NAVELLIER
Calculated Investing

PRIVATE
CLIENT
GROUP

# STUDY ON STATISTICAL ARBITRAGE MEAN REVERSION STRATEGIES AND THEIR IMPLEMENTATION

Authored by James Hopham

January 25, 2024

Navellier & Associates, Inc.
One East Liberty, Suite 504
Reno, Nevada 89501

800-887-8671
info@navellier.com

www.navellier.com

NCD-24-0323

## Overview of Mean Reversion

Mean reversion is a statistical arbitrage trading strategy that market makers use to ideally generate profit in both intraday and interday trades, and gained notoriety during the early 2000s. The method aims to exploit temporary price deviations to generate trading signals by measuring an underlying single asset or portfolio of assets to determine if a mean reverting opportunity exists. Over long-term periods, stock prices generally exhibit a geometric random walk [8]. Typical time series analysis methods deal with nonstationarity through methods such as in ARMA and ARIMA modeling to filter the data to generate a model after conditions such as constant mean and variance are achieved.

In practice there are two main applications of mean reversion, with the first being directional trading in which one asset is considered where the trader uses either a simple moving average tool or a more sophisticated ARIMA model to model price movements. Typically, this approach is not used because most stocks will still not exhibit stationary behavior sufficient enough for a reliable model. The second more commonly used strategy is for market-neutral trading in which two or more assets are considered; e.g., long-short pairs trading. This approach demonstrates more potential use because it satisfies stationarity through cointegration of a chosen unit portfolio that can contain a combination of stocks, ETFs, currencies, commodities, or futures contracts. The trader then uses predetermined signal measures to decide whether or not to enter a trade; for example, bollinger bands to measure standard deviation.

The nuances in the strategy can be attributed to how often one wants to enter or exit trades, which combination of a unit portfolio is used to determine the mean reverting process, how the data is filtered, and the setup of the price spreads or ratios. For example, a standard retail trader may make decisions based on standard deviation measured with bollinger bands (with a lookback of 1 hour, 5 hours, daily, etc.) against indicators such as simple moving average or volume-weighted average price (VWAP). A variety of alternatives to standard bollinger bands could also be used to determine entry and exit points, such as the Ornstein-Uhlenbeck equation to compare profits of a single round-trade or the HJB (Hamilton-Jacobi-Bellman) equation to compare the return of a single trade based on distance from the mean. Other approaches presented by institutional traders and academia may employ more sophisticated filtering techniques against the data before looking for significant deviations from the mean.

Markus Pelger [4] for example suggests in a paper with one of his PhD students that convolutional neural networks and transformers may be used as a way to filter the data and detect patterns between corresponding filters of the neural network. The inputs or patterns that are estimated to contribute towards the most optimal minimization of a cost function such as the MSE will be assigned corresponding weights. A multilayer feed-forward network is constructed in which each layer of nodes receives inputs from the previous layer to continue this process with the output being modified by a nonlinear function to give input for the next layer to reduce extreme outliers for example through a nonlinear sigmoid function. The transformers are then used to extract the most notable weight functions.



NAVELLIER'S PRIVATE CLIENT GROUP
www.navellier.com | info@navellier.com | 800-887-8671

Ernest Chan [2, 3] on the other hand objects to the usage of deep learning methods in this particular instance. Financial institutions have successfully leveraged deep learning models to model credit card transactions and fraud detection; however, they have billions of transaction data collections to train their models. In a linear model, there are relatively few parameters to consider in contrast to a machine learning model that may have hundreds of parameters and a deep learning model which may have thousands to consider. As a result, backtested performance of neural networks tends to be high while suffering low out-of-sample accuracy when the model is employed in real time. Techniques such as cross-validation and dropout can potentially reduce overfitting; however, low frequency trading using daily time series data would not offer enough data points to allow these methods to sufficiently bypass this underlying issue. Other approaches such as using the Kalman filter, exponential smoothing, or bootstrapping could be tested for usage as alternative methods.

### Potential Uses in Quantitative Trading HFT Settings

Although single stocks and unit portfolios of assets may not meet statistical significance for stationarity, if the assets of interest have a significance level that holds some stationarity (e.g., exhibit at least a weak level of cointegration) then they can potentially be used in a high frequency trading strategy. In the long term, stocks generally exhibit a random walk; however, short intervals, such as those within intraday strategies, can potentially depict mean reverting behavior. This provides opportunities for firms such as Citadel who have algorithms automated to test for trades within milliseconds to make orders based on temporary price movements. Additionally, the large number of trades allows them to take advantage of the law of large numbers when executing their models. Furthermore, low frequency trading would not have sufficient data points to support deep learning models; however, firms operating under intervals of milliseconds may be able to exploit neural networks to their advantage. For example, these types of firms may additionally be able to use reinforcement learning through recurrent neural networks to consider the reactions of people placing orders to model prices in very short timeframes. These firms also have an inherent advantage through their IT infrastructure setup that aims to minimize latency between live data feed and market orders allowing them to better take advantage of statistical arbitrage opportunities.

