Applications of Copulas and Machine Learning Algorithms on Pairs Trading

Jifan He[1, *], Jiayu Liu[1], Yihao Fu[1], Yingying Zhu[1], Ziyu Zhou[1], Yinyin Tong[1]

[1]Department of Finance and Risk Management, New York University, New York, United States

*Corresponding author: jh6208@nyu.edu

Abstract. This paper investigated three different pairs trading strategies: the usual baseline (linear) approach, the copulas method, and the machine learning technique. We selected two equity indexes, Russell 2000 (RUT) and S&P400 (SP400), for the pairs trading. It is found that the most significant reasons financial companies employ pairs trading are either its stable nature or profitability. In addition, during a recession session (eg. the Covid-19), the cumulative return of pairs trading can outperform the usual buy-hold strategy and even the market smoothly.

Keywords: Pairs Trading, Bollinger Bands, Copulas, Support Vector Machines, RUT, SP400.

1. Introduction

Pairs trading is a non-directional, relative value investment strategy that seeks to identify two companies or funds with similar characteristics whose equity securities are currently trading at a price relationship that is out of their historical trading range. Unlike the history of some famous economic theories like Game Theory, pairs trading has a relatively short history. Introduced by a group of talented technical analyst researchers, pairs trading reveals its secret to the public and proves its profitability if employed correctly [11]. It is a strategy in which a trader buys one asset while shorting another. The main premise of the trade is that when the two pairs diverge, they will likely converge again, resulting in profits for the trader [9]. This paper analyzed pairs trading strategies using Russell 2000 (RUT) and S&P400 (SP400) indexes in three different methods. We applied the traditional linear approaches, the Bollinger Bands, the copula analysis with rolling windows, and the supervised machine learning tools involving Support Vector Machine (SVM) technique. Our essential comparison measurement is each strategy's cumulative returns and the Sharpe ratio. Finally, we compared them with the buy-and-hold investment and the market returns (SPY), which generate α.

2. Data Preparation

In this section, we will briefly explain which data we used in our analysis, and how we transformed them to help us better train our models (specifically for copula and SVM models).

2.1 Data Selection

2.1.1 Close Prices

Close price series on a daily basis for RUT and SP400 ranging from the end of the year 1987 to the beginning of 2023 from Yahoo Finance were selected to study the pairs trading performance. For pairs trading using close prices, we made the training periods long enough to expose our targeted pairs to a series of global events. While we kept the trading periods long enough to train our dataset for copulas and the machine-learning methods, we used various means for copulas and machine-learning techniques.

2.1.2 Stocks & ETFs

In addition to the close prices for our pairs, we selected 55 additional features, including the market data SPDR S&P 500 ETF Trust (SPY), for the SVM model to predict our pairs' cumulative returns. We selected the daily dataset to make all of our models consistent and used the static Principal

Component Analysis (PCA) to visualize and interpret high dimensional data sets and reduce the overall number of features to combat overfitting.

2.1.3 Relative Strength Index

The Relative Strength Index (RSI) [4] is a momentum oscillator measuring the speed and change of price movements. Inspired by Szklarz *et al.* (2018) [10], we measured the RSIs and applied the traditional way for each of our pairs, RUT, and SP400 in this research. Please note that RSI is just one type of features in our SVM model, along with the features of stocks and ETFs.

2.1.4 Moving Average Convergence Divergence

Designed by Gerald Appel, the Moving Average Convergence Divergence (MACD) is another helpful oscillator that uses three moving averages to form a single momentum oscillator and to trigger the operation signals [1]. In this model, we used 12 and 26 periods for the first two subtracted moving averages, and period 9 for the line for each of our target pairs [10].

2.2 Features Engineering

2.2.1 Daily/Cumulative Returns

To help prepare our data we converted all of our collected daily close prices into either daily or cumulative returns, depending on different methods. We used daily returns to train our data for the copulas method, cumulative returns for SVM. We selected various returns for different models based on their performance comparisons. Transforming raw data into the data we want is crucial in the copulas method since the copulas method requires us to get the daily returns to make the later trading decisions. Features engineering is essential in machine learning algorithms since using the raw dataset usually makes the predictions poorly [3].

3. Pairs Selection

We used the traditional strict cointegration approach, the first Engle-Granger two-step procedure test, and then the much stricter one, the Johansen cointegration test, to further investigate whether there is a significant cointegration relationship between our pairs [2]. We selected 39 equity indexes and examined all possible combinations among them. Our data periods are from the earliest data available for each equity index to January 17th, 2023. And for pairs, the starting period is the earliest data available for the relatively new equity index in pairs to delete the missing values for our pairs. We made our periods long enough to let them undergo historical events and figure out their cointegration effect in the long run.

The resulting outcomes are the pairs passing through the Engle-Granger and the Johansen cointegration tests. We utilized some parts of our copula approach by calculating Kendall's tau for each of the selected pairs and selected the one with the highest tau. Our final winning pairs are RUT and SP400, with a tau coefficient of 0.824658.

