Sure, here are some additional details for each step of a pairs trading strategy that uses ML:

1. Data Collection:
   * Collect historical price data for the equities you are interested in trading. Ideally, you should have at least a few years of data to train your ML models.
   * Collect any additional market data that you think may be relevant for selecting pairs, such as fundamentals or news sentiment. This data can be obtained from financial data providers or scraped from public sources.
2. Feature Engineering:
   * Identify technical indicators that may be relevant for selecting pairs, such as moving averages, RSI, or Bollinger Bands.
   * Identify other market data that may be relevant for selecting pairs, such as volatility or trading volume.
   * Explore the data to identify any patterns or anomalies that may be useful for selecting pairs.
3. Pair Selection:
   * Use correlation analysis to identify pairs of stocks that have a high degree of correlation in their price movements. You can use a variety of ML algorithms, such as linear regression or neural networks, to calculate correlations.
   * Use cointegration analysis to identify pairs of stocks that have a long-term relationship. You can use ML algorithms, such as the Johansen test, to identify cointegrated pairs.
   * Use ensemble methods, such as random forests or gradient boosting, to combine the predictions of multiple ML models.
4. Model Training:
   * Split the historical data into training and testing sets.
   * Train your ML models on the training set using a variety of ML algorithms or ensemble methods.
   * Evaluate the performance of your models on the testing set and adjust the models or parameters as needed.
5. Trading Strategy:
   * Develop a trading strategy that uses the selected pairs to generate profits. This could involve placing long and short positions in the pairs based on predicted price movements or other trading rules.
   * Consider implementing risk management strategies, such as stop-loss orders or position sizing, to manage the risk of your trades.
6. Backtesting:
   * Backtest your trading strategy using historical data to evaluate its performance. You can use a variety of metrics, such as Sharpe ratio or maximum drawdown, to evaluate the performance.
   * Adjust the parameters of your ML models or trading rules based on the results of the backtesting.
7. Live Trading:
   * Deploy your trading strategy in live trading and monitor its performance over time.
   * Continuously monitor the performance of your strategy and adjust the models or trading rules as needed based on market conditions and other factors.

These are some additional details for each step of a pairs trading strategy that uses ML. The specific implementation and details of each step will depend on the data available, the ML algorithms used, and other factors.

**EXEMPLES :**

1. Data Collection:
   * Collect historical price data for the equities you are interested in trading from financial data providers such as Yahoo Finance or Quandl.
   * Collect additional data such as fundamental data, news sentiment data, or macroeconomic data from sources like Bloomberg or Reuters.
   * Use APIs or web scraping tools such as BeautifulSoup or Selenium to extract data from websites.
2. Feature Engineering:
   * Technical indicators such as moving averages, RSI, or Bollinger Bands can be calculated using libraries like TA-Lib or using custom code.
   * Other market data such as volatility can be calculated using the historical price data.
   * Unsupervised learning techniques like clustering or PCA can be used to identify patterns in the data.
3. Pair Selection:
   * Use correlation analysis to identify pairs of stocks that have a high degree of correlation in their price movements. Libraries such as NumPy or Pandas can be used for correlation analysis.
   * Cointegration analysis can be done using libraries such as statsmodels or using custom code.
   * Ensemble methods can be implemented using libraries such as scikit-learn.
4. Model Training:
   * Time series forecasting models like ARIMA or LSTM can be used to predict future prices.
   * Supervised learning techniques like linear regression or neural networks can be used for pair selection.
   * Unsupervised learning techniques like clustering or PCA can be used for feature engineering.
5. Trading Strategy:
   * Once pairs have been selected, statistical arbitrage trading strategies can be employed to generate profits. These strategies may involve placing long and short positions in the pairs based on predicted price movements or other trading rules.
   * Machine learning models can be used to identify trading opportunities or to optimize the timing of trades.
   * Risk management strategies such as stop-loss orders or position sizing can be used to manage the risk of trades.
6. Backtesting:
   * Backtesting can be done using libraries like Backtrader or custom code.
   * Different metrics can be used to evaluate the performance of the trading strategy, such as Sharpe ratio, maximum drawdown, or profit and loss.
   * Parameter tuning and optimization can be done to improve the performance of the trading strategy.
7. Live Trading:
   * Trading can be automated using APIs or trading platforms like Interactive Brokers or Robinhood.
   * Regular monitoring and analysis of the trading strategy can be done to identify and address any issues or changes in market conditions.

**Sector/ mean reversion steps :**

1. Select a universe of stocks:

* Choose a specific sector or industry to focus on.
* Research and identify stocks within the sector or industry.
* Filter stocks based on factors such as market capitalization, liquidity, and trading volume.

1. Collect historical price data:

* Identify the time period for which historical price data will be collected.
* Gather price data from reliable sources such as financial data providers or stock exchanges.
* Clean the data to remove any errors or anomalies, and adjust for stock splits or dividends.

1. Select pairs of stocks:

* Use statistical measures such as cointegration to identify pairs of stocks that are highly correlated within the sector or industry but also exhibit some degree of divergence in their prices.
* Apply mean reversion criterion to the pairs to identify opportunities for trades when the price diverges from the mean.
* Identify the trading rules or criteria that will be used to enter and exit trades based on the pair's correlation, price divergence, and mean reversion.

1. Collect additional data:

* Identify relevant sources of data for the sector or industry, such as news articles, economic data, or financial statements.
* Determine which data sources are most relevant for the pairs trading strategy and gather the necessary data.

1. Feature engineering:

* Use feature selection techniques to extract the most relevant features from the additional data sources.
* Preprocess the data to ensure it is in a suitable format for use in machine learning models.

1. Train machine learning models:

* Choose a suitable machine learning algorithm such as SVMs or random forests.
* Split the data into training and testing sets.
* Train the model on the training data and evaluate its performance on the testing data.
* Iterate on the model design and parameters to improve performance.

1. Backtesting:

* Use historical data and the machine learning models to simulate trades over a specified time period.
* Evaluate the performance of the trading strategy using metrics such as Sharpe ratio, maximum drawdown, and total returns.
* Adjust the trading rules and criteria as necessary to improve performance.

1. Implementation:

* Integrate the machine learning models into a trading platform or automated trading system.
* Monitor the performance of the trading strategy in real time and make adjustments as necessary to ensure continued success.
  1. Risk Management: • Determine the maximum risk tolerance and set appropriate stop-loss levels to prevent significant losses. • Monitor and analyze the portfolio risk to ensure that it is within the acceptable limits. • Adjust the position sizes and portfolio allocation as necessary to optimize risk-adjusted returns.
  2. Ongoing Research and Development:

• Continuously monitor the performance of the trading strategy and identify areas for improvement.

• Research new data sources, machine learning techniques, and trading approaches to enhance the strategy. • Regularly review and update the strategy to reflect changes in market conditions, industry trends, and new information.