### Methodology

To conduct a mean reversion strategy, stationarity must first be addressed to assess if a model can be made. For a single asset trade, the ADF (Augmented Dickey-Fuller) test is used, where the null hypothesis is that the proportionality constant between the change of the price series in the next period and the difference between the mean price and current price is zero (e.g., no stationarity). For a strategy containing two or more price series, the CADF (Cointegrated Augmented Dickey-Fuller) test or the Johansen test can be used; the Johansen test is particularly useful because it can help generate hedge ratios for the portfolio. Since stationarity can be thought of as being a slower deviation from the initial value than a random walk, we can further expand on this idea by using the Hurst exponent where a mean-reverting series will have $H < .5$ , and we can use the variance ratio test as a measure of statistical significance for the Hurst exponent.

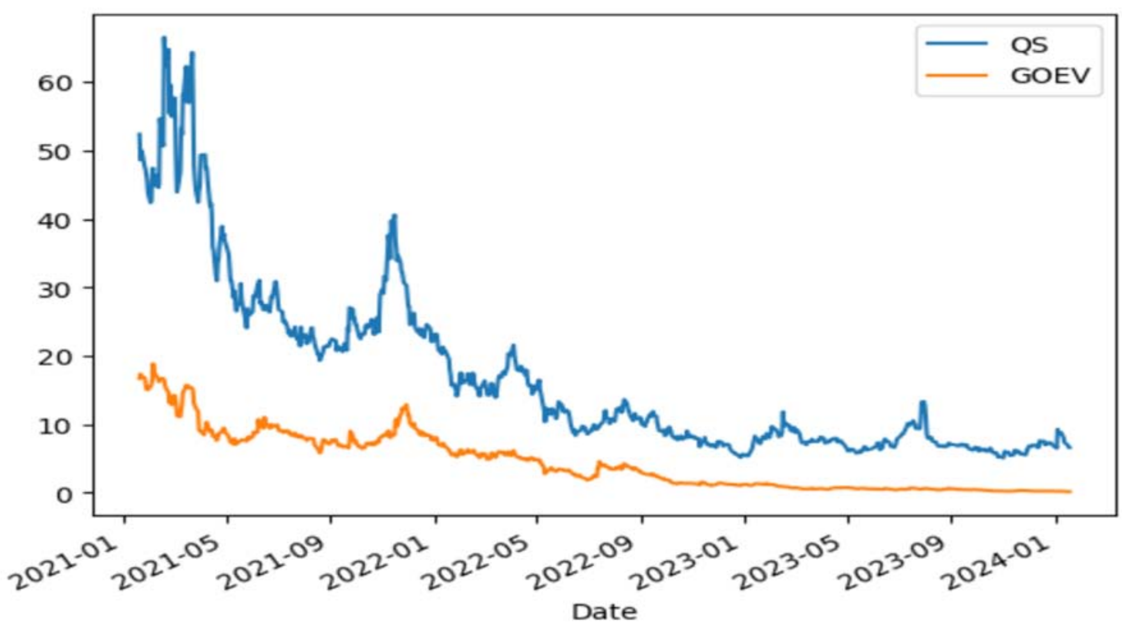
After verifying stationarity, we then decide on a mean-reverting portfolio based on either a fixed number of shares or a fixed market value for each asset. Depending on the desired type of portfolio or assets included in the portfolio, different transformations to the price series such as application of a log transformation or a ratio could be useful. Alternatively, a method to tune the data can be used to generate a more accurate model such as through applying the Kalman filter to distribute weights based on linear relationships between time points and recency of the data; the Kalman filter is useful because it can also be used to update the hedge ratio dynamically which can provide a more accurate weighting scheme than the Johansen test eigenvector. Once we generate our model, we can determine our entry and exit points through a variety of methods; bollinger bands (e.g. standard deviation) will commonly be used. Schoenberg and Corwin [9] argue that entering at two or more bollinger bands is typically suboptimal and that entering a full position is more beneficial than averaging in on this strategy.

(*note: Review of applications of the HJB equation [1] has also demonstrated positive results; another avenue that also could be useful to study for entry/exit points is the application of random forest models)

### Python Implementation Example

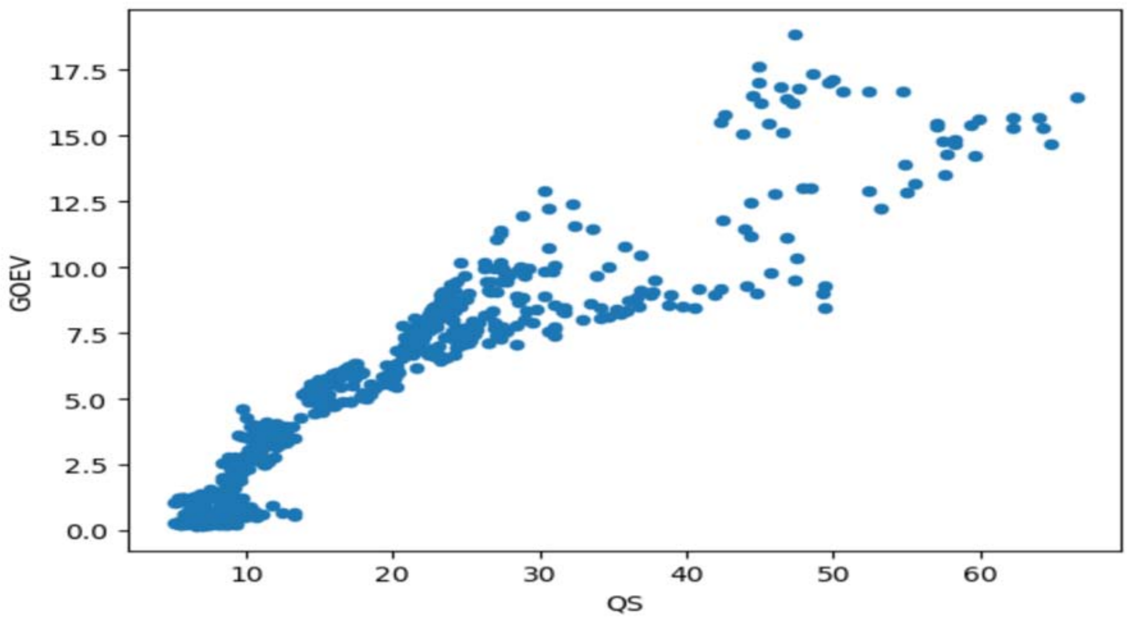
In this test of the mean reversion strategy, I used Canoo (GOEV) and QuantumScape (QS) to demonstrate pair trading. The data was sourced through the yahoo finance API and imported directly into python.

