumsum()
portfolio['total asset1'] = portfolio['holdings1'] +
portfolio['cash1']
portfolio['return1'] = portfolio['total asset1'].pct_change()
portfolio['positions1'] = signals['positions1']
portfolio['asset2'] = signals['asset2']
portfolio['holdings2'] = signals['positions2'].cumsum() *
signals['asset2'] * positions2
portfolio['cash2'] = initial_capital2 - (signals['positions2'] *
signals['asset2'] * positions2).cumsum()
portfolio['total asset2'] = portfolio['holdings2'] +
portfolio['cash2']
portfolio['return2'] = portfolio['total asset2'].pct_change()
portfolio['positions2'] = signals['positions2']
portfolio['z'] = signals['z']
portfolio['total asset'] = portfolio['total asset1'] +
portfolio['total asset2']
portfolio['return']=portfolio['total asset'].pct_change()
portfolio['z upper limit'] = signals['z upper limit']
portfolio['z lower limit'] = signals['z lower limit']
portfolio = portfolio.dropna()
```

- PLOTTING CUMULATIVE RETURNS (PORTFOLIO PERFORMANCE)

```
fig = plt.figure(figsize=(14,6))
ax = fig.add_subplot(111)
l1, = ax.plot(portfolio['total asset'], c='g')
ax.set_ylabel('Portfolio Value')
ax.yaxis.labelpad=15
ax.set_xlabel('Date')
ax.xaxis.labelpad=15
plt.title('Portfolio Performance with Profit and Loss')
plt.legend([l1],['Total Portfolio Value'],loc='upper left');
```



- COMPUTING CAGR, MAXIMUM DRAWDOWN AND FREQUENCY OF TRADES PLACED

```
final_portfolio = portfolio['total asset'].iloc[-1]
delta = (portfolio.index[-1] - portfolio.index[0]).days
print('Number of days = ', delta)
YEAR_DAYS = 365
returns = (final_portfolio/initial_capital) ** (YEAR_DAYS/delta) - 1
print('CAGR = {:.3f}%' .format(returns * 100))
Maximum_Drawdown=(portfolio['total asset'].min()-portfolio['total
asset'].max())/portfolio['total asset'].max()
print("Maximum_Drawdown is: ",Maximum_Drawdown)
print("Frequency is: ", signals['asset1'][signals['positions1'] ==
1].count())
```

```
Number of days = 1121
CAGR = 10.140%
Maximum_Drawdown is: -0.29122737972386
Frequency is: 88
```

References

➔ WEBSITES

[1] https://upstox.com/market-talk/position-sizing
[2]https://medium.com/analytics-vidhya/statistical-arbitrage-with-pairs-trading-and-backtesting
[3] https://www.investopedia.com/terms
[4] https://corporatefinanceinstitute.com/
[5] https://www.quora.com/
[6] https://pressbooks-dev.oer.hawaii.edu/
[7] https://learn.robinhood.com/
[8] https://www.xlstat.com/en/solutions
[9] https://www.oreilly.com/library/view
[10] https://github.com/

➔ BOOKS USED

[1] Mathematics for Finance: An Introduction to Financial Engineering Book by Marek Capiński and T.J. Zastawniak
[2] Ganapathy Vidyamurthy, (2004). Pairs Trading Quantitative Methods and Analysis. John Wiley Sons, Inc., Hoboken, New Jersey
[3] Andrew Pole, (2007). Statistical Arbitrage Algorithmic Trading Insights and Techniques. John Wiley Sons, Inc., Hoboken, New Jersey.