MAN 456 - Homework 2 / 3

28.04.2025

Damla Uçar

22103276

Question 1

1.a

The hourly price data from Binance was downloaded using the previous code. I also observed different curriencies to be used in the following model and the structure of the downloaded data.

1.b

The time period selected for the analysis was from January 1, 2023, to January 1, 2025. The dataset was split into training and testing sets based on the date September 1, 2024. Data before this date was used for training, while data from September 1, 2024, onwards was used for testing. As a result, approximately 84% of the data was allocated to the training set and 16% to the testing set.

1.c

First, I identified a set of candidate cryptocurrencies by downloading historical data for BNBUSDT, SOLUSDT, LTCUSDT, XRPUSDT, DOGEUSDT, ALPHAUSDT, and ALPINEUSDT. In order to ensure the suitability of the time series data for modeling, I conducted stationarity tests using Augmented Dickey-Fuller (ADF), Phillips-Perron (PP), and Kwiatkowski-Phillips-Schmidt-Shin (KPSS) tests. For each series, I first tested stationarity in levels and then again after taking the first difference. The outcomes of these tests can be seen below.

Crypto	ADF p-value	PP p-value	KPSS p-value	ADF p-value (diff)	PP p-value (diff)	KPSS p-value (diff)
BNBUSDT	0.8197 (Non-stationary)	0.8153 (Non-stationary)	0.01 (Non-stationary)	0.0 (Stationary)	0.0 (Stationary)	0.1 (Stationary)
SOLUSDT	0.7969 (Non-stationary)	0.7707 (Non-stationary)	0.01 (Non-stationary)	0.0 (Stationary)	0.0 (Stationary)	0.1 (Stationary)
LTCUSDT	0.0546 (Non-stationary)	0.0437 (Stationary)	0.01 (Non-stationary)	0.0 (Stationary)	0.0 (Stationary)	0.1 (Stationary)
XRPUSDT	0.0215 (Stationary)	0.0158 (Stationary)	0.01 (Non-stationary)	0.0 (Stationary)	0.0 (Stationary)	0.1 (Stationary)
DOGEUSDT	0.3763 (Non-stationary)	0.3541 (Non-stationary)	0.01 (Non-stationary)	0.0 (Stationary)	0.0 (Stationary)	0.1 (Stationary)
ALPHAUSDT	0.1142 (Non-stationary)	0.0957 (Non-stationary)	0.01 (Non-stationary)	0.0 (Stationary)	0.0 (Stationary)	0.1 (Stationary)
ALPINEUSDT	0.3883 (Non-stationary)	0.1828 (Non-stationary)	0.01 (Non-stationary)	0.0 (Stationary)	0.0 (Stationary)	0.1 (Stationary)

The criteria for selection required the cryptocurrency to be non-stationary in levels but stationary after the first differencing. Based on these tests, I selected the following cryptocurrencies for further analysis: BNBUSDT, SOLUSDT, ALPHAUSDT, ALPINEUSDT, and DOGEUSDT.