4. Model Explanations

4.1 Baseline Strategies

4.1.1 General Baseline Strategy

The baseline strategy indicates the trends of the indexes, as BB Trend has become widely used by traders in many markets, including stocks, futures, and currencies. BB Trend is especially useful in displaying price and volatility [7]. The essence of BB Trend involves three estimation lines: Upper Bollinger Band, Lower Bollinger Band, and Moving Average. The upper band is calculated by adding twice the daily standard deviation to the middle band; the lower band is calculated by taking the middle band minus two times the daily standard deviation. The moving average is the rolling average of the changing price in the market, with a specification of days.

BB Trend takes four additional parameters: typical price, *n* (number of days in smoothing period), *m* (number of standard deviations), and $\sigma(TP,n)$ (standard deviation during the last *n* periods of TP). The typical price is usually calculated by taking the average of our targeted asset's high, low, and close prices daily. *n* and $\sigma(TP,n)$ are all used as parameters to calculate the upper and lower bands. The trading strategy is like this: *investors enter the long position if the price goes below the lower band, enter the short position if the price goes beyond the upper band, and exit the position if the price goes back to the normal level within the upper and lower bands.*

4.1.2 Trading Strategy

Inspired by the traditional BB Trend trading method, we modified our method for BB Trend. The first is our targeted trading asset, a hypothetical one composed of subtracting SP400 from RUT. This asset is the spread between our selected indexes. The second is to combine BB Trend with Kalman Filter regression to estimate our mean and variance of hidden state distributions for times throughout our dataset, which is more accurate than a single measurement. Third, after our trials of hyperparameter tuning, we took the mean of the rolling days of 20 (*n*) of our spread price, the open threshold of 0.7 (*m*), and the close threshold (stop loss) of 0.1 as our moving average and plugged the formulas defined for the upper and lower bands. Fourth, to mitigate the risk of substantial loss, we introduced a new concept: the close threshold, the standard against a calculation similar to the concept of the absolute value of the Sharpe ratio. If this calculation is smaller than our stop loss, we exit our trading position or not enter. Our trading strategy is the following:

If our spread price goes above the upper band, we go short RUT (overbought) and go long SP400 (underbought)

If our spread price goes below the lower band, we go long RUT (underbought) and go short SP400 (overbought)

We exit our position if the absolute value of $\frac{(PriceSpread - MovingAverage)}{\sigma} < 0.1$ (Notice how close this formula compared with the absolute value of the Sharpe ratio)

We chose a mimic of the concept Sharpe ratio because the Sharpe ratio reflects the adjusted return after the volatility consideration of the financial market.

4.2 Copulas

4.2.1 Copula Method in Pairs Trading

The copulas method in Paris trading is a technique to use extensions and generalize the approaches for modeling joint distributions and dependence between financial assets. The Copula method is robust and realistic. The reason behind using the copulas method is to separate marginal distributions from dependence structures in order to capture data information. Copulas method is best used to find the shape and nature of the dependency between the RUT and S&P 400. The variety of copula choices measures upper and lower tail dependencies of different extents in an environment that considers both linear and non-linear relationships [5].

4.2.2 Copula Selection

Before using the copulas method, we calculated returns from the pair: Russell 2000 and S&P400 from September 10th, 1987, to December 31st, 2013, as the formation period and from January 1st, 2014, to January 18th, 2023, as the trading period. Then we calculated Kendall's tau for the pair.

$$\tau = \frac{nc - nd}{\frac{1}{2}n(n-1)} \tag{1}$$

We had a correlation of 0.779892. The value is slightly different than the one in the Pairs Selection due to different timeframes. Both have close tau correlation coefficients, so our selected pairs are robust. Now, we started with the copulas method procedures. First, we fitted the marginal distribution for returns of each stock in the pair. Second, we selected the most suitable copula from Gaussian

Copula, Clayton Copula, Gumbel Copula, Frank Copula, and Joe Copula. By converting the two returns series to two uniform values *u* and *v* using the empirical distribution functions, we computed the Akaike Information Criterion (AIC) and chose minimum AIC.

Frank Copula is the best one. According to the association between Archimedean copulas and the Kendall rank correlation measure, we invoked copula parameter estimation functions to get the value of theta. This two-step approach provides more alternatives in model specification, and an explicit dependence function obtained will give a more delicate description of dependence [6].

4.2.3 Copula Strategy

After the above processes, we started to build the copulas trading strategy. According to Stander Y, Marais D, and Botha I. in their paper "Trading strategies with copulas" [12], if the calculated probability is too low, it suggests that the stock is undervalued, while a too-high calculated probability indicates that the stock is overvalued. The fitted copula is used to derive confidence bands for the conditional marginal distribution function of $C(u|v)$ and $C(v|u)$ to determine mispricing indexes. When market observations fall outside the confidence band, it suggests the presence of pairs trading opportunities. The upper confidence band is set at 95%, and the lower confidence band is set at 5%, as recommended in the paper.

