**GROUP WORK PROJECT #**2
**Group Number:** 8564

MScFE 610: FINANCIAL ECONOMETRICS

| FULL LEGAL NAME | LOCATION (COUNTRY) | EMAIL ADDRESS | MARK X FOR ANY NON-CONTRIBUTING MEMBER |
|---|---|---|---|
| Guo Yuxuan | Singapore | y.xuannn03@gmail.com | |
| Franck Delma Deba Wandji | France | debafranck@gmail.com | |
| | | | |

| **Statement of integrity:** By typing the names of all group members in the text boxes below, you confirm that the assignment submitted is original work produced by the group (excluding any non-contributing members identified with an "X" above). | |
|---|---|
| **Team member 1** | **Guo Yuxuan** |
| **Team member 2** | **Franck Delma Deba Wandji** |
| **Team member 3** | |

| Use the box below to explain any attempts to reach out to a non-contributing member. Type (N/A) if all members contributed.<br>**Note:** You may be required to provide proof of your outreach to non-contributing members upon request. |
|---|
| N/A |

**Challenge 2: Modeling non-stationarity and finding an equilibrium**

In this challenge, we aim to calibrate a Vector Error Correction Model (VECM), which requires the presence of cointegrated time series. As highlighted in the literature, financial time series—particularly stock prices—are often non-stationary but may exhibit long-run equilibrium relationships, making them suitable candidates for cointegration analysis.

1. **Definition**

Vector error correlation models are related to vector autoregressive models (VARs). Given a n-dimensional of cointegrated time series $X_t = \left(X_{\{1,t\}}, X_{\{2,t\}}, \ldots, X_{\{n,t\}}\right)$

the VECM is specified as:

$$\Delta X_t = C + \alpha\beta' X_{\{t-1\}} + \sum_{\{i=1\}}^{\{p-1\}} \Gamma_i \Delta X_{\{t-i\}} + \epsilon_t$$

Where

- $C$ is the deterministic vector or matrix depending on whether there are constants and/or linear trend

- $\beta$ is the cointegrating vector (long-run equilibrium relationship)

- $\alpha$ Speed of adjustment coefficients (how fast variables revert to equilibrium)

- $\Gamma_i$ is a $n \times n$ matrix for coefficients of lagged differences of $X$ (short-term dynamic coefficients)

- $\epsilon_t$ is a white noise $n \times 1$ vector with 0 means and stable covariance

2. Description

Adjustment vectors contain the coefficients that can bring the short-term deviation of the time series back to their long-term equilibrium relationship. The coefficients in adjustment vectors are error correction coefficients. Therefore, =α β' corrects the error gap that happened in the last period and brings short-term disequilibrium among time series back to their long-term equilibrium (Hamilton, 1994).

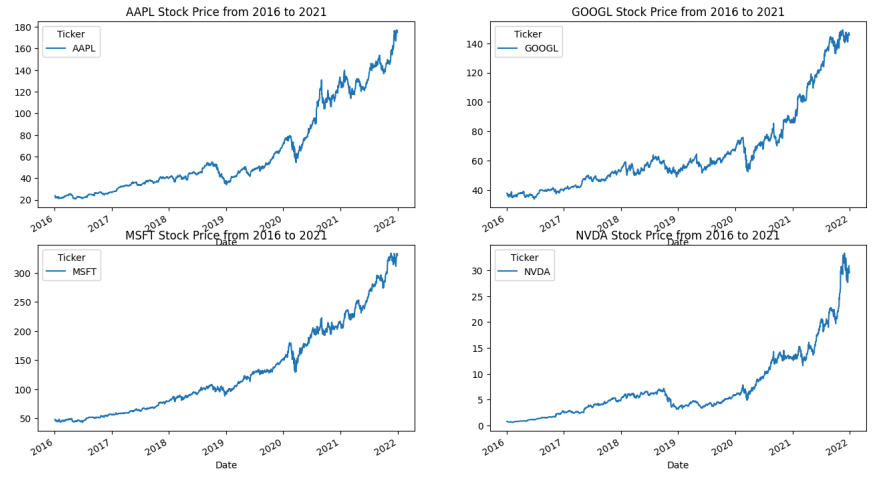
3. Demonstration

3.1. Data

We consider daily adjusted stock prices for four major technology companies: Google (GOOGL), Apple (AAPL), Microsoft (MSFT), and Nvidia (NVDA), spanning the period from 2016 to 2021. These firms operate within the same sector and are likely influenced by

common macroeconomic and industry-specific factors, which may induce cointegrated behavior among their price series.