1.d

After selecting the cryptocurrencies, I followed the Engle–Granger two-step method as described in the referenced paper. For the first step of the Engle-Granger method, I conducted an ordinary least squares (OLS) regression. “BNBUSDT” was chosen as the dependent variable and the other currencies were used as independents variables. From the OLS regression, the following result was found.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                BNBUSDT   R-squared:                       0.882
Model:                            OLS   Adj. R-squared:                  0.882
Method:                 Least Squares   F-statistic:                 2.744e+04
Date:                Tue, 29 Apr 2025   Prob (F-statistic):               0.00
Time:                        18:57:35   Log-Likelihood:                -77659.
No. Observations:               14638   AIC:                         1.553e+05
Df Residuals:                   14633   BIC:                         1.554e+05
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         97.4192      2.025     48.109      0.000      93.450     101.388
SOLUSDT        1.0592      0.018     59.924      0.000       1.025       1.094
ALPHAUSDT  -1625.3978     25.208    -64.479      0.000   -1674.809   -1575.986
ALPINEUSDT    51.7966      1.361     38.069      0.000      49.130      54.463
DOGEUSDT    2617.5381     30.450     85.961      0.000    2557.852    2677.224
==============================================================================
Omnibus:                      184.660   Durbin-Watson:                   0.006
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              176.084
Skew:                          -0.237   Prob(JB):                     5.81e-39
Kurtosis:                       2.747   Cond. No.                     7.90e+03
==============================================================================
```

The model resulted with a 0.882 R-Squared value which indicates a good fit. The F-statistic also suggest that the model is statistically significant overall. All predictors were also found statistically significant from their P values. After estimating the relationship, the residuals were extracted from the model.

In the second step, I tested the stationarity of the residuals using ADF, PP, and KPSS tests. The objective was to ensure that the residuals did not have a unit root, confirming that a long-term equilibrium relationship existed among the selected cryptocurrencies.

Test	Test Statistic	p-value	Stationarity Conclusion
ADF	-3.4715	0.0088	✅ Stationary (reject H_0 of unit root)
Phillips-Perron (PP)	-3.804	0.0029	✅ Stationary (reject H_0 of unit root)
KPSS	2.7122	0.01	❌ Non-stationary (reject H_0 of stationarity)

Above results were found from the tests. Even though the KPSS test seems to reject the null hypothesis of stationary, produced an interpolation warning indicating that the test statistic exceeded the bounds of the lookup table. Therefore, the reported p-value is an upper bound, and the actual p-value is even smaller. So it can be concluded that each test proved the stationarity.

Therefore, all three stationarity tests confirmed the existence of cointegration among BNBUSDT, SOLUSDT, ALPHAUSDT, ALPINEUSDT, and DOGEUSDT. Consequently, the conditions required for constructing a cointegrated cryptocurrency portfolio were satisfied.

Therefore, the portfolio selected by part 1 is shown below:

Asset	Symbol	Coefficient (Weight)
Intercept	-	7.1183
Solana	SOLUSDT	0.0399
Alpha Finance	ALPHAUSDT	-0.4303
Alpine F1	ALPINEUSDT	0.2069
Dogecoin	DOGEUSDT	1.0738

Question 2

Following the Engle–Granger two-step cointegration approach, the spread S_t representing a market-neutral portfolio at time t, was constructed as:

$$St = BNBUSDTt - b1 \cdot ALPHAUSDTt - b2 \cdot ALPINEUSDTt - b3 \cdot DOGEUSDTt$$

where the coefficients b1, b2, and b3 were estimated from the OLS regression in the Engle–Granger cointegration test. As demonstrated in part 1, the residuals from this regression were found to be stationary according to the ADF, KPSS, and PP tests, which confirms the existence of a cointegrating relationship among the selected cryptocurrencies. Hence, this spread series will be used to define a trading strategy and simulate profit and loss over the test period.