Given the current returns R_x, R_y of RUT and S&P400, we define the "mispricing indexes" are:

$$MI_{X|Y} = P(U \leq u|V \leq v) = \frac{\partial C(u,v)}{\partial v} \tag{2}$$

$$MI_{Y|X} = P(V \leq v|U \leq u) = \frac{\partial C(u,v)}{\partial u} \tag{3}$$

Our strategy constructed short positions in X(RUT) and long positions in Y (S&P400) on the days that $MI_{Y|X} < 0.05$ and $MI_{X|Y} > 0.95$. It constructed the short position in Y (S&P400) and long positions in X(RUT) on the days that $MI_{X|Y} < 0.05$ and $MI_{Y|X} > 0.95$. Otherwise, we exited our trading positions and did nothing. Our rules for closing the positions are:

For the long position, we do not close the position until $MI_{Y|X} < 0.4$ and $MI_{X|Y} > 0.6$

For the short position, we do not close the short position until $MI_{Y|X} < 0.6$ and $MI_{X|Y} > 0.4$

4.3 SVMs

There are some limits to previous methods: marginal distributions of log-return data; rolling windows; hedge ratio; copula parameter theta. Recognizing the potential limits of the baseline and the copula methods, we finally chose the SVM model. The Support Vector Machines model, also known as the SVM model, is a versatile and popular machine-learning technique capable of doing various tasks, including (non)linear classifications and (non)linear regressions, called the SVM Regression. Since SVM is so versatile and famous and can be adapted to nonlinear cases, which is the fact for the relationships between pairs, we used a kernelized Epsilon-Support Vector Regression model, called SVR, from the python package sklearn.svm. Note that SVM needs hyperparameters C (regularization parameter), γ, and ε (the tube's width so that no penalty is given within it in the loss function).

Besides this, as discussed in Section 2.1.2, we used PCA to address the issue of multicollinearity among features. In addition, inspired by Montana and Parella (2009) [8], we used the Weighted Majority Voting (WMV) algorithm to make our final decisions. Then, we tried several combinations of models with different sets of hyperparameters to determine our best prediction case. Finally, we backtested our performance by selecting different sets of parameters (i.e., the PCA component, lookback days, and the correction parameter β), which were demonstrated below.

4.3.1 Pairs Trading with Two Predictions - Usual Case

We first conducted our usual case of pairs trading by predicting the cumulative returns of each of our pairs the next day, RUT, and SP400, assuming that all of the data range we obtained at hand is

from the earliest day *t-lookback* till today *t* by each iteration in the python function. Second, we chose our features carefully, thinking deeply about the possible most important features affecting our pairs separately since we needed to predict each of our trading pairs. We collected all of the available data for each of our selected features to extract the most significant factors accounting for the variance of our chosen pairs' fluctuations. Now we divided our explanations into multiple sectors again to capture the most significant things readers need to know to understand what we did.

(1) Features Selection**:**

Since both RUT and SP400 are equity indexes measuring stocks of various medium companies, it is reasonable to select stocks under each of them. To reduce our computer's working pressure, we just picked the 13 most important stocks of companies under RUT and SP400. In addition, the Exchange Traded Funds (ETFs) tracking their performance are also a good candidate for our features candidates because their close prices' fluctuation gives some information on our targeted pairs.

Table 1. Features Type

RUT Predictors
13 Stocks of companies under RUT
9 ETFs tracking RUT
RSI oscillators for RUT
MACD oscillators for RUT
SPY market data
Lagged RUTs on the previous day and two days
Lagged RSIs for RUT on the previous day and two days
Lagged MACDs for RUT on the previous day and two days
SP400 Predictors
13 Stocks of companies under SP400
6 ETFs tracking SP400
RSI oscillators for SP400
MACD oscillators for SP400
SPY market data
Lagged SP400s on the previous day and two days
Lagged RSIs for SP400 on the previous day and two days
Lagged MACDs for SP400 on the previous day and two days

We chose two technical momentum oscillators, RSI and MACD, to measure the price speed and fluctuation variance among our selected couples. We also included the market data SPY due to the significance of the effect of the macroeconomic conditions on our pairs. The last type of feature we used is the lagged indicators for each couple and oscillators up to two previous days [10], considering the potential autocorrelation relationships for our pairs. Table 1 contains the types of features for our selected pairs.

(2) Pairs Trading Strategy:

We revised what Montana and Parella did by adjusting their model [8]. We chose the static PCA and used another measurement to increase our prediction accuracy by setting two thresholds and determining whether the expected ones fall between these two thresholds. As a result, we should exit the positions. Figure 1 summarizes our trading strategy for the two predictions.

