Improvement of Forex Pairs trading strategy with Machine learning algorithms

Mr. Ronnachai Jirapongpan

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering in Industrial Engineering
Department of Industrial Engineering
FACULTY OF ENGINEERING
Chulalongkorn University
Academic Year 2019

ปรับปรุงกลยุทธ์การซื้อขายเงินตราต่างประเทศแบบคู่โดยใช้การเรียนรู้ของเครื่อง

นายรณชัย จิระพงษ์พันธ์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมอุตสาหการ ภาควิชาวิศวกรรมอุตสาหการ
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2562

Thesis Title	Improvement of Forex Pairs trading strategy with Machine learning algorithms
By	Mr. Ronnachai Jirapongpan
Field of Study	Industrial Engineering
Thesis Advisor	Associate Professor NARAGAIN PHUMCHUSRI, Ph.D.

Accepted by the FACULTY OF ENGINEERING, Chulalongkorn University in Partial Fulfillment of the Requirement for the Master of Engineering

.... Dean of the FACULTY OF ENGINEERING
(Professor SUPOT TEACHAVORASINSKUN, D.Eng.)

THESIS COMMITTEE

.... Chairman
(Associate Professor WIPAWEE THARMMAPHORNPHILAS, Ph.D.)

.... Thesis Advisor
(Associate Professor NARAGAIN PHUMCHUSRI, Ph.D.)

.... Examiner
(NANTACHAI KANTANANTHA, Ph.D.)

.... External Examiner
(Assistant Professor Manop Reodecha, Ph.D.)

รณชัย จิระพงษ์พันธ์ : ปรับปรุงกลยุทธ์การซื้อขายเงินตราต่างประเทศแบบคู่โดยใช้การเรียนรู้ของเครื่อง. (Improvement of Forex Pairs trading strategy with Machine learning algorithms) อ.ที่ปรึกษาหลัก : รศ. ดร.นระเกณฑ์ พุ่มชูศรี

งานวิจัยนี้ศึกษาการนำการเรียนรู้ของเครื่องมาใช้ในการเพิ่มประสิทธิภาพของเทคนิคการเทรดแบบคู่ การเทรดแบบคู่เป็นหนึ่งในระบบการซื้อขยอัติโนมัติที่มีชื่อเสียงมาก ซึ่งถูกคิดค้นขึ้นมาช่วงศตวรรษที่80โดยทีมนักวิทยาศาสตร์และนักคณิตศาสตร์ แนวคิดของการเทรดแบบคู่คือการที่ระบบการซื้อขายใช้ประโยชน์จากการที่ราคาของหุ้นสองตัวที่เคลื่อนที่ไปพร้อมๆกันเกิดเสียสมดุลชั่วขณะ เมื่อระบบการซื้อขายนี้สามารถรับรู้การเสียสาดุลของราคาหุ้นทั้งสองด้วยด้วยการคำนวณจากสมการหรือการใช้ตัวบ่งชี้ ผู้ซื่อขายจะเปิดสัญญาขายล่วงหน้ากับหุ้นที่มีราคาสูงว่าจุดสมดุลของราคา และเปิดสัญญาซื้อล่วงหน้าเช่นเดียวกันกับหุ่นที่มีราคาต่ำกว่าจุดสมดุล เมื่อราคาของหุ้นทั้งสองตัวกลับมาสู่จุดสมดุล นักลงทุนจะทำการปิดสัญญาทั้งสองและรับรู้กำไรขาดทุน อย่างไรก็ตามความสามารถในการทำกำไรของการเทรดแบบคู่นั้นขึ้นอยู่กับหลายปัจจัย งาวิจัยนี้จึงได้นำขั้นตอนวิธีการเรียนร้ของเครื่องจักร เช่น Artificial Neural Network, Logistic regression และ XGBoost มาใช้ในการพยากรณ์ความสามารถในการทำกำไรจากตัวชี้วัดแบบตามที่เก็บรวมรวมมาจากการซื้อขายในในอดีต ขั้นตอนการศึกษาของการงานวิจัยนี้คือใช้วิธีการต่างๆเพื่อปรับแต่งขั้นตอนการเรียนรู้ของเครื่องจักรให้มีประสิทธิภาพสูงสุดเช่น feature selection, standardization และ GridSearchCV จากผลลัพธ์ของการศึกษาพบว่าประสิทธิภาพของการเรียนรู้ของเครื่องจักรเป็นที่น่าพอใจ เกณฑ์คะแนนส่วนใหญ่มีค่ามากว่า 60% แสดงว่าแบบจำลองมีความสามารถพยากรณ์ความสามารถในการทำกำไรของการเทรดแบบคู่ได้ เส้นกำไรขาดทุนสะสมหรือเส้นดุลจากการใช้งานการเรียนรู้ของเครื่องร่วมด้วยนั้นมีค่ามากกว่าเส้นดุลจากการเทรดแบบคู่แบบปกติอย่างมีนัยสำคัญ

สาขาวิชา	วิศวกรรมอุตสาหการ	ลายมือชื่อนิสิต ....
ปีการศึกษา	2562	ลายมือชื่อ อ.ที่ปรึกษาหลัก ....

6170250821 : MAJOR INDUSTRIAL ENGINEERING
KEYWOR
D:

Ronnachai Jirapongpan : Improvement of Forex Pairs trading strategy with Machine learning algorithms. Advisor: Assoc. Prof. NARAGAIN PHUMCHUSRI, Ph.D.

This thesis aims to study machine learning to improve the trading performance of pairs trading strategy. Pairs trading strategy is one of the most well-known algorithm trading developed in the 1980s by a group of scientists and mathematicians. The concept of the pairs trading strategy is to exploit the mispricing of two equities which their prices tend to move in the same manner. When the algorithm captures the mispricing behavior by equations or indicators, traders open a short position on the equity which its price is relatively higher than the equilibrium and open long position of the other equity. If the prices reach an equilibrium point, the trade positions are closed with realized profit/loss. However, there are many factors that influence the profitability of the algorithm. This thesis applies machine algorithms that consist of Artificial Neural Network, Logistic Regression, and XGBoost to predict the profitability from lagging indicators from the trading records. The methodology of the thesis aims to tune and maximize the performance of machine learning algorithms such as feature selection, standardization, GridSearchCV, etc. The result of using trained machine learning is quite satisfactory. The scores from implementing the machine learning on out-of-sample are mostly higher than 60%, meaning that the models are capable of predicting the profitability of signals from lagging indicators. The cumulative profit or the balance curve from using machine learning is significantly higher than the balance curve from normal pairs trading strategy.

Field of Study:	Industrial Engineering	Student's Signature ....
Academic Year:	2019	Advisor's Signature ....

ACKNOWLEDGEMENTS

I would like to express my gratitude and appreciation to many people who provided support and encouraged me throughout my master’s degree.

First, I would like to sincerely thank my thesis advisor, Assoc. Prof. Dr. Naragain Phumchusri, who has been very supportive during every step of the thesis from the beginning to the end, this thesis would not be completed without her valuable guidance and kindness supports.

Also, I would like to show my respect and appreciation to Assoc. Prof. Dr. Wipawee Tharmmaphornphilas, and Mr. Phongsatorn Amornvetchayakul for their time and helpful advice throughout the research.

Furthermore, I would like to acknowledge Aj. Dr. Nantachai Kantanantha, Ph.D. for all suggestions, comments, and knowledge for my thesis. In addition, I would like to thank all staff in the Industrial Engineering department as well as in Sasin Graduate Institute of Business Administration for their help during my study.

Lastly, I wish to express my deepest gratitude to all my friends and family who are always beside me. I appreciate your love and support. Thank you.

Ronnachai Jirapongpan

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

1.1 BACKGROUND

Quantitative finance, using the mathematical models and huge data to examines, analyze, and justify the financial markets and securities, attracts many investors and traders to make profits. The one who applies the quantitative finance to model the in the trading strategy is called “quants”.

A lot of research has derived the trading rules to model the price movement and market behaviors from historical data and information in the past. Such behaviors are studied and parameterized to forecast and make a profit in the future with acceptable risk. Then, quants use these models to trade and make a profit on various assets.

Pairs trading strategy is one of the most popular and successful quantitative methodologies developed in the 1980s by the team of scientists, mathematicians, and computer scientists. The group is formed by the Wall Street quant Nunzio Tartaglia. The method was developed from the statistical model and computer algorithm without the decision from the human. According to [3] The pairs trading strategy worked very well in the first move, but the performance was not consistent in the long-term.

The concept of the pairs trading strategy is very straightforward. It exploits the benefit of the market inefficient. The first step is to search for the pairs of assets which share the same fundamental or the price is likely to move together such as stocks in the same industry. Then, when the prices diverge and exceed the historical threshold, so-called spread, the prices tend to converge to the symmetrical price or the market neutral position. It is anticipated that the asset prices which are below the symmetrical price will rise and the other will fall. In practice, traders open short positions the high asset price and long position of the lower asset price. Soon, a pair of assets will move to the equilibrium price and traders will close the position and take the profit.

Many papers have tried to find the mathematical models in order the identify the method to select the pair of assets [3-9], the models are to measure abnormality of the assets and the method to identify the spread threshold to trigger the trading signal.

1.2 PAIRS TRADING FRAMEWORK

Pairs trading is a rule-based investment strategy exploiting the mispricing behavior of securities. Statistical arbitrage is one type of mispricing. When the two time series data share the same characteristics such as the same operating businesses, listed in the same industry or expose to the similar risk factors, they tend to move together, following the law of one price. When the price of two securities moves diverse from each other, they generate the price difference called "spread". The spread of the pair is considered as a stationary and mean-reverting process. The spread of the price is assumed to be white noise, which means that if the spread exceeds the statistical threshold, typically 2-standard deviation, the investors can open the position by opening the short position the winner stock and opening the long position the loser stock. When the spread converges to the mean, the position is closed and realized the profit (loss) of the pair. The questions are raised such as the performance of the strategy, how to form proper pairs, Is the spread mean-reverting, and so on. There is a lot of research trying to answer, develop, and improve the method of pairs trading strategy. The example of pair trading strategy focusing on a practical approach is shown in the next topic.