***Figure 1. QuantumScape and Canoo Price Movement***



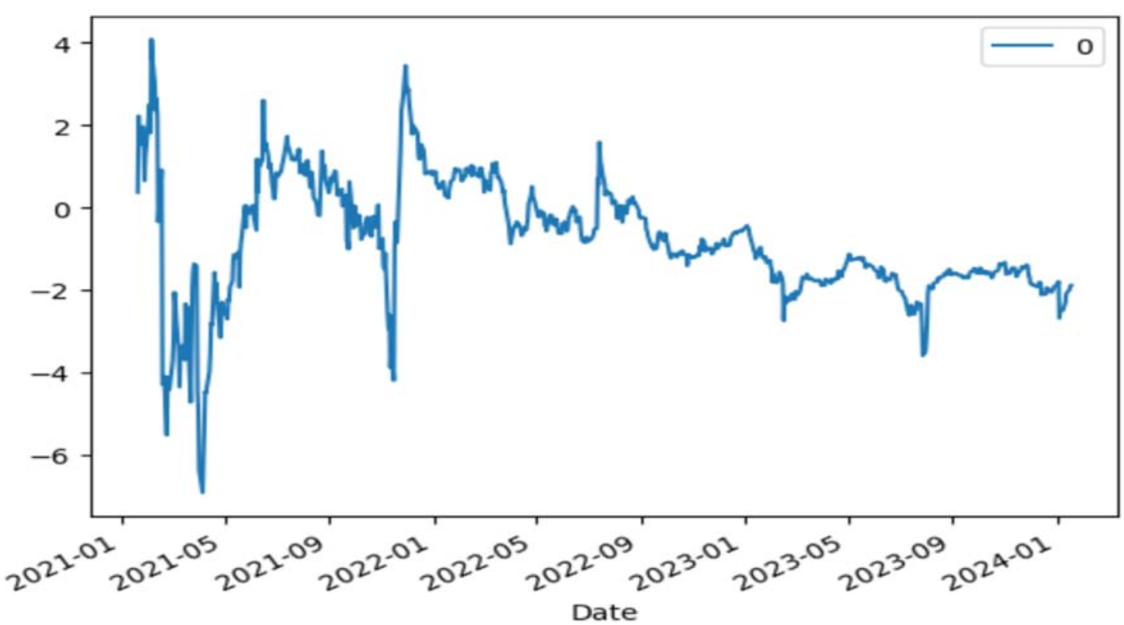

Source: Price data from Yahoo Finance. Calculations and statistical tests were completed by author James Hopham in Python through both manual calculations and using open-source packages/tools. Graphs are for illustrative and discussion purposes only. Please read important disclosures at the end of this commentary.