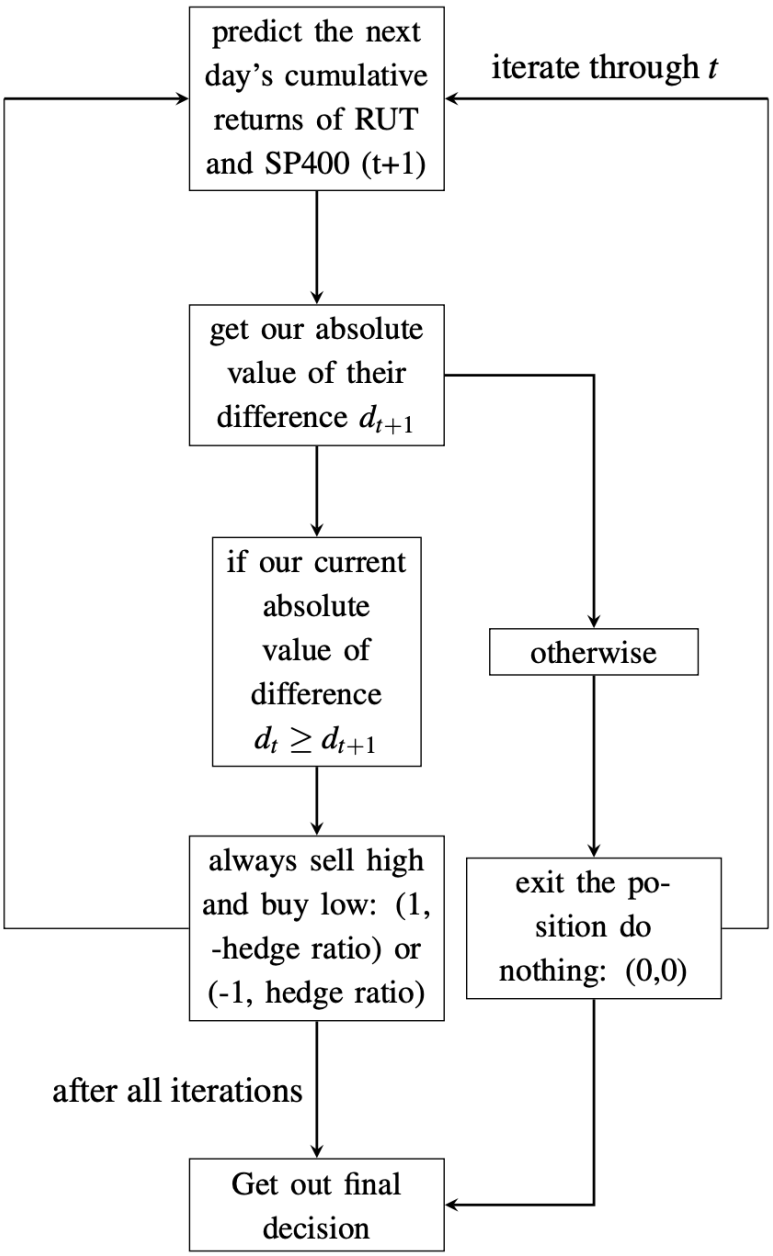

Figure 1. SVM Two Predictions

(Note: *t* starts from t-lookback days, with lookback days a varying parameter in function.)

4.3.2 Pairs Trading with Spread

By evaluating a hypothetical index, we concocted asset's formula is expressed as follows:

$$Asset = |Spread| = Index(i) - Index(j) \quad (4)$$

Typically, we "*short our hypothetical asset if today's return is bigger than the predicted next day's return, and long otherwise. If* they *equal each other, we enter our position and do nothing*".