1.1.1 Co-integration pairs trading in practice

The method is referred to [3]. This paper studied the practical approach of a pairs trading strategy. The performance of the strategy is collected in the systematical method. They identify the potential pairs of stock, test the profitability of the pairs, and evaluate the excess return.

1.1.2 Methodology

The approach is to separate the data into 2 sections. The first section is a 12-month period or formation period. The second section is the next 6-month period or trading period used to trade and collect data of excess return from pairs trading strategy. The formation period and trading period are chosen arbitrarily and imposed this method in the entire data.

1.1.3 Pairs Formation

The stock in CRSP daily files are screened and qualified by eliminating the stocks which have one or more days without trading for relative liquidity purpose. Next, the qualified stocks in CRSP daily files are used to construct the log-return cumulative profit. Then, the bootstrap approach is performed to generate pairs of stocks and calculate the sum of squared deviations between the two normalized price series in close daily price framework.

The bootstrap method eliminates the bias of selecting the pairs of stocks. The core idea of the pairs trading is to find pairs of securities that tend to move together. The word "move together" could be interpreted in many solutions, for example, select the stock in the same industry. Thus, they systematized the selection criteria by ranking the sum of squared deviations. Another benefit is the stocks can be selected in cross-industry

1.1.4 Trading Period

The authors tested the pairs trading strategy with daily historical prices from 1962-2002 in liquid stocks. They collect the data in CRSP daily files. Then, they create the cumulative total return from the in-sample stock price. Next, they form the pairs of stock which move together and have co-integration criteria by minimizing the sum of squared deviations (SSD) between the two normalized price series. Once they calculated the SSD, they ranked from the lowest SSD and grouped the pairs of stock to the top 5, top 20, and 101-200 lowest SSD pairs. Then, they start to simulate the trade from the last day of pairs selecting period with the specified rules of pairs trading strategy.

Figure 1-1 illustrates the pairs trading strategy using two stocks, Kennecott and Uniroyal are traded in a trading six-month period (trading period) starting in August 1962. The two series lines show the normalized of two stocks with dividends re-invested. The bottom line shows the status of orders of the pair daily timeframe. The normalized prices clearly show the co-movement of the stocks, the spread between two stock is widened and narrowed periodically. The spread of the stocks is widened significantly on the seventh day, passing the criteria to open the position (2-standard deviation criteria). Until the 36th day, the normalized prices are converged, creating zero spread. Then, the position of each stock is close.

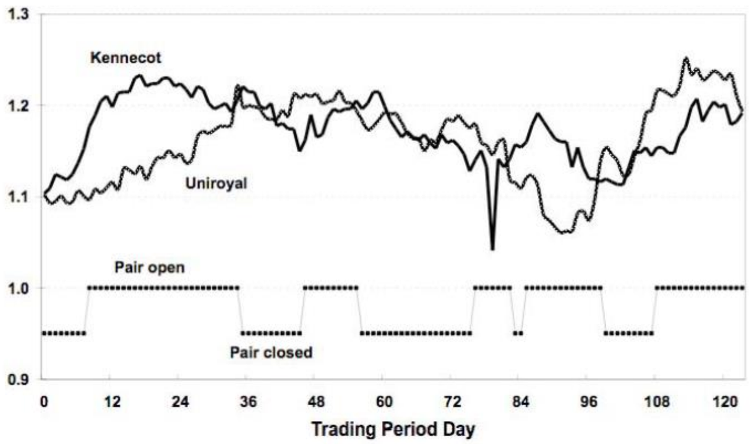

Figure 1-1 The pairs trading strategy of Kennecott and Uniroyal in six-month period [3]

1.2 STRESS INDICATOR METHODOLOGY

In [10] purposed easier way to apply pairs trading strategy. In the reading, the authors present the different methodology from [3], called relative pricing difference. The authors introduce the stress indicator such as Relative Strength Index (RSI), Stochastic Oscillator and Stress Indicator to measure the momentum of stock price. Next, the historical standard deviation of the difference of indicator between the pair is measures and set as the rule-based parameter to open the order.

The authors simply selected the stock by the fundamental framework, the correlation could be implemented but it is not necessary from this framework. The trader can visually check by plotting the graph against each other. The graphs should move in similar manner such as reacting the news in same pattern. Another fundamental criterion is both stocks are in same industry and compete each other such as Hewlett-Packard and Dell in technology industry, Eastern Airline and American airline in airline industry.

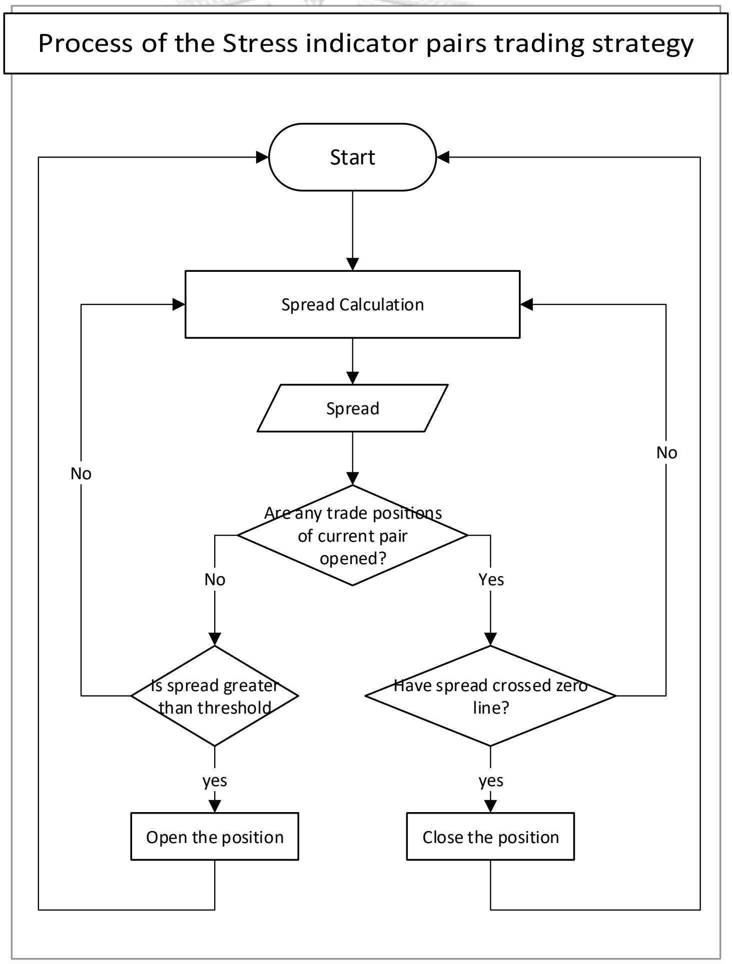

Figure 1-2 Process of stress indicator pairs trading strategy

Figure 1-2 shows the process of the stress indicator pairs trading strategy. When the process is run on the pair of assets, the spread of the stress indicator is calculated. The algorithm checks the trade positions at the current time whether any positions is opened. If the trade positions have not been opened yet, the algorithm opens the positions if the spread is greater than the threshold. If the current trade positions are still opened, the algorithm is going to close the positions if the spread value crosses the zero. When the trade positions are closed, the profit/loss is realized.

The authors selected LCC, CAL, AMR, and LUV in airline industry to illustrate the trading methodology. The authors anticipated that these company operate the business in the same manner, reacting to the economics similarly. Figure 1-3(top) shows the price series of the LCC and CAL against each other.

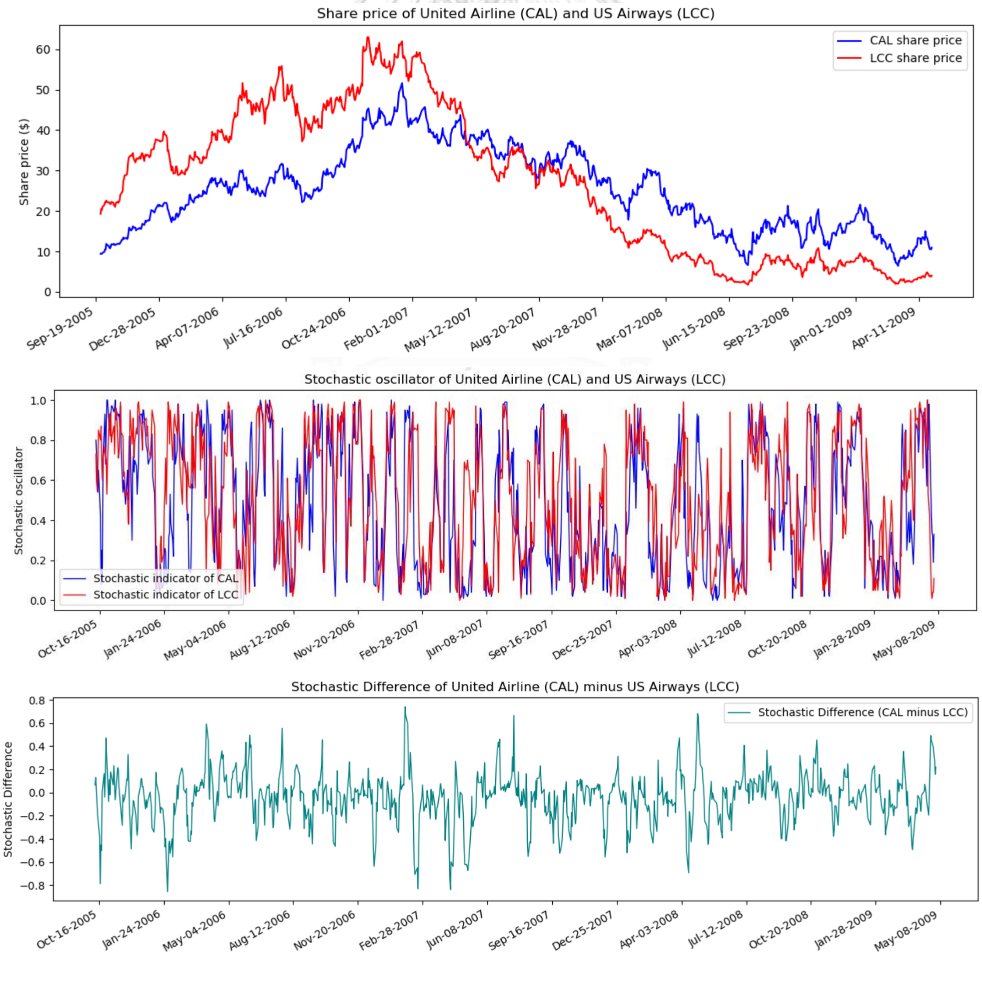

Figure 1-3 The share price (top), Stochastic oscillator (middle) and Stochastic difference (bottom) of vs CAL [1, 2]

For relative difference indicator, the authors chose Stochastic indicator period 14 to measure the momentum of the stocks. Next Stochastic indicator of LCC, Stochastic indicator of CAL and the difference of the indicator were plotted in Figure 1-3(middle) and Figure 1-3(bottom) respectively.