*Figure 2. QuantumScape vs. Canoo Daily Price*



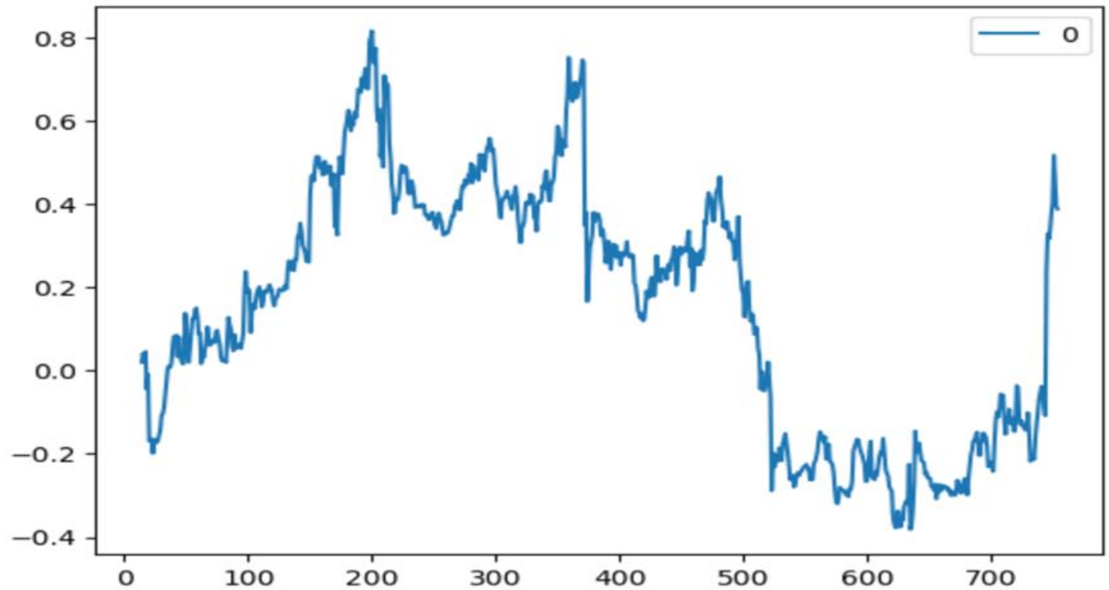

Source: Price data from Yahoo Finance. Calculations and statistical tests were completed by author James Hopham in Python through both manual calculations and using open-source packages/tools. Graphs are for illustrative and discussion purposes only. Please read important disclosures at the end of this commentary.

*Figure 3. QuantumScape and Canoo Spreads (e.g., Cointegration Step)*




Source: Price data from Yahoo Finance. Calculations and statistical tests were completed by author James Hopham in Python through both manual calculations and using open-source packages/tools. Graphs are for illustrative and discussion purposes only. Please read important disclosures at the end of this commentary.

*Figure 4. Cumulative Returns of GOEV - QS Strategy*




Source: Price data from Yahoo Finance. Calculations and statistical tests were completed by author James Hopham in Python through both manual calculations and using open-source packages/tools. Graphs are for illustrative and discussion purposes only. Please read important disclosures at the end of this commentary.

I decided to compare these companies as one of the pairs I tested because of their similar involvement in batteries. The CADF test demonstrated weak cointegration below the 90% significance level with a p-value of .12 but a half-life to reversion of around 14.45 trading days. Overall this demonstrated an APR of 11.6% and a Sharpe ratio of .48; however, this test did not factor in transaction costs. All python code is attached in the appendix at the end of this report.

### Potential Use-Cases for Short Covering Rallies for F-Rated Stocks

The mean reversion strategy relies on the assets or unit portfolios grouped together to be subject to similar pressures and fundamental catalysts within the market. Correlation between assets measures short term tracking between two assets; however, in this strategy, we use cointegration which measures the long term tendency for the assets to revert. This allows us to have temporary fluctuations in their spread; however, it also assumes that they will eventually react to these pressures with similar responses.

In the case of F-Rated stocks, if it has been observed that these stocks tend to temporarily lead in short covering rallies then a potential arbitrage strategy can be performed by finding suitable assets or portfolio groupings to pair and observing the behavior between them. If a combination of assets can be found that all tend to react to the same economic influences the same as a stock of interest, then pairs trading could potentially be used to take advantage of this. In order to implement this, instead of daily data we would use either hourly or minute data to train our model. The downside of this is that because mean reversion modeling depends on the property of stationarity, we would also need to take a corresponding long or short position in the F-Rated stock involved in the short covering rally which itself is inherently risky. I would consider it to be risky to buy into a pairs trading scenario that would require one to short a stock that has recently undergone a jump up in its price. Potentially another play that would be possible would be to preemptively buy into an F-rated stock if one knows that important news or updates could lead to a jump in its price in the following days and use a pairs strategy to short a cointegrated asset as a hedge.

Another approach could be to pre-identify cointegrated pairs of assets including certain F-rated stocks of interest and once a short covering rally has occurred, one could monitor the model to see if the APR, half-life of mean reversion, and sharpe ratio look promising enough to enter either a short intraday trade or short interday trade so long as it is reasonable to believe that the asset or assets paired to the F-rated stock will react positively in conjunction with the stock of interest. It is important to also note that the F-rated stock must not be hard-to-borrow as the model will potentially suggest to short it with respect to the pairs strategy depending on the timing of the trader's entry. Another possible approach could also be to incorporate the F-rated stock into a unit portfolio to be paired against another stock or unit portfolio that can be cointegrated against to avoid being too heavily affected by potential losses from unexpected behavior following the short covering rally.

### Discussion of Strengths and Weaknesses

This strategy has strengths in its straightforward implementation as well as its flexibility in allowing for many different nuanced approaches such as in the diversity of generating weightings of hedge ratios (Johansen Test, Kalman Filter, Random Forest, etc.), possible transformations to the price spreads (regular price, log transformations, ratios, etc.), and different combinations of asset classes or number of assets paired. There are also a large variety of applications for mean reversion including long term, short term, and seasonal opportunities. Another positive is that although high frequency trading firms also have algorithms set up to act as market makers through this strategy, it is unlikely that everyone will set up their strategy exactly the same, which potentially leaves room for certain strategies to remain profitable; however, it has been noted that methods such as these have been less profitable than in the past possibly due to increased usage of this strategy [7]. Periods of volatility are when methods such as these should perform the best since volatility increases the spread between the cointegrated pairs; on the other hand, times of economic stability should have less opportunities.