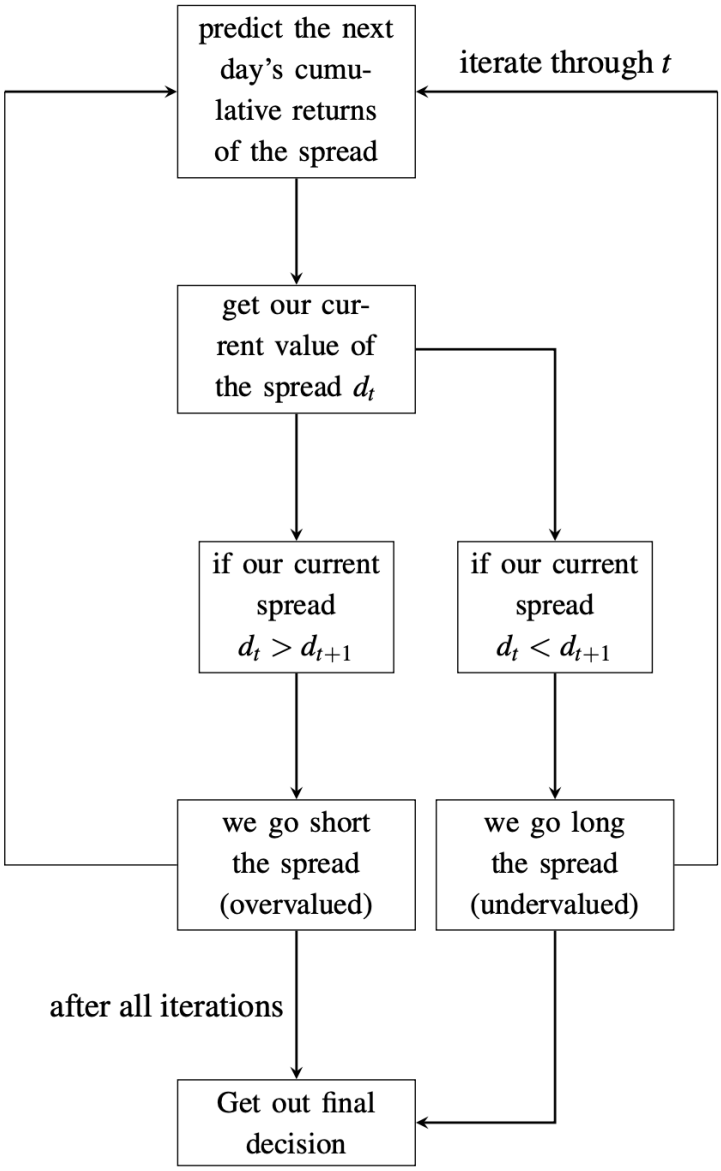

Figure 2. SVM Spread Prediction

(Note: t is the same as before. When $d_t = d_{t+1}$, we exit our trading position if we have one before.)

Figure 2 generalizes our trading logic for the spread prediction in position entry. Notice the difference between the trading logic here and the previous one. The difference is between trading pairs exercising the opposite positions between them and a single asset composed of the combination of pairs; thus, shorting the asset means selling the spread. The spread is the case of selling the index with a higher value (i.e., Index(i) and buying the one with a lower value (i.e., Index(j)):

$$-(Index(i)-Index(j))= Index(j)-Index(i) \quad (5)$$

In real applications we just used the single-traded asset for our pairs trading. So there is a connection between our algorithm for the single-traded asset and the usual pairs-trading strategy. As a result, we should expect more profitable outcomes and a more significant maximum drawdown for our later backtesting procedure, which was confirmed when we plotted our cumulative returns diagram. In addition, the single-asset trading approach shares some similarities with the Bollinger Band model, both of which predict the spread between individuals between pairs. But Bollinger Band still utilizes the traditional pairs trading strategy by explicitly selling high and buying low. In contrast, this trading method simply assumes buying and shorting the hypothetical asset, the spread between

indexes. Finally, except for the differences between these trading strategies in SVM, all of the other procedures, including features selection - *just with all of our selected features predicting our spread* - and our backtesting strategies, are the same for these two methods.

5. Results

In this section, instead of dividing the discussion of our results into multiple subsections, we combined all of our corresponding results from our three models and gave our interpretations and explanations. The critical point is that we compared our results with each of our selected models and gave general insights. First, we compared the method of the baseline Bollinger Band, starting with Figure 3 below, which compares the baseline Bollinger Band, the copula models with rolling windows, and the market.