To identify the threshold, the strategy triggers the signal when the stochastic difference moves below -40 for short signal and above 40 or long signal

From Table 1-1, the positions are traded 13 times during the period and PL/share is greater than 0.354. Thus, the pair LCC-CAL is anticipated as profitable. Next, the size of the positions is compensated by the volatility.

Table 1-1 Summary of pairs trading LCC - CAL [10]

Pair	**# trades**	**Correlation**	**PL/Share**
LCC-CAL	13	0.765	0.354

The volatility adjustment plays the important role when the volatility of each leg is different. For example, if the price of AMR fluctuates more than CAL, the loss from the AMR could overwhelm to profit from CAL. To solve the problem, the position size of lower volatility leg is traded greater than the higher volatility leg to compensate the range of price movement. There are plenty of indicator to measure the volatility such as standard deviation of daily return and Average True Range (ATR). Next this value is used as volatility adjustment. For example, at the time the position is being open, the volatility of AMR and CAL is 0.903 and 0.748, respectively. Therefore, the position size of CAL is compensated by multiplying 1.21 (.903/.748). Let assume that the position size of AMR is 100 shares, meaning the position size of CAL is traded at 121 shares.

Next, the indicator difference pairs trading strategy and volatility adjustment are applied in other assets which are explained in the next section.

1.3 PROBLEM STATEMENT

Practical result in currency exchange

In this section, stress indicator methodology is extended further. Regarding to the pair selection method, the authors suggested that the profitability of stress indicator pairs trading strategy mostly relates to the correlation of the pairs. Our proposition is, in quantitative perspective, if the correlation is the main factor of profitability of pairs trading strategy, this strategy should be extended further to the other assets as long as traders can seek for the pairs of asset which satisfy the correlation criteria, These pairs should generate cashflow as same as the pairs of stock in airline industry exemplified in previous topic regardless of the class of asset, timeframe and fundamental of asset. Therefore, the foreign exchange (forex) is tested to apply the stress indicator trading strategy

Foreign exchange, aka FOREX or currency trading, is the world largest currency trading market and most liquid in the world. The application Metatrader 5 in Windows platform shown in Figure 1-4 is used to perform strategy testing. Next, the testing procedure is identified to apply the strategy systematically.

Figure 1-4 The exchange rate of Euro against US Dollar in Hour timeframe [11]

Period identification

The exchange rates based on hourly timeframe are tested by the strategy. The exchange rates in the pairs trading are assigned as Leg1 and Leg2. The exchange rate used to perform the calculation is based on open price. The data are separated into 2 periods. First is data collection period, performed in January 2008 to December 2009, and second is testing period, performed in January 2010 to December 2014. The data collection period is to screen the pairs by calculating the correlation and collecting historical indicator spread for generating 2-standard deviation rule.

Pair selection

The pair selection criteria are performed by calculating and collecting the correlation of the pair of time series exchange rate in data collection period. The correlation of pairs is calculated in hourly basis every week from first week of 2008 to last week of 2010. In this section, the pair of exchange rate used to perform the correlation testing is AUDUSD as Leg1 and EURUSD as Leg2

Average of the correlation is 0.685. Regarding to Figure 1-5, the skewness of the histogram is clearly negative (clustered on right side). It can imply that the pair of AUDUSD/EURUSD is highly correlated.

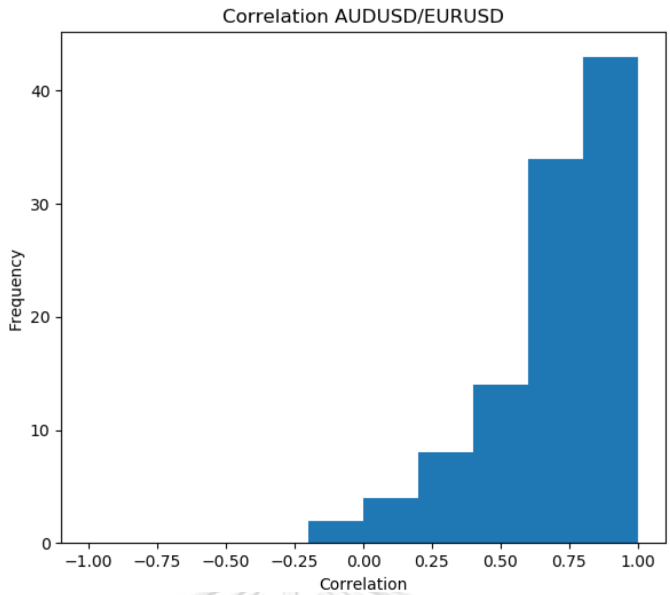

Figure 1-5 Histogram plot of correlation of AUDUSD/EURUSD

For comparison, the pair of EURGBP/AUDCAD is exemplified in Figure 1-6. The figure clearly shows that the weekly correlation of EURGBP/AUDCAD pair is scattered and average is only 0.081. Thus, the EURGBP/AUDCAD is not correlated.

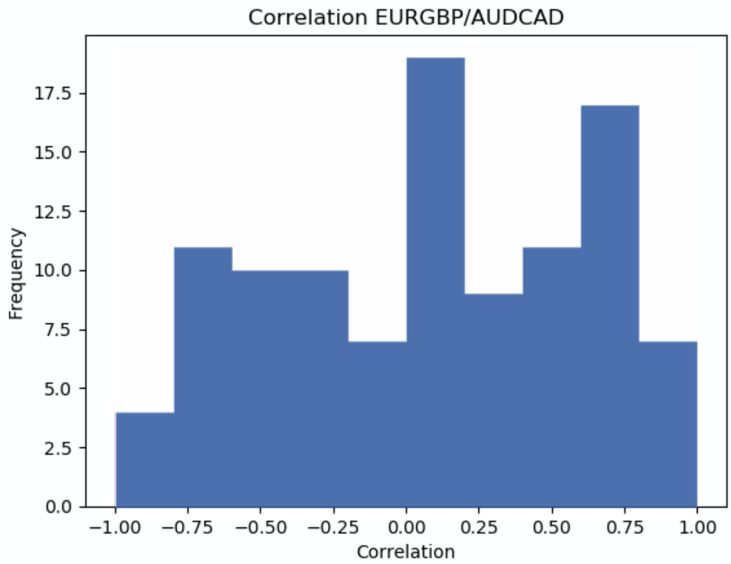

Figure 1-6 Histogram plot of correlation of EURGBP/AUDCAD

Stress indicator

For this simulation, Relative Strength Index (RSI) period 14 is used to measure the momentum and calculate historical spread. The data are collected, calculated the mean, and identified 2-standard deviation value. In this research, the spread is defined RSI_leg1 – RSI_leg2

Figure 1-7 shows the exchange rate of Leg1 and Leg2. The lower windows show the RSI value of current hour. Leg1(AUDUSD) and Leg2(EURUSD) indicate that RSIs are 47.67 and 43.50, respectively. Therefore, the indicator spread is 4.17

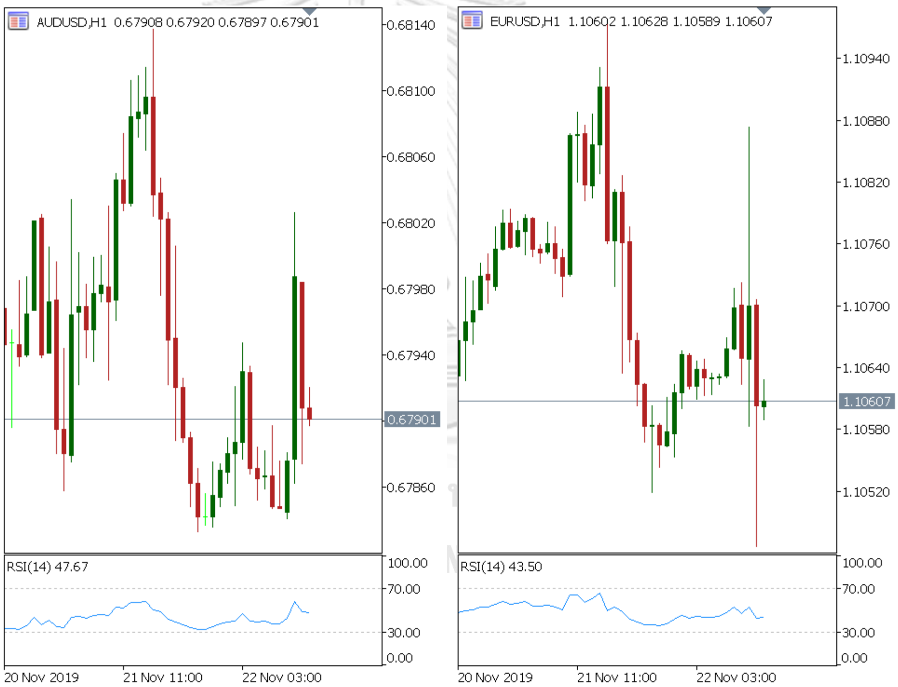

Figure 1-7 Candlestick and RSI plot of AUDUSD and EURUSD in H1 timeframe [12]

Data collection

The indicator spread from January 2008 through December 2009 is recorded and plotted in Figure 1-8. The standard deviation of the spread is 10.085. Our implement is to open the position of Leg1 and Leg2 if spread greater(lower) than 20.17(-20.17), short(long) position on Leg1 and long(short) on Leg2 are opened. Next is the mean of spread, from the data, mean of the spread is 0.5. To simplify the rule, zero is simply set as the threshold to close the positions.