This method's weakness primarily has to do with its requirement for cointegration. Futures and currencies, in particular, rarely exhibit cointegrating behavior. Identifying groups of stocks or assets that demonstrate cointegration can be tricky because two assets from the same market industry will not necessarily cointegrate even if their prices have strong correlation. Any pairs identified need to have transaction costs factored in, depending on how often portfolio rebalancing must be done. Additionally, this strategy is highly sensitive to any changes to fundamental factors affecting the prices of the underlying stocks if cointegration is negatively affected. It also can be dangerous during periods of extreme drops or drawdowns because during a sudden plunging of stock prices, the model will suggest to continue buying since it inherently assumes reversion to the mean. Additionally, a stop loss safeguard could be put in place; however, it must be set to either a high standard deviation level or a point where further losses can no longer be stomached. If the stop loss is set too close to the stock's price fluctuations, it could lead to a series of realized losses.

**Disclosures;**

Navellier & Associates does not own Quantumscape Corp (QS), or Canoo (GOEV). Louis Navellier and James Hopham do not personally own Quantumscape Corp (QS), or Canoo (GOEV).

**Works Cited**


1. Boguslavskaya, E., Boguslavsky, M., & Muravey, D. (2020). *Trading Multiple Mean Reversion*. Brunel University London, UK; TradeTeq, UK; Lomonosov State University, Moscow, Russia

    2009.09816.pdf

2. Chan, E. P. (2008). *Quantitative Trading: How to Build Your Own Algorithmic Trading Business*. Wiley.
3. Chan, E. P. (2013). *Algorithmic Trading: Winning Strategies and Their Rationale*. Wiley.
4. Guijarro-Ordonez, J., Pelger, M., & Zanotti, G. (2022, September 25). Deep Learning Statistical Arbitrage.

    2106.04028.pdf

5. Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and Practice* (3rd ed.). Otexts.
6. Kahneman, Daniel. *Thinking, Fast and Slow*. New York: Farrar, Straus and Giroux. 2011.
7. Khandani, Amir, and Andrew Lo. "What Happened to the Quants in August 2007?" Preprint, 2007. Available at http://web.mit.edu/alo/www /Papers/august07.pdf.
8. Lo, Andrew W., and A. Craig MacKinlay. *A Non-RandomWalk Down Wall Street*. Princeton, NJ: Princeton University Press, 2001.
9. Schoenberg, Ron, and Alan Corwin. "Does Averaging-in Work?" 2010.

## Example of Mean Reversion Strategy Implementation

January 25, 2024

**Python Code Implementation**

This program demonstrates an example of implementing a mean reversion strategy with the Johansen Test.

```
[2]: #import Libraries
import numpy as np
import pandas as pd
import statsmodels.formula.api as sm import
statsmodels.tsa.stattools as ts import
statsmodels.tsa.vector_ar.vecm as vm from
datetime import datetime
import yfinance as yf
```

```
[105]: #Create a function to return a dataframe for daily adjusted closing prices

def pull_prices(company_tickers):
    price_data= pd.DataFrame()
    ticker_list= list()

    for i in company_tickers:
        price_data= pd.concat([price_data,
                               pd.DataFrame(
                                   yf.download(i, start=datetime(2021, 1, 19),
                                               end=datetime(2024, 1, 19)
                                           ).iloc[:,4]
                                       )
                              ], axis = 1
                             )
        ticker_list.append(i)

    price_data.columns= ticker_list
    price_data['Date']= price_data.index
    return price_data
```

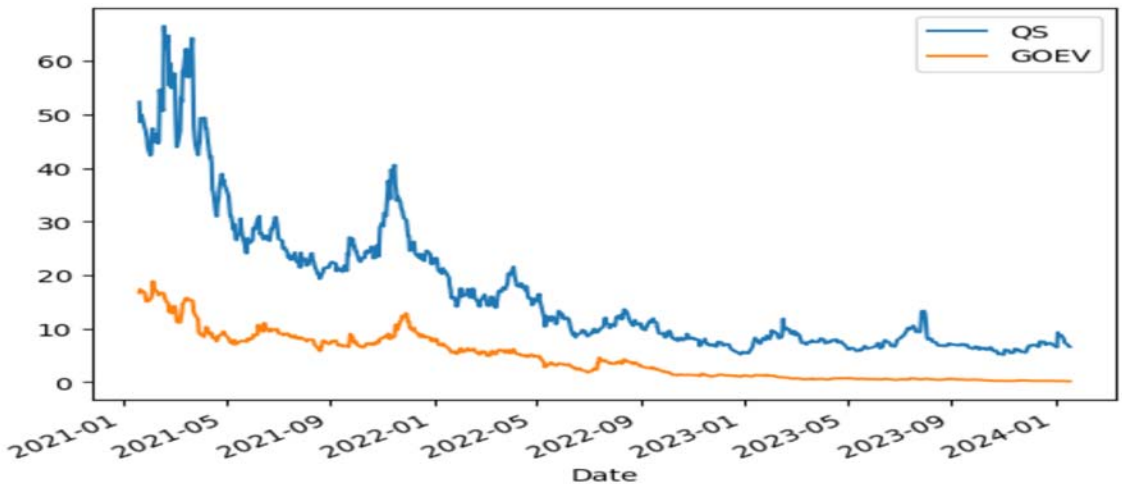
```
[106]: #Pull historical prices from 2021 to 2024
tickers= ["QS", "GOEV"]
df= pull_prices(tickers)
```