Following the backtesting methodology outlined in Section 4 of Leung and Nguyen (2019), the trading rule is based on mean-reversion behavior of the spread. Specifically, multiple combinations of entry and exit thresholds were assessed by varying a threshold multiplier to find a profitable and stable configuration.

The trading strategy uses mean-reversion behavior in the spreads. A long is taken when $St < \mu - c\sigma$, and a short position is taken $when\ St > \mu + c\sigma$ where μ and σ are the historical mean and standard deviation of the spread, and c is a threshold multiplier. No trades occur while the spread remains within the range $[\mu - c\sigma, \mu + c\sigma]$, reducing noise and avoiding overtrading.

Firsly in this analysis, Profit and loss (P&L) were computed for the train set regarding each threshold level [0.5, 0.75, 1.0, 1.25, 1.5]. The graph for the Z-score and the thresholds is seen in the graph below. This shows how the trading would differ for different thresholds.

After observing the graph, the metrics of the thresholds were found. In the following table, especially the net trading PnL, Sharpe Ratio and Profit to MaxDraw was assessed.

Metric	Threshold 0.50	Threshold 0.75	Threshold 1.00	Threshold 1.25	Threshold 1.50
Num.Trades	2.0	2.0	2.0	1	1
Net.Trading.PL	11.908383	46.89495	120.927439	2.432805	40.591794
Avg.Trade.PL	5.954191	23.447475	60.46372	2.432805	40.591794
Largest.Winner	112.415249	112.415249	150.960463	2.432805	40.591794
Largest.Loser	-100.506866	-65.520299	-30.033024	2.432805	40.591794
Gross.Profits	112.415249	112.415249	150.960463	2.432805	40.591794
Gross.Losses	-100.506866	-65.520299	-30.033024	0.0	0.0
Percent.Positive	50.0	50.0	50.0	100.0	100.0
Percent.Negative	50.0	50.0	50.0	0.0	0.0
Sharpe Ratio	0.887835	4.18373	10.606258	0.0	0.0
Max.Drawdown	100.506866	65.520299	30.033024	0.0	0.0
Profit.To.MaxDraw	0.118483	0.715732	4.026482	NaN	NaN
Max.Equity	112.415249	112.415249	150.960463	2.432805	40.591794
Min.Equity	11.908383	46.89495	120.927439	2.432805	40.591794
End.Equity	11.908383	46.89495	120.927439	2.432805	40.591794

The performance comparison showed that using a threshold of 1.0 standard deviation provided the best trade-off between risk and reward.

However, after conducting back-testing over the training dataset, it was observed that the spread between the assets diverged more often than expected. This behavior resulted in relatively few trading signals, especially for higher thresholds.

After training the model and finding the threshold, it was applied to the test set to evaluate its performance. The results were as follows:

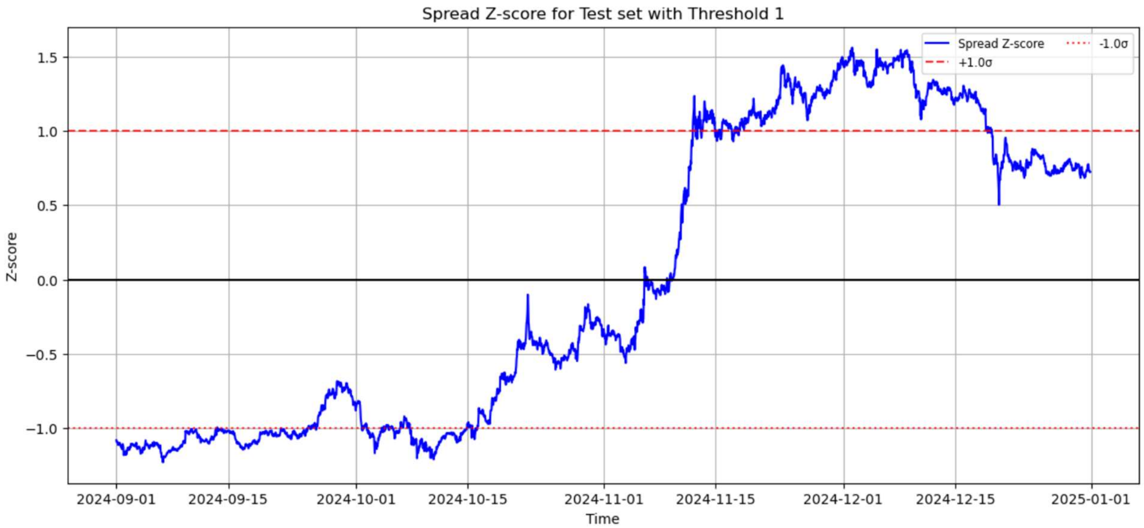
Total Trades: 3

Net Profit & Loss (P&L): 226.56

Average Trade P&L: 75.52

While the model generated a small number of trades, the spread showed signs of divergence during the test period. This behavior led to fewer trading opportunities, but the single trade that was executed resulted in a positive P&L of 0.72.

The divergence in the spread might have limited the model's ability to identify additional entry/exit signals, which contributed to the lower trade count. The graph showing the Z-score and the trading points can be seen below.



Bonus

To address the divergence issue observed in the static model, a rolling-window-based dynamic strategy was implemented, so that the model can adapt to the divergence of the spreads. In the code, I wrote the test period for bonus as "bonus_start" and "bonus_end", these can be changed to the wanted date for the trial.

Instead of relying on fixed thresholds calculated from the training period, this approach continuously recalculates the mean and standard deviation of the spread using a rolling window, dynamically updating the z-score. Entry and exit signals are generated based on these live statistics, making the strategy more responsive to sudden shocks or structural shifts in the market.