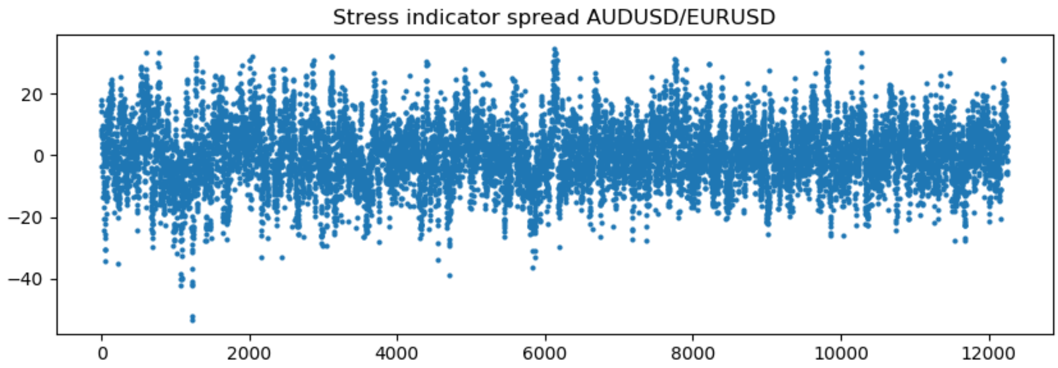

Figure 1-8 Historical of RSI difference of AUDUSD and EURUSD

Volatility

The volatility is measured by calculating the weekly standard deviation of the return of each hour from January 2008 through December 2009 as same as stress indicator. Then, the factor to apply the volatility scaling is needed to be identified. The volatility of each leg is scale in the different approach. The volatility of each leg is plotted by Kernel Density Estimate method (KDE).

Figure 1-9 illustrates the KDE of Volatility leg1 and leg2. It clearly shows that the volatility of each leg is clustered at the single point. Therefore, the average of volatility is used to be volatility scaling factor. To elaborate, the mean of volatility of leg1 and leg2 are 0.00241 and 0.00156, respectively. Then, the position size of leg2 is adjusted by multiplying the volatility scaling factor which equals 1.54

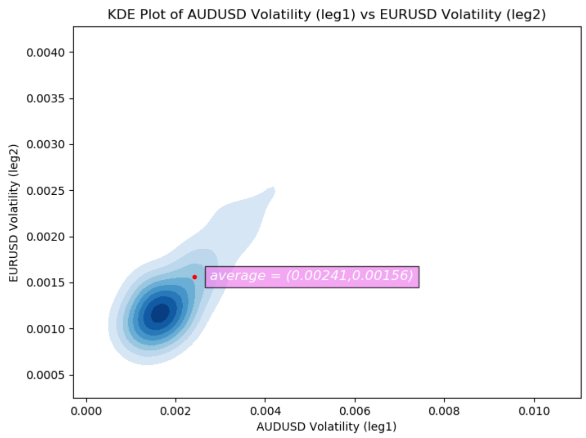

Figure 1-9 Volatility plot of AUDUSD (leg1) vs EURUSD (leg2)

Perform the test

In this section, the strategy is simulated from January 2010 to December 2014. The rule-based parameter is set from the calculation as shown previously. The position is opened following the rules

- Start margin at $10,000
- Position size of leg1 and leg2 are 0.1 lots and 0.15 lots
- Short the leg1(AUDUSD) and long the leg2(EURUSD) if the spread is greater than 20.17
- Long the leg1(AUDUSD) and short the leg2(EURUSD) if the spread is lower than -20.17
- Close the positions (both leg1 and leg2) if the spread cross zero.

Result

The total profit is not quite good. Figure 1-10 shows the total balance after realizing the profit. According to the back-test result, the total profit is -$8092 from January 2010 to December 2014, total trade is 582 orders. The key measure of the performance is Sharpe ratio, the average return or expected return is -$13.93 The result implies that the strategy is not profitable.

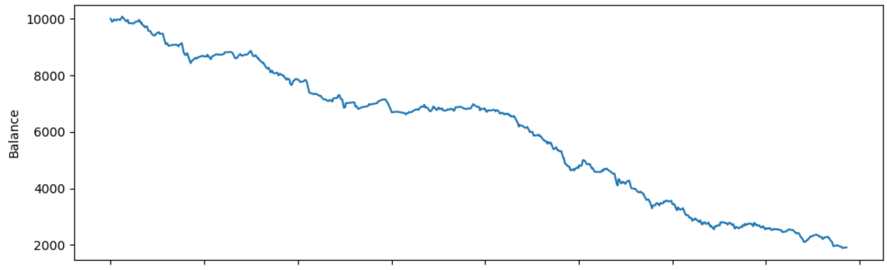

Figure 1-10 Cumulative balance of pairs trading strategy of EURUSD vs AUDUSD from January 2010 to December 2014

Regarding to the result, the strategy cannot generate the cashflow as same as the pairs of airline stock stated in previous topic. [10] introduces the volatility threshold to filter the signal. When the spread is eligible to open the position, the volatility of each pair is calculated. If the realized volatility exceeds the threshold, the position is opened and vice versa. According [13], the profitability of pairs trading. The authors refer the methodology of [3] and investigate that how do the volatility and correlation influence the return per trade. They find that High pair volatility and low correlation are advantageous for the return per trade, but disadvantageous for the trading frequency. They also suggest that the traders should optimize the volatility and correlation level the maximize the return. Thus, the volatility and correlation might play an important role in pairs trading strategy.

In conclusion, the problems are that

1. Do the volatility and correlation influence the trading performance? Regarding to the explanation above, the correlation and volatility would be the potential influential factors of pairs trading strategy. To improve the pairs trading efficiency, the relation of volatility, correlation, and other factors to filter the signals is needed to study.
2. If the volatility, correlation, or the other factors significantly influence the profitability, what is the methodology to calculate the threshold? For example, the volatility is dynamically changed over time. The question is that if the static threshold is set, traders might lose the

opportunity to trade causing low frequency of trade. The study shows that the correlation and volatility influence the profit, but it could not imply that low volatility cannot generate cashflow.

3. If the pairs trading strategy is improved by the signal filter, the expected return should increase statistically. Therefore, the statistical t-test to the hypothesis is implemented to test the improvement.

1.4 OBJECTIVE

The objective of the study is to introduce the novel approach to qualify the signal based on stress indicator pairs trading strategy.

1. Search for the potential of the pairs of exchange rates.
2. Construct the machine learning models to predict the profitability of the signal from stress indicator pairs trading strategy.
3. Compare the machine learning, Integrate the machine learning to pairs trading strategy and conclude the performance.

1.5 SCOPE

1. Our study mainly focuses on currency exchange based on Metatrader 5 platform. The historical data used in the study is provided by Metatrader server from January 2010 to December 2018. The pairs trading strategy is also operated Metatrader 5 coded in MQL5.
2. 5 major euro currencies which are EURAUD, EURGBP, EURCAD, EURNZD and EURUSD is studied in the thesis. The preposition is that all of 5 exchange rates consists of euro currency. The fundamental of these 5 rates is similar. Therefore, the pairs of these rates should have similar manners. Stress indicator pairs trading strategy is studied further. The machine learning algorithms which are focused to be trained in this thesis are Logistic regression, Artificial Neural Network and XGBoost.
3. RSI indicator period 14 is used to measure the spread of each pair.
4. The performance of the model is evaluated by 2 criteria

- F1 score from confusion matrix calculated by equation (1, the problem in this thesis is justified as classification problem, meaning the ML models can predict whether the trading signals are profitable or not. F1-score is implemented when the False Positives and False Negatives are important

$$\text{F1 score} = 2\,x\,\frac{\text{Precision} * \text{Recall}}{\text{Precision} + Recall} \quad (1)$$

where
$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \quad (2)$$

and
$$\text{Recall} = 2\,x\,\frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \quad (3)$$

In recall aspect calculated by equation (2, if the recall is low, it means that the algorithm fails to catch the profitable signal. Therefore, the machine learning model will reject a lot of signal. Traders would miss the opportunity to make profit. In precision aspect calculated by equation (3, if the precision is low, it means that the algorithm accepts a lot of false signals, causing huge losses. Thus, f1 score is implemented and maximized to balance the advantages of recall and precision.

- Profit from ML algorithm, Although the precision score is relatively high, the profit may not increase significantly, e.g. the False positive (false signal) causes huge loss. Therefore, the results of the trade are statistically tested by hypothesis test to prove whether the ML filter helps improve the trade or not.

2 LITERATURE REVIEW

2.1 LITERATURE SUMMARY

In this topic, some relevant papers used pairs trading and statistical arbitrage concept in quantitative trading method is studied and gathered. There are many frameworks used to rule the trading signal such as High frequency data, OLS, Machine learning and so on. It is called "Backend algorithm". Not only in stock market but also in various assets such as ETF, exchange rate future index and so on.

