## Algorithm User Guide: Pair Trading Strategy for Financial Markets.

### Introduction

This guide aims to provide comprehensive instructions on using the algorithm effectively, including adjusting parameters for different risk profiles and understanding the flexibility in data frequency.

### Requirements

Before getting started, ensure you have Python installed on your system along with the required Python libraries: yfinance, matplotlib, pandas, and numpy. You can install these libraries using pip from “requirements.txt” file:

pip install yfinance matplotlib pandas numpy sklearn statsmodels datetime

### Understanding the Algorithm

#### Overview

Our algorithm implements a statistical arbitrage strategy known as pairs trading. This strategy involves identifying two correlated assets and trading them simultaneously based on deviations from their historical price relationship. The algorithm aims to exploit these pricing inefficiencies in the market for potential profit.

#### Components

1. **The Stock Pair**:

* The pair of stocks chosen, Gicre and Zeelearn, exhibit a high correlation and show cointegration. This means their prices move together over time, and any deviations from this relationship are expected to revert to the mean. This mean-reverting behaviour of their price ratio is the foundation of our trading strategy.

1. **Ratio of the Stocks**:

* Dynamic prices of the two stocks are used to calculate their price ratio over time. This ratio helps identify potential trading opportunities when the ratio deviates from its mean. And using the fact that are stocks are well correlated and cointegrated, we can expect the ratio to come back to their mean value.

1. **Moving Averages and Standard Deviation**:

* Two moving averages (MA1 and MA2) are calculated over different rolling windows (window1 and window2). MA1 is the short-term moving average, and MA2 is the long-term moving average.
* The standard deviation of the ratio is also calculated over the long-term window (window2). These values are used to calculate the z-score, which measures how many standard deviations the ratio is away from the mean.

1. **Z-Score Calculation**:

* The z-score is computed as (MA1 - MA2) / std. It standardizes the difference between the short-term and long-term moving averages, allowing us to gauge how extreme the current ratio is relative to its historical behaviour.

1. **Trading Logic**:

* **Entry Points**:
  + **Long Position**: When the z-score is below -0.6, the algorithm buys Zeelearn and shorts Gicre. This anticipates that the ratio will revert to the mean, allowing the positions to be profitable.
  + **Short Position**: When the z-score is above 0.6, the algorithm shorts Zeelearn and buys Gicre, expecting the ratio to decrease.
* **Exit Points**:
  + If the z-score is between -0.25 and 0.25, the algorithm closes all positions, assuming the ratio has reverted to the mean.

### Interpreting Trading Signals

#### Buy Signal

* **When**: The z-score is below -0.6.
* **Action**: Buy the underperforming stock (Zeelearn) and short the outperforming stock (Gicre).
* **Rationale**: The spread between the stocks has deviated significantly below the mean, and it is expected to revert back, generating a profit.

#### Sell Signal

* **When**: The z-score is above 0.6.
* **Action**: Short the outperforming stock (Zeelearn) and buy the underperforming stock (Gicre).
* **Rationale**: The spread between the stocks has deviated significantly above the mean, and it is expected to revert back, generating a profit.

#### Hold Signal

* **When**: The z-score is between -0.25 and 0.25.
* **Action**: Maintain current positions or close any open positions, assuming the ratio has reverted to the mean.
* **Rationale**: The spread between the stocks is within the normal range, indicating no significant deviation from the mean.

### Using the Algorithm

#### 1. Setting Parameters

The algorithm offers flexibility through various parameters that can be adjusted to customize your trading strategy:

* **Start and End Dates**: Define the start and end dates for the historical data retrieval, as different time frames have different dynamics.
* **Moving Average Windows (window1 and window2)**: Set the window sizes for computing the moving averages of the ratio between the two assets. MA1 is the short-term moving average, and MA2 is the long-term moving average.
* **Data Frequency**: Choose the frequency of data for computing moving averages. Options include daily (‘1d’), weekly (‘1wk’), and 5-day (‘5d’) intervals. It depends on the investor whether they want long term investment or short term trading.
* **Z-score Threshold**: Adjust the threshold for generating buy and sell signals based on z-score deviations. According to user risk profile, z-score threshold can be choosen. Threshold closer to zero indicates more trades, thus a higher risk.

#### 2. Frequency Flexibility

The algorithm provides flexibility in data frequency, allowing you to tailor your trading strategy to different time horizons:

* **Daily Data (‘1d’)**: Suitable for short-term trading strategies with a focus on intraday price movements.
* **Weekly Data (‘1wk’)**: Ideal for medium-term trading strategies that capture broader market trends.
* **5-Day Data (‘5d’)**: Offers a balance between short-term and medium-term strategies, providing a smoother signal compared to daily data.

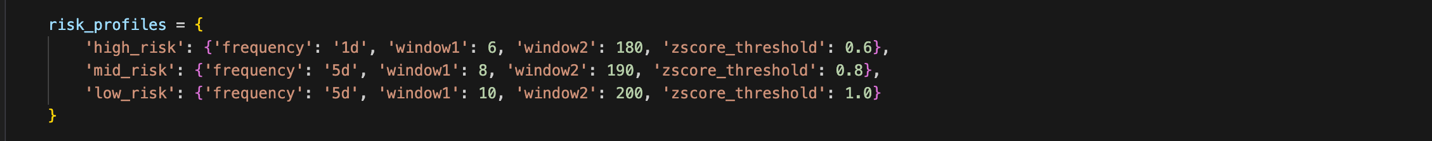
Choose the data frequency that aligns with your trading objectives and risk preferences.

#### 3. Risk Profiles

To accommodate different risk preferences, the algorithm supports multiple risk profiles with predefined parameter sets:

* **High-Risk Profile**: Aggressive trading strategy with shorter moving average windows and lower z-score thresholds, suitable for traders seeking high returns with higher volatility.
* **Mid-Risk Profile**: Balanced trading strategy with moderate moving average windows and z-score thresholds, offering a mix of risk and return potential.
* **Low-Risk Profile**: Conservative trading strategy with longer moving average windows and higher z-score thresholds, emphasizing capital preservation and stability.

Select the risk profile that matches your risk tolerance and investment goals.



### Conclusion

With its flexibility in parameter settings, data frequency options, and support for multiple risk profiles, our algorithm offers a versatile tool for implementing statistical arbitrage trading strategies. By adjusting parameters and selecting the appropriate risk profile, traders can tailor the algorithm to their preferences and objectives, potentially enhancing their trading performance.