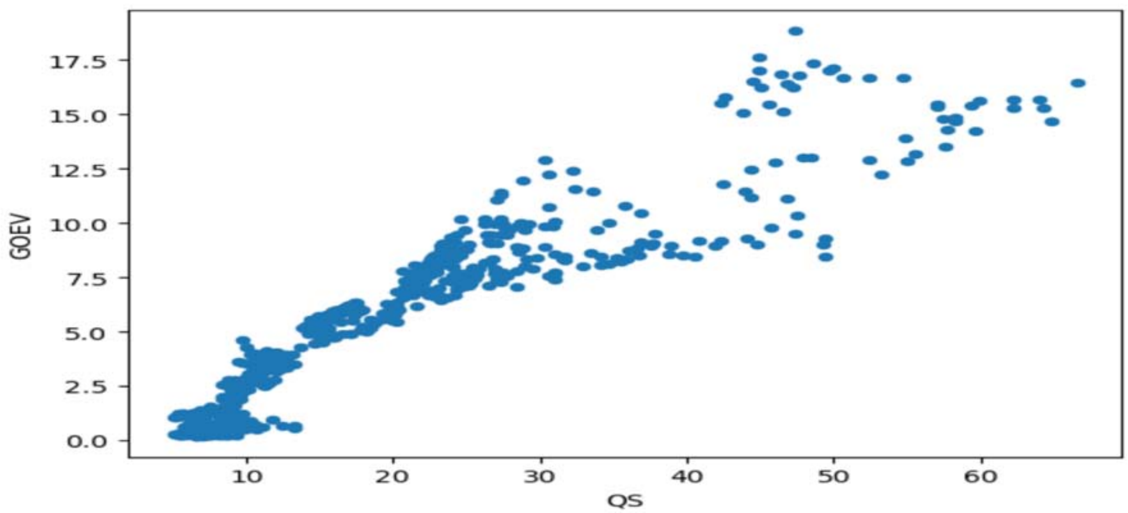
```
#Check data range
print(df.head())

print(df.tail())
```

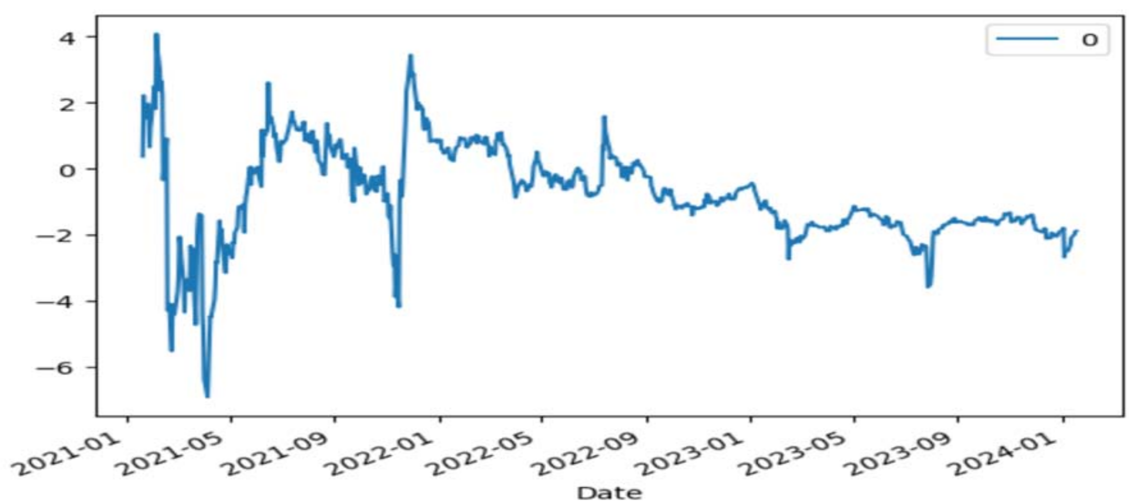
```
[*********************100%%***********************]  1 of 1 completed
[*********************100%%***********************]  1 of 1 completed
                  QS       GOEV       Date
Date
2021-01-19  52.360001  16.680000 2021-01-19
2021-01-20  48.590000  17.360001 2021-01-20
2021-01-21  50.000000  17.120001 2021-01-21
2021-01-22  49.680000  17.000000 2021-01-22
2021-01-25  47.630001  16.799999 2021-01-25
              QS   GOEV       Date
Date
2024-01-11  8.10  0.211 2024-01-11
2024-01-12  7.37  0.203 2024-01-12
2024-01-16  6.97  0.185 2024-01-16
2024-01-17  6.65  0.180 2024-01-17
2024-01-18  6.61  0.176 2024-01-18
```

[107]:
```
#Generate diagnostic plots
df.set_index('Date', inplace=True)

#Plot daily chart data
df.plot()

#Plot daily price correlation
df.plot.scatter(x='QS',
                y='GOEV')

#Visualize hedge ratios for cointegration
results= sm.ols(formula="GOEV ~ QS", data=df[['QS', 'GOEV']]).fit()
hedgeRatio= results.params[1]

pd.DataFrame((df['GOEV']-hedgeRatio*df['QS'])).plot()
```

[107]:
```
<Axes: xlabel='Date'>
```




Source: Price data from Yahoo Finance. Calculations and statistical tests were completed by author James Hopham in Python through both manual calculations and using open-source packages/tools. Graphs are for illustrative and discussion purposes only. Please read important disclosures at the end of this commentary.