The concept and methodology to trade pairs of assets are summarized in this section. In [8], The authors studied the pair of 50 equities in Eurostoxx in High frequency timeframe 5-minute, 10-minute, 20-minute, 30-minute, 60-minute and daily. The benefit of the HFT (High Frequency trading) is it can gain more information ratio (IR) than lower frequency trade. The concept of this paper relies of pairs trading strategy, open the position when spread is wider than historical standard deviation. [8] introduced novel approach which can improve traditional pair trading strategy. First, they select the pairs best on in-sample information ratio and highest t-stat of ADF test of the residuals from cointegration regression on daily frequency then they form the 5 best trading pairs. [8] formed 4 models in the research, Cointegration, Rolling OLS, Double exponential-smoothing prediction and Kalman filter model. These my models estimate the beta parameter in linear relationship of pairs of stocks in equation (4)

$$Y_t = \beta X_t + \varepsilon. \quad (4)$$

In cointegration model, a pair is formed based on cointegration coefficient from the same industry. [8] argued that there is no evidence that the pairs that are not cointegrated cannot profit. Therefore, non-cointegrated pairs are not rejected. Next, the cointegration coefficient is estimated by OLS regression and take the stationary of the residual from linear equation by

Augmented Dickey-Fuller unit root test (hereinafter ADF) at 95% confidence level.

In Rolling OLS, the rolling beta is calculated by performing rolling OLS. Beta used in linear equation at time t is calculated from n previous points. The average OLS rolling window length for the 6 pairs found using genetic algorithm was 200 points, which was then used for all the remaining pairs and frequencies in the out-of-sample period.

Next model is Double exponential-smoothing prediction model (DESP). The model uses the Double exponential-smoothing equation to estimate the beta in linear equation using single exponentially smoothed series equation and double exponentially smoothed series equation.

Similarly, Kalmann filter was used to estimate the vary beta parameter. [8] claims that the Kalman filter is more robust than rolling OLS for adaptive parameter estimation.

Next, mean reversion based on Autocorrelation was introduced in [5]. [5] shows practical and simple methodology of pairs trading using autocorrelation method to select and trade the pairs of ETF and shares. The authors also purposed more frequency data using half daily close price. They expected that the result from more frequency data could gain more information ratio. The authors filtered the pair which have conditional correlation greater than 0.8. High correlation strongly support that these two ETF move together. The conditional correlation is calculated by equation from JPMorgan approach. Subsequently, the authors check whether the autocorrelation of the pair is within certain bounds. In the result section of the paper, the authors presented results for the pairs based on the pertinence to one of these autocorrelation intervals: from -1 to -0.4, from -0.4 to - 0.2, from -0.2 to 0, from 0 to 0.2, from 0.2 to 0.4 and from 0.4 to 1.

[4] implemented the mean reversion behavior of the two shares or two ETFs with simple trading strategy. The selected pairs of shares and ETFs have correlation greater than 0.8 and autocorrelation is categorized in the range.

The authors also choose only the pairs which have the highest normalized return. The parameters are selected based on the value which can achieve the good results in the in-sample-data across the daily and half-daily timeframe.

The result in [4] showed that the pairs of ETFs significantly provides greater information ratio than the pairs of shares. They also found that the half-daily timeframe provides higher information ratio than daily timeframe. The authors also found that the pairs which have negative first order autocorrelation are easier to forecast the returns than the pairs which have positive autocorrelation. The authors stated that the pairs can maintain their behaviors of in-sample period in the out-of-sample period, which means that investors can exploit the strategy from the historical information ratio of pairs.

The genetic algorithm was studied in [5]. The authors studied the novel approach of pairs trading strategy to construct the portfolio. [5] showed that the genetic algorithm-based model significantly outperforms the baseline and their model can generate the reliable result to beat the dynamic variable in financial behavior. The proper group of assets combined in the portfolio with the optimized parameter can perform the mean reversion. The set of asset assets, $S1$, ..., Sm, and the corresponding time series of stock prices, $P1(t)$, ..., $Pm(t)$; a statistical mispricing may be considered as a linear combination B = ($\beta1$, $\beta2$, ..., βm) so,

$$. \; ln\left(\frac{P(t)}{P(t_0)}\right) = \alpha(t - t_0) + \sum_{i=1}^{m}\beta_i ln\left(\frac{P_i(t)}{P_i(t_0)}\right) + \gamma(t) \tag{5}$$

where $\gamma(t)$ is a mean-reverting process and vector B consists the proportions of weight of capital assigned to each asset in the portfolio. The equation shows the behavior of synthetic assets which the price tends to move around the average price. At the market timing and pairs trading equations, the performance of trading system should profitable when the parameters in the equation are optimized at the market timing. Unfortunately, the parameters of the timing model consisting of n-period moving average, n-period and x band of Bollinger band that control the entry and exit criteria of standard deviation, weighting term (βi's) of asset capital in synthetic asset.

In [5], The authors applied the GA in 2 groups, 10 Stocks from the Semiconductor Industry and 10 Stocks with the Largest Market Capitalization. The authors set the baseline by allocating the capital to the assets equally, buy-and-hold and realizing the return of the 10 Stocks from the Semiconductor Industry and 10 Stocks with the Largest Market Capitalization. The authors define the training period and testing period. The task of the GA algorithm is to actively search for optimal proportion for long and short of assets in portfolio. In addition, the GA also searches for the optimal timing for buying and shorting the stocks adaptively using the Bollinger Bands.

Regarding to the result in [5], GA-based optimized trading strategy consists of optimal arbitrage and market timing models. Therefore, the portfolio is constructed based on advantages of arbitrage system which is optimized by GA algorithm

Another technique is price ratio. [6] aimed to create an empirical system to estimate the value of a pair of share prices ratio. Daily stock price ratio info, its lagged values, the ratio's departure from the mean and technical indicators such as momentum and Bollinger Band were calculated around this price ratio. These factors are then built into a nonlinear learning framework for analysis and forecast. The authors' anticipation is that investors can profit from the strategy if they can forecast the movement of the price ratio.

The rate of change of the denominator and numerator influences the price ratio. The authors use the technical indicator such as momentum and Bollinger band to explain the mean reverting properties of price ratio.

[6] presents the forecasting method to be used in pairs trading strategy. They select the pairs of stock from the same industry, anticipating the price ratio is mean reverting process. They use lagged technical indicator as input in machine learning algorithm. SVR, RF and ANFIS are used for predicting method. As a result, the input parameter can predict the price ratio effectively based on MAPE and MSE. The authors exemplify that momentum plays and important role involved with pairs trading.

In [14], The authors purposed the pairs trading strategy using statistical arbitrage of Sao Paulo stock exchange. They screened the stocks using cointegration method to select potential mean reverting pairs. The 1,225 stocks were paired up for all possible combinations and apply the Johansen and Engle-Granger cointegration test for long term mean reverting equilibrium. Next, all the cointegrated pairs are ranked by Sharpe ratio. 90 cointegrated pair were then selected in formation period. Firstly, they separated the data into training period and testing period. The parameters calculated in the training period were used in the consecutive testing period. They used one year for training and four months for testing.

For the signal, the algorithm is defined to trigger the trading signal based on z-score. To implement the strategy, traders need to follow trading rules, i.e. to calculate the criteria to open and to close a position. Next, the standardized spread between the stocks is calculated, and trades in-sample in simulated.

The spread is calculated by

$$\varepsilon_t = P_t^l - \gamma P_t^s. \tag{6}$$

where ε_t is the value of the spread at time t.

Next, they compute the z-score indicates the distance to the long-term mean equilibrium by following equation the

$$z_t = \frac{\varepsilon_t - \mu_\varepsilon}{\sigma_\varepsilon}. \tag{7}$$

The result referred to the out-of-sample analysis (from January 2006 to October 2012). The profitability shown had already been discounted for transaction costs. One can also note that the strategy presents a relatively low volatility of 12.49% in annualized terms, and a correlation coefficient with the market of -0.103, indicating that the strategy can be considered market neutral.

Another research studied further from mispriced index (MI) [4], which was based on the conditional probability modeled by copula method, to manipulate 1-minute frequency data of EURUSD vs EURSGD. The complex nonlinear models, i.e. CART, Logistic regression and Neural Network were implemented. The k-fold validation method was applied to evaluate the model performance, providing more reliable results and eliminating the data selection bias. The empirical results showed that MI feature influences the Neural Network performance but not in CART and Logistic regression. The profitability was significantly improved, i.e. Recall, F-measure, ROC and KS statistic.

[9] proposed a novel approach high frequency pairs trading system in Taiwan Stock Index Futures (TX) and Mini Index Futures (MTX) market based on deep learning techniques. [9] used the time series visualization method to convert historical volatilities with different time frames into 2D images which were able to catch arbitrage signals. Moreover, [9] improved

convolutional neural networks (CNN) model by integrating the financial domain knowledge and filterbank mechanism. [9] proposed Filterbank CNN to obtain high-quality features by substituting the random-generating filters with the arbitrage filters. According to the results, the accuracy was improved through this method.

Not only algorithms affect the profitability, but also external factors. [13] studied further from [3], the classic pairs trading strategy, disentangling which factors influence the profitability of the strategy across the market, industries, macroeconomic circumstances and business operations. The author focuses in two main parameters, volatility, and correlation, that how they influence the pair selection, trading algorithm and total return.

The authors use the traditional pair selection criteria from [3]. Then they calculated and ranked the correlation and volatility the investigate the effect in pairs trading strategy to the total return. Next, they studied further how the volatility and correlation affect the trading frequency. They classified the pairs into five quantiles, Corr_Q1 (Corr_Q5) includes the pairs with the lowest (highest) pair correlation, quintile VOL_Q1 (VOL_Q5) includes the pairs with the lowest (highest) level of pair volatility. The authors found that the return per trade for normal trades increased for higher pair volatility levels within same level of correlation and decreased for higher correlation within same level of volatility.