Figure 1: Stock prices from 2016 to 2021




### 3.2. Testing for stationarity

Figure 2: ACF of stock prices

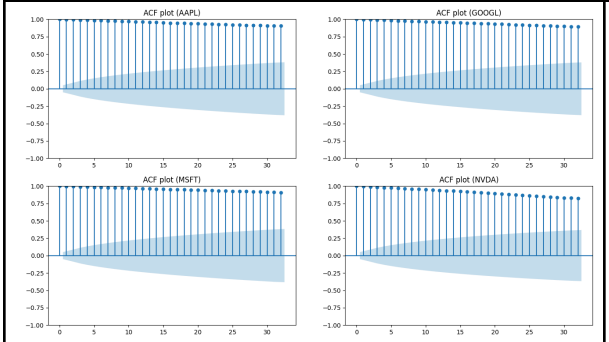
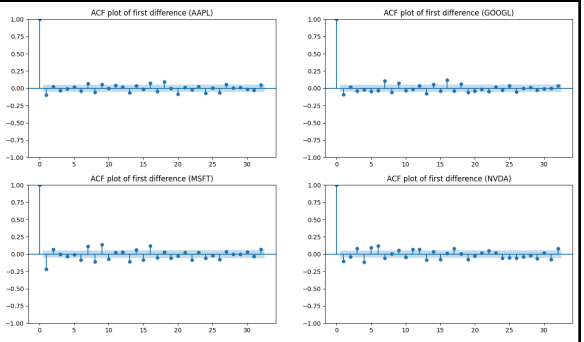



Figure 3: ACF of 1st stock prices difference




The autocorrelation function (ACF) of the raw stock price series exhibits a gradual decay across lags (Figure 2), indicating persistent trends in the price movements—a hallmark of non-stationary behavior. In contrast, the ACF of the first-differenced series (Figure 3) displays no significant autocorrelations, with all lags falling within the confidence bounds of a white noise process. This sharp divergence between the level and differenced series strongly suggests that the stock prices are integrated of order one I(1), where the initial non-stationarity is eliminated after first differencing.

To formally test stationarity, we often use the Augmented Dickey-Fuller (ADF) test and Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test.

ADF Test hypotheses: - $H_0$: The time series is non-stationary (unit root), - $H_1$: The time series is stationary (Dickey & Fuller, 1979). Decision rule: Reject $H_0$ if Test stat < critical value.

KPSS Test hypotheses: - $H_0$: The time series is stationary (around a level or trend), - $H_1$: The time series is non-stationary (Kwiatkowski et al., 1992). Decision rule: Reject $H_0$ if Test stat > critical value.

These are complementary tests that have “opposite” null hypotheses.

Table 2: ADF and KPSS tests summary

| **Test** | **Test statistic** | **AAPL** | **GOOGL** | **MSFT** | **NVDA** |
|---|---|---|---|---|---|
| ADF | Level | 3.518 | 3.044 | 4.135 | 3.138 |
| | 1st Diff | -42.425 | -42.370 | -47.944 | -13.583 |
| | 5% Critical Value | -1.941 | -1.941 | -1.941 | -1.941 |
| KPSS | Level | 1.294 | 1.055 | 1.309 | 0.994 |
| | 1st Diff | 0.044 | 0.08 | 0.030 | 0.124 |
| | 5% Critical Value | 0.147 | 0.147 | 0.147 | 0.147 |
| Décision | | I(1) | I(1) | I(1) | I(1) |

Table 2 shows that Apple, Google, Microsoft and Nvidia stock prices are I(1) with support for both ADF and KPSS tests.

3.3. Testing for cointegration

Cointegration tests are used to determine whether two or more time series, which are individually I(1), have a long-term equilibrium relationship(Engle & Granger, 1987). This is particularly relevant when analyzing technology stock prices, as these stocks may exhibit common trends in the long run, even though their short-term behaviors might diverge due to volatility, market reactions, or other temporary factors.

If time series are cointegrated, it implies that there is a long-term equilibrium relationship between them. In other words, despite short-term fluctuations and individual trends, the series tend to move together over time.

The Johansen cointegration test is one of the most popular methods for testing for cointegration. It provides a framework to test for the number of cointegrating relationships in a multivariate setting. The Johansen test works under the assumption that the time series are I(1) and tests whether there exists a linear combination of these series that is stationary.