[108]:
```
#Check for stationarity using the CADF test

coint_t, pvalue, crit_value=ts.coint(df['QS'], df['GOEV'])

print('t-statistic=%f' % coint_t)
print('\npvalue=%f' % pvalue)
print('\ncritical values')
print(crit_value)
```

```
t-statistic=-2.972421

pvalue=0.116907

critical values
[-3.91102404 -3.34424558 -3.05007972]
```

[109]:
```
#Conduct Johansen Test to generate eigenvalues and eigenvectors
result= vm.coint_johansen(df[['QS', 'GOEV']].values,
                          det_order=0, k_ar_diff=1
                         )

print('Eigenvalues:')
```

```
print(result.eig)

print('\nEigenvectors:')
print(result.evec)
```

```
Eigenvalues:
[0.020022   0.01042366]

Eigenvectors:
[[ 0.2260519  -0.05411617]
 [-0.69890475 -0.07344132]]
```

[110]:
```
#Calculate the half-life of mean reversion

#Net market value of portfolio
portfolio_value= pd.DataFrame(np.dot(df.values, result.evec[:, 0]))

#Lag values and differencing lag=
portfolio_value.shift()
differencing= portfolio_value-lag

#Fit a regression
df_regress= pd.concat([lag, differencing], axis=1)
df_regress.columns=['lag', 'differencing']
regression= sm.ols(formula="differencing ~ lag", data=df_regress).fit()
```

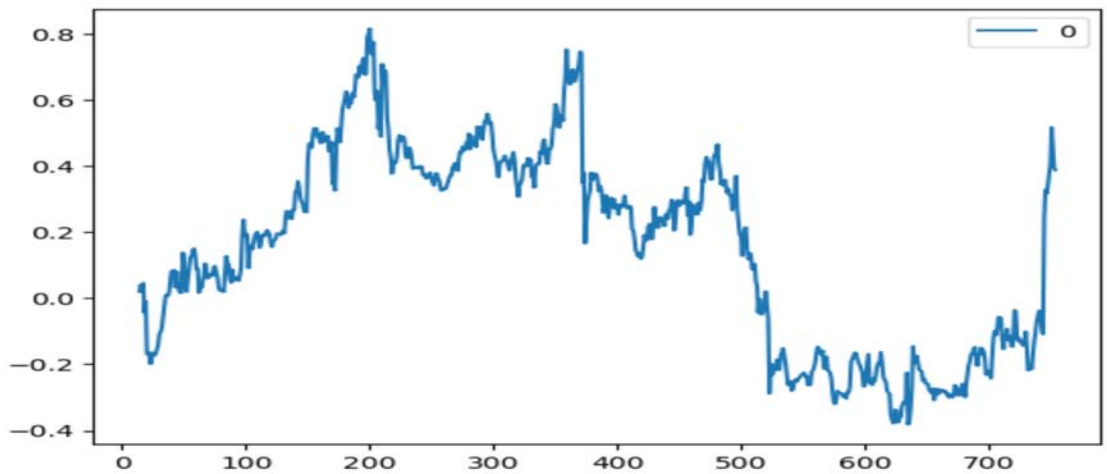
```
#Calculate Half-Life
half_life= -np.log(2)/regression.params['lag']
print('\nHalf-life to Mean Reversion= %f Days' % half_life)
```

```
Half-life to Mean Reversion= 14.452225 Days
```

[114]:
```
#round
round_half_life= np.round(halflife).astype(int)

#calculate portfolio weightings
portfolio_units = -(portfolio_value-portfolio_value.rolling(round_half_life).
 ↪mean()
          )/portfolio_value.rolling(round_half_life).std()

value_of_positions= pd.DataFrame(np.dot(portfolio_units.values,
                              np.expand_dims(
                                  result.evec[:, 0],
                                  axis=1).T)*df.values
                    )
```

```
#Daily P&L
pnl=np.sum((value_of_positions.shift().values)*(df.pct_change().values), axis=1)

#Returns
returns=pnl/np.sum(np.abs(value_of_positions.shift()), axis=1)

#Plot of Cumulative Returns
pd.DataFrame((np.cumprod(1+ret)-1)).plot()
print('APR=%f \nSharpe=%f' %
           (np.prod(1+returns)**(252/len(returns))-1,
            np.sqrt(252)*np.mean(returns)/np.std(returns)
           )
      )
```

```
APR=0.115786
Sharpe=0.482382
```




Source: Price data from Yahoo Finance. Calculations and statistical tests were completed by author James Hopham in Python through both manual calculations and using open-source packages/tools. Graphs are for illustrative and discussion purposes only. Please read important disclosures at the end of this commentary.