For the trading frequency, the authors found that high volatility pairs generate less trade than low volatility pair. Moreover, High correlation pairs produces significantly more trades than low correlation pairs. In summary, low volatility and high correlation generate the higher number of trades.

Table 2-1 Summary of the Literature by trading frequency (Timeframe), backend algorithms and assets traded in literatures

Author	Timeframe	Algorithm	Asset
Rudy et al.	5-minute, 10-minute, 20-minute, 30-minute, 60-minute and daily	Cointegration, Rolling OLS, Double exponential-smoothing prediction and Kalman filter model.	50 equities in Eurostoxx
Dunis et al.	Daily and Half daily	Autocorrelation	ETF shares
Huang et al.	Daily	Genetic algorithm	10 Stocks from the Semiconductor Industry and 10 Stocks with the Largest Market Capitalization in China
Ghosh et al	Daily	SVR, RF and ANFIS predicting the price ratio	Indian stock market
Caldeira et al.	Daily	Cointegration method	1,225 stocks in Sao Paulo stock exchange
C. C. F. Chu et al.	1-minute	Copula method Integrated with CART, Logistic regression and Neural Network	EURUSD vs EURSGD
Y. Chen et al.	Tick data	Image and pattern recognition with CNN	Taiwan stock Index Futures and Mini Index Futures
This thesis	Hourly	Stress indicator with Machine Learning	Foreign exchange

The techniques or backend algorithms from each literature are summarized in Table 2-1 including the trading frequency and traded assets.

This thesis proposes novel approach based on the stress indicator. According to the problem statement. Stress indicator pairs trading strategy works in some circumstance such as same fundamental, high volatility, high correlation, etc. and has not been extended to any other assets. Therefore, the strategy in foreign exchange in hourly timeframe is extended, and the machine learning algorithms are implemented to improve the performance of the strategy which is new to pairs trading strategy researched by the others.

2.2 RELATED THEORIES

2.2.1 Artificial neural networks

Artificial neural networks (ANNs), simple termed as Neural Networks, the inspiration that resulted in this system is due to the concept of biological neural networks that make up the entire brain of the animals.

This neural network is based on the collection of the linked units or nodes known as artificial neurons, that openly develop the neurons in a brain. Signals are transmitted to various neurons considering the links existing in it that resemble to that of a brain which are established through neurotransmitters. Artificial neurons communicate in such a way that a single neuron accepts a signal from the other neuron, measures it and communicates with other neurons linked to it. Since, the "signal" received by every neuron is a real number, its ending result depends on the non-linear function that calculates it by using the initial sources. The learning process stays in progress while the weights of the neurons and edges alternate. The quality of a signal either increases or decreases reciprocally with the weights at the links. The output is linked with the limitation of the neurons focusing on the signal which is imparted provided the aggregate signal exceeds that standard limit value. Under normal circumstances, arrangement of the neurons is done in layers. There is always an impact of the multiple layers on the initial sources which might cause changes. The signals are to cross the layers on various occasions, beginning from the initial layer (input layer) to reach the end goal that is the final state (output state).

Training

The training of a neural network requires the formation of the connections of weights depending on the probability. These connections are established using the processing models that encompass two important constitutes, “input” and “result” that are known, which make this connection possible. These two parts of the processing models are put within the neural net. The preparation of a neural network from a given model is generally led by deciding the contrast between the handled output of the neural network (probably a hypothesis) and an objective output. This is considered to be an error. The network at that point alters its weighted connections as per a learning rule and utilizing this value of the error. Progressive modifications will make the neural network produce an ending goal which is progressively like the objective goal. After an adequate number of these changes the preparation can be ended dependent on specific rules. Which is termed as Supervised Learning. These structures function through the process of learning to attempt errands by considering the models. Keeping in view, a major part of remains the same through accurate task guidelines. Explaining using an example, in image recognition, the identification of the pictures containing an animal, specifically a “feline” in this example, through investigation of example pictures that maybe named similar to what is displayed in the pictures. In this case, either feline or no feline might be the names. Finally, the outputs are under consideration for identifying the felines in a picture. This performance is not based on their earlier encounter with any sort of pictures containing a feline, focusing on the features of a cat, that it has fur, a tail, feline like countenances and other physical cat characteristics. They opt a different method that involves recognition of the qualities of an image with the example pictures that they have processed earlier during training.

Model

ANNs began in the form of a venture to maneuver the systemization of the human mind to observe errands that regular calculations failed with. They after some time, reoriented towards improving observational outcomes, generally abandoning ventures to stay persistent along with biological antecedents. The linkage of the neurons exists in various forms, to permit the ending result of certain neurons to turn into the initial values for the other neurons. The network shapes a coordinated, weighted diagram.

An artificial neural network comprises of an assortment of recreated neurons. Either connections between the different neurons are found to be similar to that of biological axon-synapse-dendrite associations. A weight is assigned to evert connection, that depicts the impact quality of a node on a different node.

Components of ANNs

Neurons

ANNs are made out of artificial neurons that are thoughtfully gotten from natural neurons. Each artificial neuron has information sources and produce a solitary yield which can be sent to different neurons. The inputs can be the component estimations of an example of data that exists externally, for example, pictures or archives, or they can be the yields of different neurons. The goal of the last output neurons of the neural net reach the errand, for example, recognition of an object in a picture.

To discover the result of the neuron, it is required to initially find the sum of the weights of all the initial sources , the weights are calculated using the weights of the links that are established from the initial sources to the final neuron. A term is added to the total sum which is biased. Such a sum which has been weighted this way is usually termed as activation. This weighted whole is then gone through a (normally nonlinear) activation faction to create the final goal. The pictures and documents are considered as a part of the data that exists externally. A definitive output achieves the requirements, for example, perceiving an article in a picture.

Connections and weights

Network is composed of associations, every association giving the yield of one neuron as a contribution to another neuron. As previously mentioned, each link providing the ending point of a single neuron as the initial value for the next neuron. A corresponding importance depicting each weight that is assigned to each of the edges. For any neuron, there exists more than one connection, both input and output.

Propagation function

The inputs and the connections of the neurons are computed using a propagation function. The input of a neuron is calculated using the previously connected neurons and the neuron associations are calculated as a sum depending on weights. As aforementioned, the propagation function can also use a bias value for finding the final output.

Organization

Layering is done of the multiple neurons in a sequential manner, specifically in an important concept of Machine Learning that is Deep Learning. The linkage of the neurons in a single layer is done towards the preceding layer from the previous layer. External information is taken into the network through an initial layer which is known as the input layer. Similarly, the layer that produces the required result is known as the output layer. Hidden layers may or may not exist between them. They can either be zero or a few hidden layers. More than one association designs can be conceived between any two layers of the neural network. The edges are formed between the neurons existing in one layer communicating with the neurons in the very next layer. However, this linkage is established through a single neuron of one layer connected to the preceding layer, consequently lessening the amount of neurons existing in that layer, which is considered as the method of pooling. Neurons with just such associations structure can lead to the formation of an directed acyclic graph. Such neuron associations lead to network which are usually termed as feed-forward networks. Networks that consider associations laying amidst the same or the last layers are called recurrent networks.

Hyperparameter

Hyperparameter is considered a consistent boundary that has a worth that is settled prior to the learning cycle starts. The estimations of boundaries are determined through training. Instances of hyperparameters incorporate learning rate, the quantity of hidden layers and group size. Hyperparameter estimations are subjected to different hyperparameters. Seeing an example, the magnitude of certain layers relies upon the general number of layers.

Learning

Learning is the variation of the network to all the more likely handle an assignment by considering test samples. Learning includes altering the network weights (and discretionary limits) to enhance the precision of outcome. An end is put to it by applying a limitation on the errors. Learning finishes while looking at extra observations does not conveniently diminish the rate of the error. Even in the wake of learning, the rate ordinarily does not arrive at 0. In the event that subsequent to learning, the error is excessively large, the network regularly should be updated. A cost function is used which is assessed intermittently through the process of learning. This function is characterized to finish it. However, as its yield keeps on lessening, reciprocally, learning proceeds. The cost is regularly characterized as a measurement whose worth must become relative. The yields are real numbers, and when the error is not too big, the distinction between the yield (very likely a feline) and a correct option (feline) may be little. When considering the perceptions, the endeavors of learning are to lessen the major differences. Optimization theory and statistical estimation are utilized by many learning models.

Learning rate

In every perception, the magnitude of the remedial advances is characterized by the rate of learning that the model can use to make changes in the errors. The preparation time is abbreviated using a high learning rate, however with less extreme exactness, meanwhile, more time is consumed by a lower learning rate, yet having greater possibility for more noteworthy precision. Optimizations, for example, Quickprop are essentially pointed toward accelerating mistake minimization, while different upgrades fundamentally attempt to expand unwavering quality. So as to dodge wavering inside the network, for example, for making the rate of convergence higher along with rotating the association weights, a versatile learning rate that increases/decreases, whichever suits best, is used for the rectifications. The concept of momentum permits the harmony among inclination and also the past modification still needed to be weighted with end goal. This harmony states that the weight modification depends somewhat on the past change. A force near 0 underlines the inclination, while a worth near 1 underscores the previous change.

Cost function

As long as the characterization of a cost function is conceivable, every now and again the decision is controlled by the capacities attractive properties, (for example, convexity) or in light of the fact that it emerges from the model (e.g., in a probabilistic model the model's posterior likelihood has a chance of being utilized in the form of a backwards cost).