One key step in performing the Johansen cointegration test is to specify the number of lags to be used in the Vector Autoregression (VAR) model. The lag length is important because it determines the number of past values of the series included in the model, which influences the dynamics and the results of the cointegration test.

To determine the appropriate number of lags, a lag selection procedure is performed. This is typically done using VAR specification, where the optimal lag length is selected based on various criteria, such as AIC and BIC. We selected the optimal lag length based on BIC (1 lag).

| | **Test statistic** | **Critical values (90%)** | **Critical values (95%)** | **Critical values (99%)** | **Decision** |
|---|---|---|---|---|---|
| rank=0 | 58.839 | 44.492 | 47.854 | 54.681 | Reject H0 |
| rank<=1 | 28.188 | 27.066 | 29.796 | 35.462 | Fail to reject H0 |
| rank<=2 | 7.377 | 13.429 | 15.494 | 19.934 | Fail to reject H0 |
| rank<=3 | 0.199 | 2.705 | 3.841 | 6.634 | Fail to reject H0 |

At the first step of the procedure, we test (H0): rank = 0 vs (H1): rank >= 1. since the test statistic is greater than the critical value (5% significance level), (H0) is rejected. Thus, there is at least one cointegrating relationship. The step aims to test (H0): rank = 1 vs (H1): rank >= 2. Since the test statistic is lower than the critical value (5% significance level), then (H0) is not rejected. We conclude that there is exactly one cointegrating relationship. This means that the stock prices of Apple, Google, Microsoft and Nvidia have a long-term equilibrium relationship—they are cointegrated with exactly one cointegrating vector.

3.4. Calibrating VECM parameters

The calibrated VECM model's parameters with one lag specification is given as follows:

Figure 4: Long-term equilibrium


$$S_t = -16.5637 + 1 * AAPL\_t + 0.6624 * GOOGL\_t -$$

Short term dynamic ($\Gamma_1$)

| | | | |
|---|---|---|---|
| -0.0299 | -0.0057 | 0.0440 | -0.0063 |
| 0.0570 | 0.0193 | 0.0155 | 0.0052 |
| -0.1050 | -0.0973 | -0.2846 | -0.0151 |
| 0.3324 | 0.4488 | 0.7323 | -0.0272 |

**Model Validation:** Stationarity Tests of cointegration relation. With an ADF test statistic -2.084 < 5% critical value (-1.94) we reject the null hypothesis of unit root in $S_t$. However this is not confirm with the KPSS test which rejects stationarity of $S_t$ around a trend.

**Short term dynamics:** The negative short-term coefficient for Microsoft (-0.2846 on its own lag in the $\Gamma_1$ matrix) reveals two key dynamics:

- The significant negative autoregressive coefficient suggests MSFT exhibits self-correcting behavior - a 1% price increase today typically leads to a 0.28% decrease tomorrow, all else equal. This contrasts with NVDA's strong positive momentum (0.7323 coefficient).
- The consistently negative cross-effects (e.g., -0.1050 on AAPL, -0.0973 on GOOGL) imply MSFT acts as a stabilizing force in the tech portfolio. When other stocks diverge from equilibrium, MSFT adjusts inversely to restore balance, consistent with its mature enterprise software business model which tends to be less volatile than consumer tech or semiconductors.

**Interpretation of the cointegration vector** $\beta = (1, 0.6624, -0.5198, -3.6798)'$: A 1% increase in AAPL requires 0.66% increase in GOOGLE, 0.52% decrease in MSFT and 3.68% decrease in NVDA. Normalization on AAPL: The coefficient of 1 on AAPL establishes it as the benchmark stock in this relationship. GOOGLE moves in the same

direction as AAPL but with lower elasticity, suggesting AAPL is the sector leader while MSFT and NVDA show inverse relationships, potentially acting as hedging instruments. The large coefficient (-3.6798) indicates NVDA requires disproportionate movements to maintain equilibrium, likely due to its distinct growth characteristics in the semiconductor space.

**Interpretation of the speed adjustment coefficient error correction mechanism**

$\alpha = (-0.0188, -0.0036, -0.0132, -0.0024)'$: Apple dominates correction with the largest alpha (-0.0188), Apple bears most of the adjustment burden. The small alpha for Nvidia (-0.0024) suggests its price is less responsive to equilibrium deviations, possibly due to stronger momentum trading. All negative coefficients satisfy the error correction condition confirming mean-reversion (Engle & Granger, 1987).

3.6. Damage

Volatility Clustering (ARCH effects in residuals): It looks like the variance depends on time. The kurtosis shows an excess kurtosis of 10.844 meaning that the long-run relation exhibits fat tails.