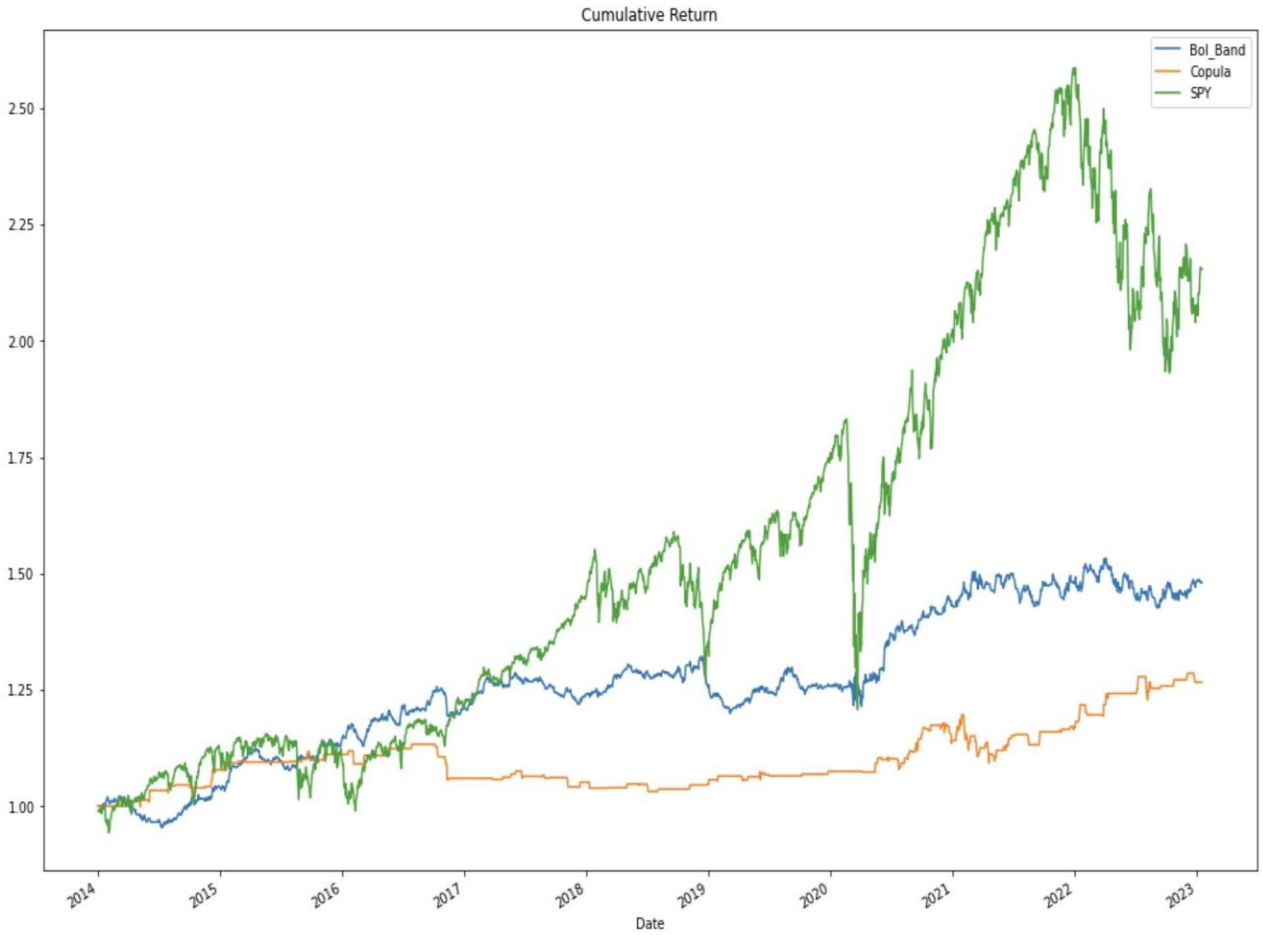

Figure 3. The Comparison with Bol_Band, Copula, and Market

Figure 1 shows that the baseline strategy is significantly better than our copula method regarding the cumulative returns' method. However, neither model can make gamblers happy because they almost do not generate alpha to outperform the market. However, by comparing the three lines in Figure 1, we can spot that compared with the market, both of our algorithms are more stable because they produce steadily increasing and smoothing curves, while for the market (SPY), sometimes days are like riding a roller coaster, with a substantial decrease at the beginning of 2020, the start of the pandemic and several other dropdowns smaller yet bigger than our algorithms. This result is what should be expected for risk-averse or risk-neutral investors because the pairs trading strategy, a particular case of statistical arbitrage, is market-neutral, meaning that the expected returns for investors should be stable yet not exciting. Even though their expected returns are not high, the good

news is that investors can use the money sold for the higher value index to buy the value for the lower one, not needing additional funding costs, which are generally required by financial leverage.

Meanwhile, we also calculated the Sharpe ratio for both the BB Trend and copula methods. The result reverses in that we got a higher Sharpe ratio for the copula method (0.57) than the BB Trend (0.45). In addition to the Sharpe ratios, Table 2 shows the Sharpe ratios, annual returns, and annual volatilities for both the baseline and copula. Both have similar annual returns ranging from 2.5% to 3.0%, which is disappointing for many investors, particularly under the current speculative economic situation: the increase-interest-rate by the Fed; the copula method has a slightly smaller annual volatility (4.8%) than BB Trend (6.8%). Since both of these values are relatively small compared with trading strategies associated with more aggressive investments, we proved that *traditional and some more complicated pairs trading strategies have the stable (excellent) result for investors who care more about the stability of the financial market, rather than the potential high returns generated by the investments*.

Table 2. BB Trend V.S. Copula

Statistics	Baseline	Copula
Sharpe ratios	0.47	0.57
Annual returns	2.9%	2.6%
Annual volatilities	6.8%	4.8%

Finally, our SVM algorithms show that besides measuring the cumulative returns for our algorithms, we tracked the Sharpe ratios, annual returns, total returns, top dropdown, and the duration of our maximum dropdown. All of these measures are the criterion for potential profits and losses. First, we again proved the stability of our SVM pairs trading strategy with two predictions we discussed in the previous section by looking at Figure 4 that shows our SVM algorithm outcompetes the usual buy & hold.