Backpropagation

Backpropagation is a strategy to change the association weights to make up for every error discovered during learning. During the process of learning, the association weights need to be upgraded in return of every error observed in the network. This methodology is termed the process of Backpropagation. The error sum is viably partitioned among the associations. The connections existing in the network have the error sum partitioned over it. While considering the weights, a given state is related to the subordinate of the cost function that is determined by backprop. The weights need to be updated using either stochastic gradient descent or different strategies, for example, Extreme Learning Machines, "No-prop" organizations, preparing excluding backtracking, "weightless" organizations, and ADAM

Types

ANNs have developed into an expansive group of methods that have progressed the cutting edge over various areas. The least difficult sorts have at least one static segments, including number of units, number of layers, unit weights and geography. Dynamic sorts permit at least one of these to develop by means of learning. The last are considerably more confounded yet can abbreviate learning periods and produce better outcomes. A few kinds permit/expect figuring out how to be "supervised" by the administrator, while others work freely. A few kinds work simply in equipment, while others are absolutely programming and run on broadly useful PCs.

A portion of the principle advancements involve: convolutional neural networks that prove to be good enough to demonstrate especially in being effective in preparing visual and other two-dimensional information; long transient memory dodge the evaporating angle issue and can deal with signals having a blend of low and high recurrence segments supporting enormous jargon discourse acknowledgment, text-to-speech amalgamation, and photo-real talking heads; serious networks, for example, generative adversarial networks that hold numerous networks (of shifting structure) rival one another, on errands, for example, dominating a game or on misleading the adversary about the realness of information.

2.2.2 Logistic regression

There are two potential results of the classification issues whose likelihood is determined by the Logistic regression. This is considered as the augmentation in terms of the classification issues of a linear regression model.

Theory

An answer to the arrangement is logistic regression. A logistic function is utilized by a regression model to make the output stay in between 0 and 1 of a linear equation. This is done in the place adjusting a straight line or a hyperplane. The calculated logistic function is characterized as:

$$\text{logistic}(\eta) = \frac{1}{1 + \exp(-\eta)} \tag{8}$$

The progression from linear regression to logistic regression is somewhat clear. Considering the linear regressions model, we have demonstrated the connection among result and highlights with a linear equation:

$$\widehat{y^{(i)}} = \beta_0 + \beta_1 x_1^{(i)} + \cdots + \beta_p x_p^{(i)} \tag{9}$$

We require the values of the likelihoods for classification within the range of 0 and 1. Therefore, we use the logistic function to bound the right side of the equation within it. This helps the result to always end within the desired range of 0 and 1.

$$P\left(y^{(i)} = 1\right) = \frac{1}{1 + exp\left(-\left(\beta_0 + \beta_1 x_1^{(i)} + \cdots + \beta_p x_p^{(i)}\right)\right)} \tag{10}$$

Logistic Regression can be implied with classification to deduce better results. The efficiency can be increased for classification with or without logistic regression by using a standard value of 0.5. It is to be noted, that consideration of any extra point in a graph does not generally influence the curve.

Interpretation

There is a difference between the understanding of the weights between the logistic regression and the linear regression. This is due to the results obtained through logistic regression is a likelihood that ranges between 0 and 1. And hence, the weights have no linear impact on the likelihood. The logistic function changes the sum of the weights into a likelihood. Because of this, it is required to transfer the linear term to the right side of the function for a correct understanding.

$$log\left(\frac{P(y=1)}{1-P(y=1)}\right) = log\left(\frac{P(y=1)}{P(y=0)}\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p \quad (11)$$

"Odds" is a term existing inside the log () function which is actually equivalent to the likelihood of the event over the likelihood of no event. And it is when shifted inside the logarithm, it is termed as "log odds".

This formula depicts that the logistic regression model is to be taken as a linear model for the log odds. With a touch of rearranging of the terms, you can make sense of how the hypothesis changes when one of the characteristics x_j is changed by 1 unit. To do this, we would first be able to apply the exp () capacity to the two sides of the condition:

$$\frac{P(y=1)}{1-P(y=1)} = odds = exp(\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p) \quad (12)$$

At that point we look at what happens when we increment one of the characteristic values by 1. In any case, rather than taking a gander at the difference, we take a gander at the proportion of the two predictions: Then we look at what happens when we increment one of the component values by 1. However, rather than taking a gander at the difference, we take a gander at the proportion of the two forecasts:

$$\frac{odds_{x_j+1}}{odds} = \frac{exp(\beta_0 + \beta_1 x_1 + \cdots + \beta_j (x_j + 1) + \cdots + \beta_p x_p)}{exp(\beta_0 + \beta_1 x_1 + \cdots + \beta_j x_j + \cdots + \beta_p x_p)} \quad (13)$$

The following rule is applied:

$$\frac{exp(a)}{exp(b)} = exp(a - b) \tag{14}$$

More than one terms are removed:

$$\frac{odds_{x_j+1}}{odds} = exp\big(\beta_j(x_j + 1) - \beta_j x_j\big) = exp(\beta_j) \tag{15}$$

At long last, we have something as basic as exp()of a component weight. An adjustment in an element by one unit alters the odds ratio (multiplicative) by a factor of $\exp(\beta_j)$

We could likewise decipher it along these lines: A change in x_j by one unit builds the log odds ratio by the estimation of the comparing weight. A great many people decipher the odds ratio since pondering the log () of something is known to be challenging for the mind. Deciphering the odds ratio as of now requires some becoming acclimated to. For instance, in the event that you have chances of 2, it implies that the likelihood for y=1 is twice as high as y=0. In the event that you have a weight (= log odds ratio) of 0.7, at that point expanding the particular feature by one unit duplicates the chances by exp (0.7) (roughly 2) and the chances change to 4. Yet, generally you don't manage the chances and decipher the weights just as the odds ratio. Since for really computing the chances you would need to set an incentive for each component, which possibly bodes well on the off chance that you need to take a gander at one explicit occurrence of your dataset. Following are the logistic regression model understandings accompanied by various types of features:
Mathematical characteristic: The odds ratio alters through a factor of $\exp(\beta_j)$ when one unit in increased in the value of the feature

- Binary Categorical Characteristic: One of the two estimations of the characteristic is the reference classification (in certain dialects, the one encoded in 0). Changing the characteristic x_j from the reference class to the next classification turns the odds that were estimated through a factor of $\exp(\beta_j)$

- Classification Characteristics with multiple classifications: One answer for managing various classes is found to be a one-hot-encoding, implying that every class has its own section. There is only the requirement of L-1 sections having L classification for a classification characteristic, else, it is over-defined. The L-th class is then the reference classification. Linear regression provides multiple encodings, one of which fits to be used. The understandings of binary classifications are considered to be proportionate to the understandings of every class.

Intercept β_0 : Odds which are estimated are $\exp(\beta_0)$ only when the classification characteristics are placed under the class of reference and a zero is obtained for every mathematical classification.

2.2.3 XGBoost

A Machine Learning method that is based on decisions is usually known as XGBoost. It utilizes a framework that has been boosted using slopes. The artificial neural networks are found to give an outstanding performance in comparison to the other methodologies while considered the issues related to the hypothesis that uses data that is not structured, such as pictures or text. Notwithstanding, these algorithms based on decisions trees have proved to work brilliantly when considering the minutely structured data that might be in a tabular form. If you don't mind see the graph beneath for the development of tree-based calculations throughout the past.

XGBoost Algorithm was created as an examination venture at the University of Washington. Tianqi Chen and Carlos Guestrin introduced their paper at SIGKDD Conference in 2016 and found a massive place in the Machine Learning world readily. Since its presentation, this calculation has not exclusively been credited with winning various Kaggle rivalries yet in addition for being the main impetus in the engine for a few forefront industry applications. Therefore, there is a solid network of information researchers adding to the XGBoost open-source ventures having ~350 supporters and ~3,600 submits over GitHub. The calculation separates itself in the accompanying manners:

1. A wide scope of uses: Can be utilized to settle regression, grouping, positioning, and client characterized hypothesis issues.

2. Smooth running on various operating systems including Windows, Linux, and OS X.

3. Languages: It works with almost all the high-level programming languages such as C++, Python, R, Java, Scala, and Julia

4. Cloud Integration: It backs up AWS, Azure, and Yarn clusters and collaborates with Flink, Spark, and various ecosystems.

The intuition of the XGBoost

The building of intuition is not an easy task and complications may occur in its completion in comparison to the simple decision trees that are easier to perceive especially when it is built for the future algorithms that are tree-based. See underneath for a basic relationship to more likely to comprehend the advancement of the algorithms that are tree-based.

Envision that you are an employing administrator meeting a few competitors with incredible capabilities. Each progression of the advancement of tree-based algorithms can be seen as an adaptation of the process of the interview meeting.

1. **Decision Tree**: Each employing supervisor has a lot of rules, for example, instruction level, number of long stretches of understanding, performance in the interview. A decision tree is practically equivalent to a recruiting supervisor talking with applicant’s dependent on their own measures.

2. **Bagging**: Presently envision rather than a solitary questioner, presently there is a meeting board where every questioner has a vote. Bagging or bootstrap accumulating includes joining contributions from all questioners for a ultimate conclusion through a vote based democratic cycle.

3. **Random Forest**: It is an algorithm based on the concept of bagging with a key contrast wherein just a subset of characteristics is chosen aimlessly. As such, every questioner will just test the candidate on certain arbitrarily chosen capabilities (for example a specialized meeting for testing programming aptitudes and a conduct meet for assessing non-specialized abilities).

4. **Boosting**: This is an elective methodology where every questioner modifies the assessment rules dependent on criticism from the past questioner. This 'helps' the effectiveness of the meeting cycle by sending a more powerful assessment measure.

5. **Gradient Boosting**: An important instance of boosting where errors are limited using the gradient descent algorithm. For example, the procedure counseling firms influence by utilizing case meetings to remove less qualified competitors.

6. **XGBoost**: Consider XGBoost gradient advancing on 'steroids' (the same reason it is also known as Extreme Gradient). It is an ideal mix of programming and equipment streamlining procedures to yield predominant outcomes utilizing less computer resources in the most brief measure of time.

The performance

The gradient descent architecture is utilized through the application of guidelines of boosting weak learners (CARTs generally). This application is done through both the XGBoost and Gradient Boosting Machines (GBMs) that are grouped within the category of tree-based algorithms.