Structural Breaks (2020 COVID shock): it seems to be a regime changing around 2020.

3.7. Directions

- log transformation: These issues might be resolved by stabilizing variance with a log transformation of the series. When doing that on our data set, the Johansen test results show an absence of cointegration between log series, despite the are I(1). In that situation, a VAR model on the first difference is more suitable to model the log series.

- GARCH-VECM: Model volatility explicitly

- Regime-Switching VECM: Capture structural breaks

- Shorter Time Window: Reduce impact of regime shifts

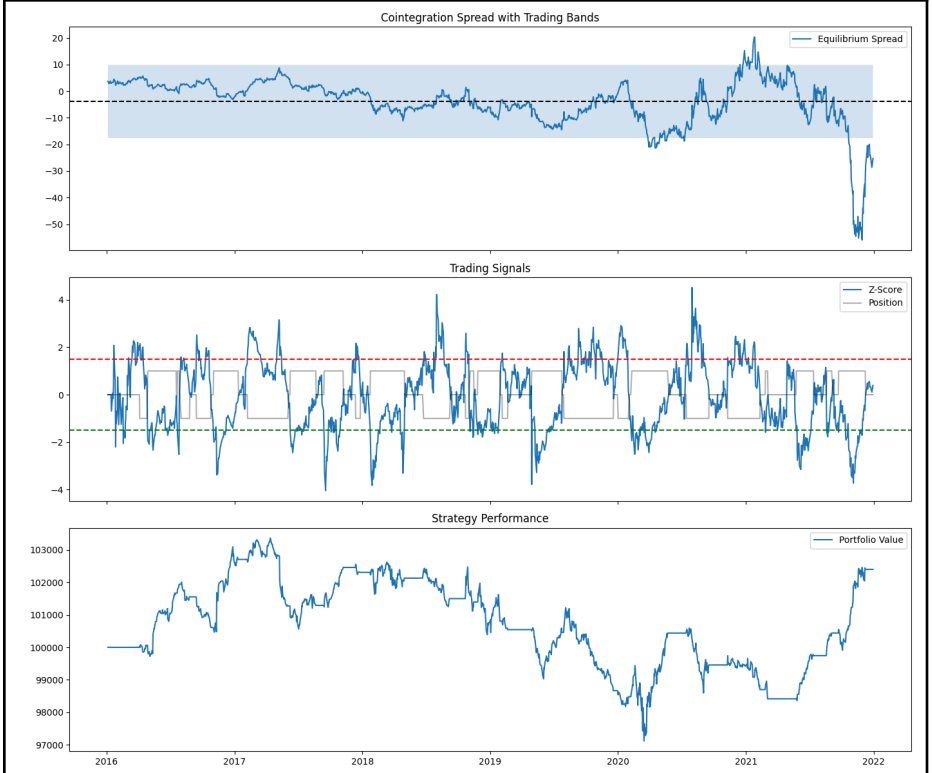
3.8. Deployment

- The identified cointegration Validates a long-run equilibrium relationship among the stocks and thus Suggests shared exposure to common risk factors. Justifies pairs trading strategies (Vidyamurthy, 2004).

- Risk Management: - Monitor equilibrium for regime shifts.- Hedge tech exposure during large deviations

- Asset Allocation: - Dynamic weights based on rolling β. - Rebalance when equilibrium breaks down.

Figure 5: Deployment for a pairs trading (backtesting)



**Challenge 3**

In this challenge, we aim to calibrate the multiple linear regression model to better handle datasets with the presence of multicollinearity.