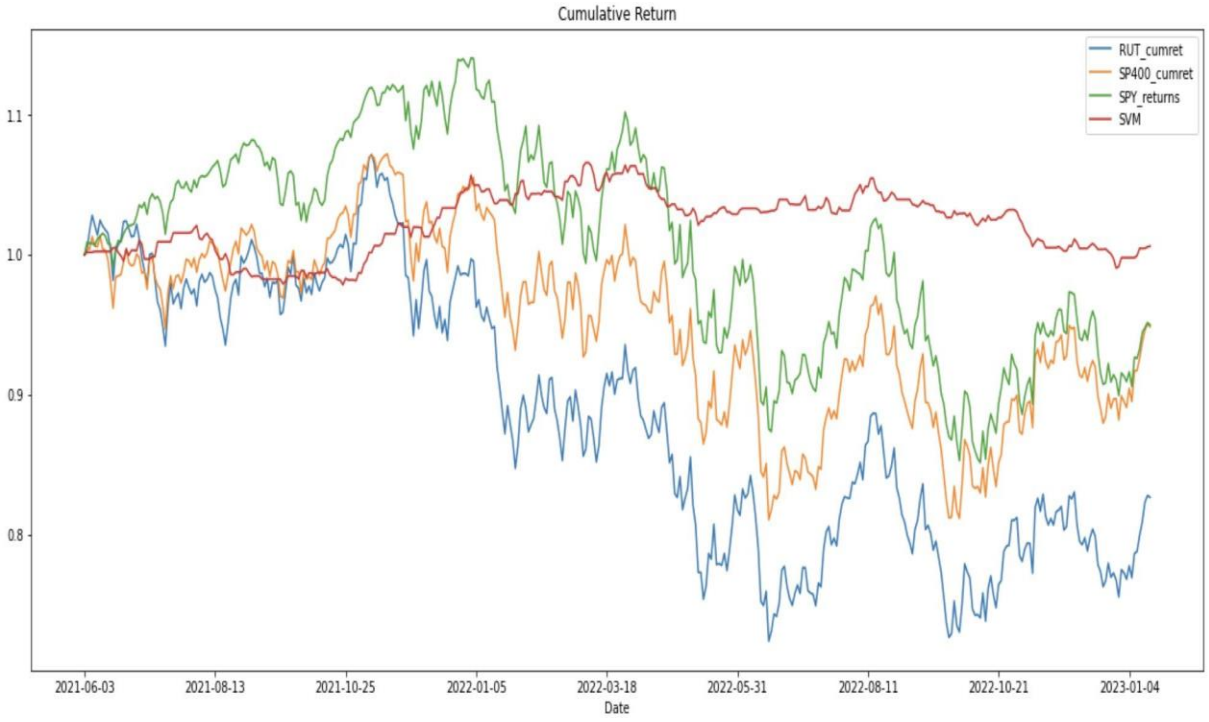

Figure 4. The Comparison with Buy & Hold, Market, and SVM with Two Predictions

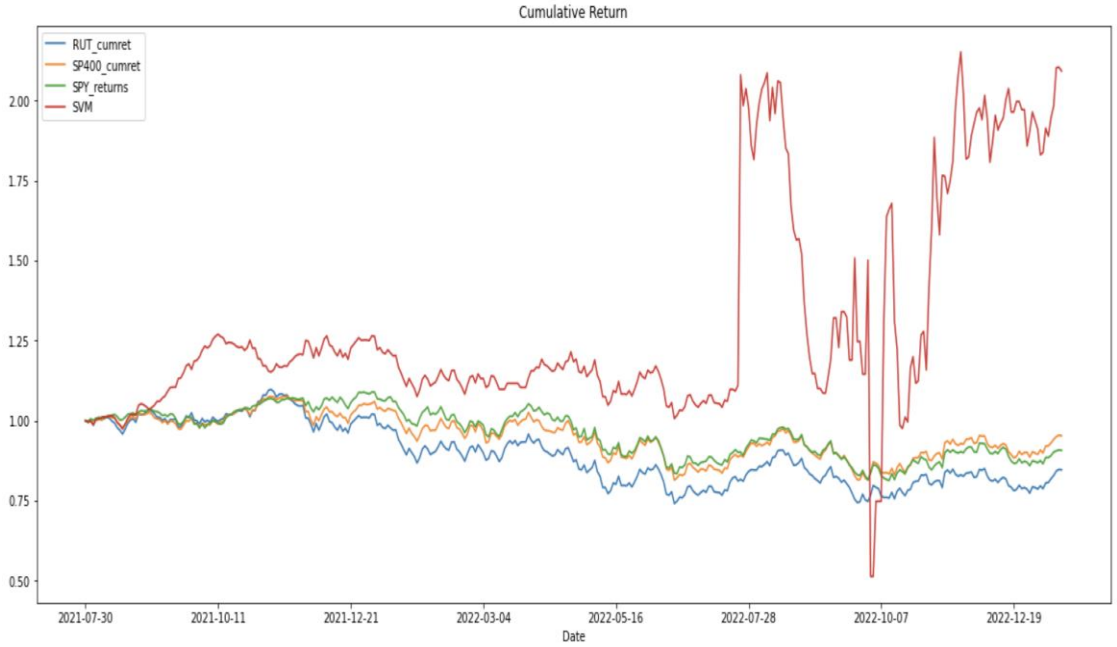

Figure 5. The Comparison with Buy & Hold, Market, and SVM with Spread

We moved on to our second SVM model, which uses the spread as a hypothetical price to design entering and exiting positions. The most important thing to remember in this trading strategy is the spread price, which cannot go below zeros. So, the spread price in this case is the absolute value of the price difference between our pairs, and the reason we warned this would come out soon. Figure 5 shows the fascinating outcomes for aggressive investors or maybe even gamblers, who like riding the roller coaster when deciding their investment strategies: the cumulative returns are steadily increasing (with not a big amount), and then one day it goes extremely high, the other day extremely low, and then high again. Even though this is not what risk-averse people like, the likelihood of earning high profits just by investments gives courage for investors across the globe (especially the people on Wall Street).