## IMPORTANT DISCLOSURES

Any holdings identified do not represent all of the securities purchased, sold, or recommended for advisory clients and it should not be assumed that investments in securities identified and described were or would be profitable. Performance results presented herein do not necessarily indicate future performance. Results presented include reinvestment of all dividends and other earnings. Investment in equity strategies involves substantial risk and has the potential for partial or complete loss of funds invested. Investment in fixed income components has the potential for the investment return and principal value of an investment to fluctuate so that an investor's shares, when redeemed, may be worth less than their original cost. It should not be assumed that any securities recommendations made by Navellier & Associates, Inc. in the future will be profitable or equal the performance of securities mentioned in this report.

This report is for informational purposes and is not to be construed as an offer to buy or sell any financial instruments and should not be relied upon as the sole factor in an investment making decision. The views and opinions expressed are those of Navellier at the time of publication and are subject to change. There is no guarantee that these views will come to pass. As with all investments there are associated inherent risks. Please obtain and review all financial material carefully before investing. Although the information in this communication is believed to be materially correct, no representation or warranty is given as to the accuracy of any of the information provided. Certain information included in this communication is based on information obtained from sources considered to be reliable. However, any projections or analysis provided to assist the recipient of this communication in evaluating the matters described herein may be based on subjective assessments and assumptions and may use one among alternative methodologies that produce different results. Accordingly, any projections or analysis should not be viewed as factual and should not be relied upon as an accurate prediction of future results. Furthermore, to the extent permitted by law, neither Navellier nor any of its affiliates, agents, or service providers assumes any liability or responsibility nor owes any duty of care for any consequences of any person acting or refraining to act in reliance on the information contained in this communication or for any decision based on it. Opinions, estimates, and forecasts may be changed without notice. The views and opinions expressed are provided for general information only.

The S&P 500 Index consists of 500 stocks chosen for market size, liquidity and industry group representation. It is a market value weighted index with each stock 's weight in the index proportionate to its market value. The reported returns reflect a total return for each quarter inclusive of dividends. Presentation of index data does not reflect a belief by Navellier that any stock index constitutes an investment alternative to any Navellier equity strategy presented in these materials, or is necessarily comparable to such strategies and an investor cannot invest directly in an index. Among the most important differences between the indexes and Navellier strategies are that the Navellier equity strategies may (1) incur material management fees, (2) concentrate investments in relatively few ETFs, industries, or sectors, (3) have significantly greater trading activity and related costs, and (4) be significantly more or less volatile than the indexes. All indexes are unmanaged and performance of the indices includes reinvestment of dividends and interest income, unless otherwise noted, are not illustrative of any particular investment and an investment cannot be made in any index.

FactSet Disclosure: Navellier does not independently calculate the statistical information included in the attached report. The calculation and the information are provided by FactSet, a company not related to Navellier. Although information contained in the report has been obtained from FactSet and is based on sources Navellier believes to be reliable, Navellier does not guarantee its accuracy, and it may be incomplete or condensed. The report and the related FactSet sourced information are provided on an "as is" basis. The user assumes the entire risk of any use made of this information. Investors should consider the report as only a single factor in making their investment decision. The report is for informational purposes only and is not intended as an offer or solicitation for the purchase or sale of a security. FactSet sourced information is the exclusive property of FactSet. Without prior written permission of FactSet, this information may not be reproduced, disseminated or used to create any financial products. All indices are unmanaged and performance of the indices include reinvestment of dividends and interest income, unless otherwise noted, are not illustrative of any particular investment and an investment cannot be made in any index. Past performance is no guarantee of future results.

Mapsignals Disclosure: Jason Bodner is a co-founder and co-owner of Mapsignals.com, a Developed Factor Model for isolating outlier stocks using its proprietary quantitative equity selection methodology. Mapsignals was founded in 2014. Data used by Mapsignals, for periods prior to its founding in 2014, is data derived from Factset.

Mr. Bodner is an independent contractor who is occasionally hired to write articles and provide his editorial comments and opinions. Mr. Bodner is not employed by Navellier & Associates, Inc., or any other Navellier owned entity. The opinions and statements made in this article are those of Mr. Bodner and not necessarily those of any other persons or entities. Jason Bodner is a co-founder and co-owner of Mapsignals. Mr. Bodner is an independent contractor who is occasionally hired by Navellier & Associates to write an article and or provide opinions for possible use in articles that appear in Navellier & Associates weekly Market Mail. Mr. Bodner is not employed or affiliated with Louis Navellier, Navellier & Associates, Inc., or any other Navellier owned entity. The opinions and statements made here are those of Mr. Bodner and not necessarily those of any other persons or entities. This is not an endorsement, or solicitation or testimonial or investment advice regarding the BMI Index or any statements or recommendations or analysis in the article or the BMI Index or Mapsignals or its products or strategies.

The views and opinions expressed do not constitute specific tax, legal, or investment or financial advice to, or recommendations for, any person, and the material is not intended to provide financial or investment advice and does not take into account the particular financial circumstances of individual investors. Before investing in any investment product, investors should consult their financial or tax advisor, accountant, or attorney with regard to their specific situation.

Please note that Navellier & Associates and the Navellier Private Client Group are managed completely independent of the newsletters owned and published by InvestorPlace Media, LLC and written and edited by Louis Navellier, and investment performance of the newsletters should in no way be considered indicative of potential future investment performance for any Navellier & Associates product.