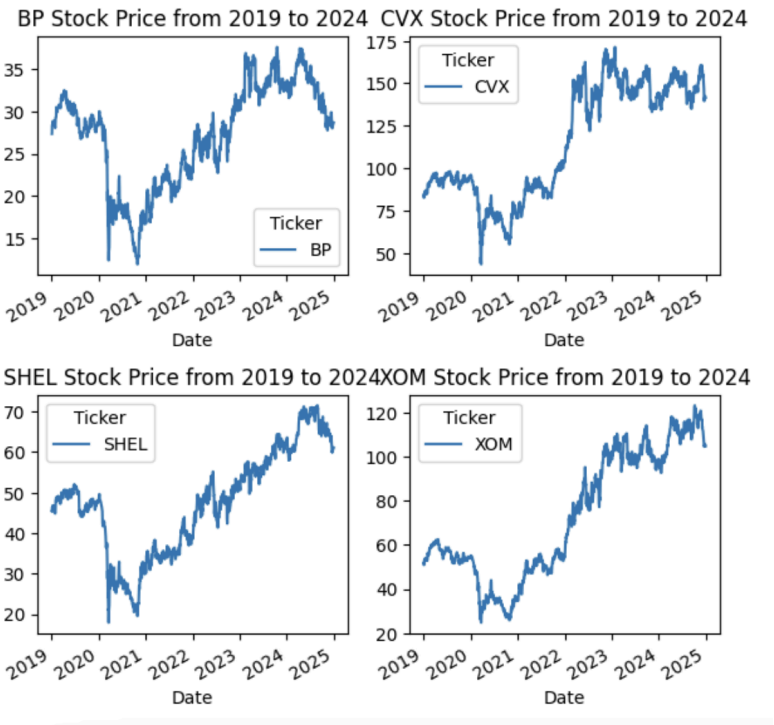
## 2. Description

Multicollinearity occurs when the predictors in a regression model are highly correlated. This makes it difficult to analyse what is the impact each individual predictor has on the dependent variable, causing model results to be unreliable and unstable.

## 3. Demonstration

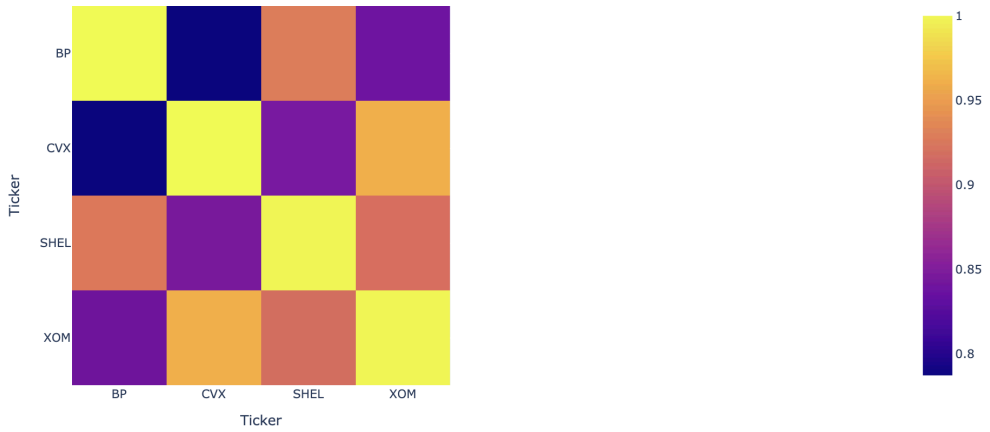
### 3.1 Data

We consider the daily outright stock prices from 2019 to 2024 for the big 4 oil majors – BP, ExxonMobil, Shell and Chevron. These firms operate in the same market, exposed to similar market influences. Hence they will be affected in the same way from market movements, causing their stock prices to be correlated. We will be doing formal tests to test for the multicollinearity between the stock price.



3.2 Test for multicollinearity

A naive approach to testing for multicollinearity is to look at the correlation matrix between the stock prices.




A correlation close to +1 or -1 would indicate a strong linear relationship between the two stocks, suggesting potential multicollinearity. As seen in the heatmap, all 4 stocks are highly correlated with each other, with a correlation coefficient of at least 0.8 between any two stocks.

We can then calculate the variance inflation factor (VIF) for the stocks. As mentioned previously, the presence of multicollinearity causes parameter coefficient estimates to be unreliable due to higher variances. VIF is a metric commonly used to detect multicollinearity in data as it quantifies how much of the variance is inflated due to the presence of multicollinearity.

For each $X_i$, VIF is calculated to be: $VIF(X_i) = \frac{1}{1 - R_i^2}$, where $R_i^2$ is the $R^2$ obtained by running a regression model of each $X_i$ on the other predictors to isolate its effect.

| | Variable | VIF |
|---|---|---|
| **0** | BP | 151.429249 |
| **1** | CVX | 107.347301 |
| **2** | SHEL | 213.993023 |
| **3** | XOM | 101.026498 |

Ideally, VIF should be around 1, which suggests that there is no multicollinearity in the data and gives us confidence to continue using them as predictors in the regression model. Any VIF greater than 10 would signify that there is very high multicollinearity. Hence, the VIF calculated for our group of stock prices would point to very high multicollinearity.