With these highly unstable and yet fancy cumulative returns, we spotted a speculative pattern here. The returns with high volatility began with a substantial drop in the spread price, with RUT and SP400 getting close to each other. However, this inverse relationship was only sometimes genuine since we spotted a significantly big dropdown in the cumulative returns with our algorithm, corresponding to a zero spread price. In the financial world, the financial institutions holding the product may go bankrupt.

Table 3 compares all of the other significant criteria for measuring our algorithms. We reported the Sharpe ratio, annual return, total return, maximum drop-down, its duration, and the correlation of our trading strategies with our traded assets and the market.

Table 3. SVM_1 V.S. SVM_2

Statistics	SVM_1	SVM_2	RUT	SP400	SPY
Sharpe ratios	0.1014	1.0219	-0.3397	-0.0304	-0.0517
Annual returns	0.0039	0.6554	-0.1106	-0.0319	-0.0312
Total returns	0.0063	1.0919	-0.1733	-0.0513	-0.0501
Max_Draw	-0.0707	-0.7538	-0.3246	-0.2439	-0.2536
Max_Dur	215.0	196.0	298.0	292.0	260.0
Correlations					
	RUT	SP400	SPY		
SVM_1	0.2476	0.1792	0.1986		
SVM_2	0.0475	0.0419	0.0223		

Table 3 shows the robustness of our trading algorithms compared with the usual buy & hold trading strategies. The Sharpe ratios for both are significantly larger than the ones for both our assets and the market, even though the annual and total returns for the SVM with two predictions are low compared to the other. We found that our first SVM algorithm produces a stable result compared with our second one by comparing the maximum drawdown level (*Max_Draw*) with the first method -0.07 and the second one -0.75. Specifically, the significantly larger Sharpe ratio in our first trading strategy again illustrates our method's high stableness. And the Sharpe ratio of over one (1.02) in our second method confirms that investors can really make much money; the numerator level, which is the reward for our method's returns, is significantly bigger than the volatility of the financial market to produce that 1.02. In contrast, the durations for the maximum drawdown show the opposite story in that the duration (196.0) of our second SVM method is significantly shorter than the ones of the other three assets and the ones in our first SVM method.

In addition to these measurements, Table 3 shows a low correlation is associated with a good one since this means our strategies are almost unaffected by real-world fluctuations. Our first SVM model produces a significantly higher correlation than our second one, which is reasonable because the first one's logic requires our predictions of two prices. However, measuring the stability shows that our second method is more stable regarding the sensitivities to the outside world instead of the volatility associated with itself.

6. Conclusion

First, we proved the robustness of the pairs trading as a market-neutral strategy with steady, smoothing, and increasing returns by the baseline and copula models. Second, we tried the more advanced SVM model and simultaneously proved the potential profitability (the second SVM model with the spread prediction) and the stableness again (the first SVM model with two predictions) for the pairs trading strategy.

7. Future Improvements

We realized that there are several limits to our research. The most important one is our strict criteria for selecting our candidates. We should try different methods like the now famous machine learning clustering algorithms and try out a larger dataset to find pairs generating potentially higher returns [9]. Future research should also focus more on some other pairs trading algorithms such as the reinforcement learning method.

References

[1] Appel, Gerard (2008). Understanding Macd (Moving Average Convergence Divergence).

[2] Bilgili, Faik (1998). Stationarity and cointegration tests: Comparison of Engle - Granger and Johansen methodologies.

[3] Géron, Aurélien (2019). Hands-on machine learning with Scikit-Learn, Keras and TensorFlow: concepts, tools, and techniques to build intelligent systems. 2nd ed.

[4] J. W. Wilder (1978). New Concepts in Technical Trading Systems.

[5] Liew, R., & Wu, Y (2013). Pairs trading: A copula approach. J Deriv Hedge Funds 19, 12–30 (2013).

[6] Ling, Hu (2006). Dependence patterns across financial markets: a mixed copula approach.

[7] Mitchell, Cory (2022). Using Bollinger Bands to Gauge Trends.

[8] Montana, G., & Parella, F. (2009). Data Mining for Algorithmic Asset Management.

[9] Radovanovic, Igor (2022). Cluster Analysis – Machine Learning for Pairs Trading.

[10] Szklarz, J., Rosillo, R., Alvarez, N., Fernández, I., Garcia, N. (2018). Application of Support Vector Machine on Algorithmic Trading.V

[11] idyamurthy, Ganapathy (2009). Pairs Trading: Quantitative Methods and Analysis. 1st ed.