4. Directions

4.1 Ridge and Lasso regression

The ordinary least squares approach is unable to handle multicollinearity effectively, leading to unreliable parameter estimates and poor analysis. We could instead use regularization methods like Lasso regression (L1 regularization) or Ridge regression (L2 regularization), which are more robust by introducing penalty terms that constrain the model coefficients.

Ridge regression fine tunes the classic OLS model to better handle multicollinearity by introducing a L2 penalty term to the loss function, called the sum of squared coefficients.

$\min_{\beta} \left[ \sum (y_i - \hat{y}_i)^2 + \alpha \sum \beta_j^2 \right]$ , where $\alpha$ is the strength of the penalty. If $\alpha$ = 0, then the regression becomes a ordinary least squares regression and there is no penalty on the parameter (Schreiber-Gregory, 2018). As variance tends to be inflated due to the presence of multicollinearity, the L2 penalty works against that and instead shrinks the coefficients closer to zero, reducing their variance. It thereby distributes the influence among correlated variables without excluding any of them from the model, making it effective to model data containing multicollinearity.

Lasso regression works slightly differently where it introduces the L1 penalty term, the sum of absolute coefficients, to the loss function (Schreiber-Gregory, 2018). Unlike ridge, Lasso regression is a feature selection model where it forces some coefficients to be zero. This is particularly useful in datasets that contain multicollinearity as lasso would tend to only retain the most influential predictor in a group of highly correlated variables. This not only handles the multicollinearity, but simplifies the model and improves the reliability of the model estimates.

As these two models work very differently and for different goals for the model, there are also hybrid approaches, known as ElasticNet. This method seeks to combine the L1 and L2 regularization and is effective when the predictors are highly correlated (Schreiber-Gregory, 2018).

### 4.1 Principal Component Analysis

Apart from fine tuning the model, we can also manipulate the multicollinear data instead by using Principal Component Analysis. Principal component analysis (PCA) is a dimensionality reduction method that is especially useful for analysing highly correlated datasets. PCA allows us to identify and simplify this group of stocks to just a few common factors that are driving asset returns while keeping the most information that are able to explain as much of the variation of the stocks as possible (Tsay, 2010). We can then run these common factors through regression models and the resulting models will not face any issues of multicollinearity as these factors are independent by construction.

## 5. Deployments

We demonstrate how we can improve the OLS model with the aforementioned methods.

The mean squared errors is used as the metric to measure the model performance and compare between any two methods. A lower MSE would suggest that the forecast generated by the model is closer to the actual values of the model, giving more confidence of the accuracy of the results.

We first split the data into a train-test split as the ordinary least squares model is always optimized to give the best fit line in the data that it is trained on, hence we do not expect that the MSE for ridge or lasso would be significantly better than that of the OLS model, even in the presence of multicollinearity. Due to the bias-variance tradeoff, ridge and lasso add in bias in order to reduce the variance inflated by the highly correlated factors. Hence, insample MSE would perform poorer than OLS. However, due to their ability to better handle multicollinearity, they would be able to produce lower out of sample mse.

```
Linear Regression MSE: 64.8841451715274
Ridge Regression MSE: 64.88295845332546
Lasso Regression MSE: 64.82028732992939
```

We then applying PCA onto our predictors first before using that as an input into a OLS model. As can be seen from the following results, the PCA method caused the MSE to significantly decrease, as compared to the OLS model, indicating that it is better able to handle multicollinearity, thus resulting in a more reliable model.

```
Linear Regression MSE: 64.8841451715274
PCA Linear Regression MSE: 11.367463277287827
```

**References:**

1. Dickey, D. A., & Fuller, W. A. (1979). Distribution of the Estimators for Autoregressive Time Series With a Unit Root. Journal of the American Statistical Association, 74(366), 427–431.
2. Engle, R. F., & Granger, C. W. J. (1987). Co-Integration and Error Correction. Econometrica, 55(2), 251–276.
3. Hamilton, J. D. (1994). Time Series Analysis. Princeton University Press.
4. Kwiatkowski, D., Phillips, P. C. B., Schmidt, P., & Shin, Y. (1992). Testing the Null Hypothesis of Stationarity. Journal of Econometrics, 54(1-3), 159–178.
5. Vidyamurthy, G. (2004). Pairs Trading: Quantitative Methods and Analysis. Wiley
6. Schreiber-Gregory, Deanna. (2018). Regulation Techniques for Multicollinearity: Lasso, Ridge, and Elastic Nets.
7. Tsay, Ruey S. Analysis of Financial Time Series. 1st ed. Wiley Series in Probability and Statistics. Wiley, 2010. https://doi.org/10.1002/9780470644560.