Development of the System:

1. **Parallelization**: XGBoost uses a cycle of successive tree building utilizing parallelized usage. It’s conceivable because of compatible idea of loops utilized for building base students; the external loop that lists ending nodes of a tree, and the second internal loop that figures characteristics. Settling of loops bounds parallelization on grounds that without finishing the inward loop (all the more

computationally requesting of the two), the external loop can't be begun. Subsequently, to improve run time, the request for loops is exchanged utilizing the process of initialization through a worldwide search everything being equal and arranging to utilize equal strings. This switch enhances algorithmic execution by counterbalancing any parallelization overheads in calculation.

2. **Tree Pruning:** Halting measure for tree parting inside GBM structure is insatiable naturally and relies upon a non-positive misfortune, basis at the purpose of split. XGBoost utilizes 'max_depth' boundary as indicated rather than basis first and commences pruning trees in reverse. This 'depth first' methodology enhances execution of computations essentially.

3. **Hardware Optimization**: A methodology that is intended to utilize the external device resources. This is cultivated by reserve mindfulness by assigning interior buffers in each string to store statistics of the gradient. Further improvements, for example, 'out-of-core' processing upgrade accessible storage space while taking care of large information frames that are impossible to fit the storage.

Algorithmic Enhancements:

1. **Regularization**: Castigation is performed by the algorithm on complicated figures for avoiding misclassifications by using the two kinds of regularizations i.e. LASSO (L1) and Ridge (L2).

2. **Sparsity Awareness**: XGBoost normally concedes scanty characteristics for initial values through the natural 'learning' of the most appropriate value that is not available. It is worth relying upon preparing the loss and manage various sorts of sparsity designs in the information all the more effectively.

3. **Weighted Quantile Sketch:** The weighted Quantile Sketch algorithm is employed by the XGBoost for the efficient search of the highest split points amidst weighted datasets.

4. **Cross-validation**: A cross-validation approach is implied through this algorithm at every repetition. This is done by removing the need to unequivocally program this inquiry and to indicate the specific number of boosting cycles required in a solitary run.

3 METHODOLOGY

The methodology to develop predicting model is thoroughly described in this section. The lagging indicators from trading history are collected. The potential pairs of foreign exchange rates are screened and used to develop the predicting model by machine learning algorithm. Many techniques are implemented to maximize the performance of the machine learning algorithm such as, feature selection, k-Fold cross validation, regularization etc. Next, when the machine learning models are developed, Machine learning algorithms are integrated to normal pairs trading strategy to help improve the trading performance. Finally, the performance of machine learning is tested by the out-of-sample data and statistically evaluated to justify whether the machine learning help improve the pairs trading strategy or not in the next section. The process of methodology is shown in Figure 3-1.

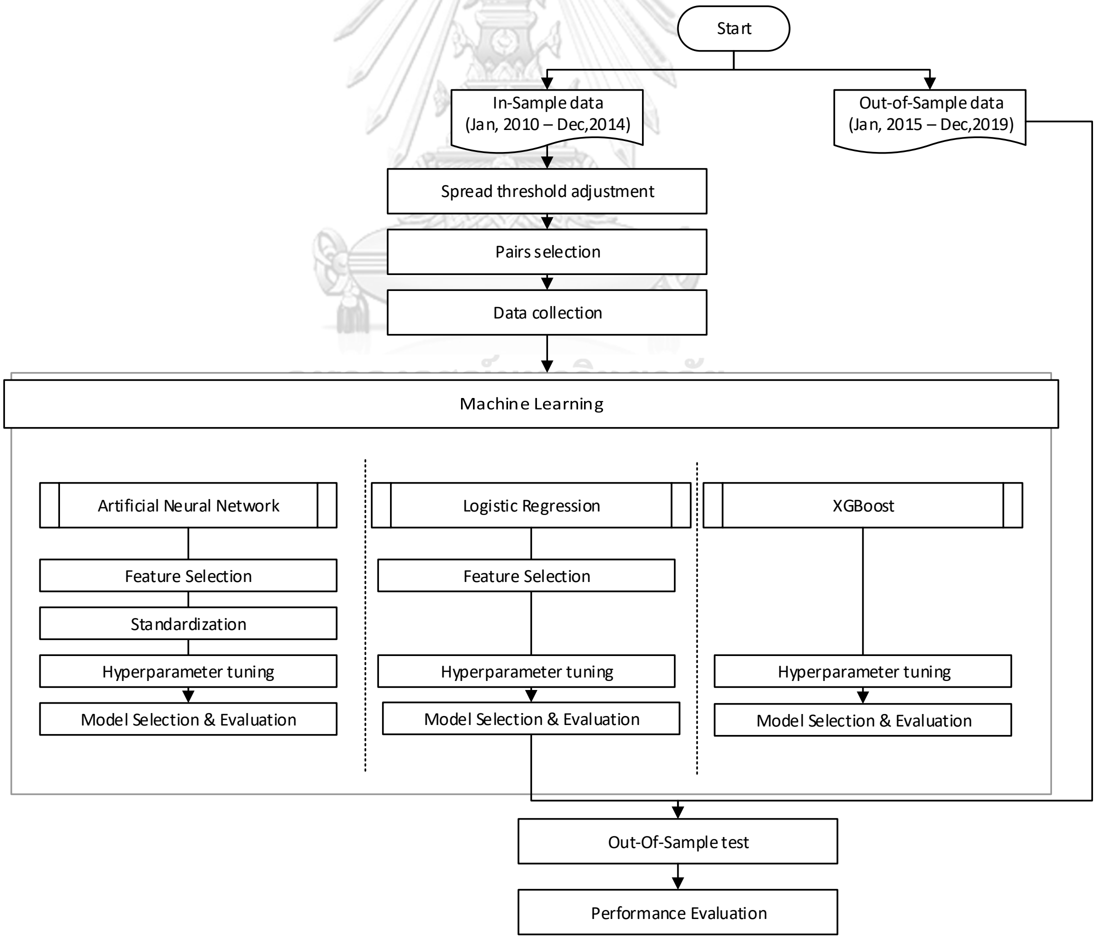

Figure 3-1 The process of methodology

3.1 SPREAD THRESHOLD ADJUSTMENT

In this thesis, Only the relationship of EURO currency exchange rates is studied only. The list of capable EURO exchange rates is EURAUD, EURCAD, EURGBP, EURNZD and EURUSD. Next, the exchange rates are paired totally 10 pairs and traded in pairs trading strategy. After finished selecting the exchange rates, the spread threshold which is the criteria to trigger the signal is tested in this section. The criteria to set the spread threshold is not available, but the fact is that if the spread threshold is set too low, the signal is generated more frequently and high chance of false signal. In the other words, A lot of signals are triggered, but most signals are false. Inversely, if the spread threshold is set too high, only few signals are triggered, e.g. 3 orders a year, and cannot guarantee that these signals are profit. The methodology is to trial and error the threshold levels from the in-sample data from January 2010 to December 2014 and collect the average return per trade and the number of trades. Relative Strength Index (RSI) period 14 is used as stress indicator to measure the momentum. The spread calculation is calculated by equation *(*16

$$\mathrm{Spread}_{\mathrm{t}} = \mathrm{RSI}_{\mathrm{t,leg1}} - \mathrm{RSI}_{\mathrm{t,leg2}} \qquad (16)$$

Next, the exchange rates are paired and tested for profitability in every pair by varying the threshold. ST = {15,20,25,30,35,40}. Then, the profit/loss is gathered from January 2010 to December 2014. To measure the performance, position size of 1 Lot is traded in entire thesis. The result from the strategy is not capable to anticipate. The proper solution is to test every possible pairs and gather all the trading result. These results represent the performance of the pairs trading strategy. Therefore, the suitable pairs to trade in pairs trading strategy is justified in next section.

The number of pairs of exchange rate traded by the pairs trading strategy is 10 pairs. Next, the normal pairs trading strategy is run through all these pairs. The number of trades and the average profit by varying the spread threshold is reported in Table 3-1

Table 3-1 The result of varying the spread threshold in EURAUD vs EURCAD

Currency pair	EntryThreshold	Average profit per trade	Trades
EURAUDEURCAD	40	-21.55028	36
	35	32.51566	106
	30	17.42340	282
	25	-21.77577	620
	20	-24.12543	1166
	15	-33.35163	2010
Currency pair	EntryThreshold	Average profit per trade	Trades
EURAUDEURGBP	40	-54.44413	138
	35	-61.42644	298
	30	-68.40626	550
	25	-56.77171	900
	20	-49.42588	1370
	15	-45.65184	2086
Currency pair	EntryThreshold	Average profit per trade	Trades
EURAUDEURNZD	40	-53.60900	30
	35	-30.15429	70
	30	-72.68826	132
	25	-55.11324	324
	20	-50.72861	782
	15	-47.05596	1642
Currency pair	EntryThreshold	Average profit per trade	Trades
EURAUDEURUSD	40	39.77083	156
	35	24.55435	322
	25	-9.11951	1008
	30	-26.35122	556
	20	-25.55455	1498
	15	-73.18280	1366
Currency pair	EntryThreshold	Average profit per trade	Trades
EURGBPEURCAD	35	45.16850	160
	40	-18.69568	74
	30	24.73401	352
	25	5.24330	624
	20	-10.83423	994

	15	-20.97176	1522
Currency pair	EntryThreshold	Average profit per trade	Trades
EURGBPEURNZD	40	5.56732	138
	35	-9.46457	280
	30	-25.41711	550
	25	-47.72495	908
	20	-56.70983	1382
	15	-50.26688	1990
Currency pair	EntryThreshold	Average profit per trade	Trades
EURGBPEURUSD	40	27.37236	72
	35	-3.50928	152
	30	3.11846	358
	25	-17.69905	640
	20	-2.91896	1192
	15	-13.82044	1978
Currency pair	EntryThreshold	Average profit per trade	Trades
EURCADEURNZD	40	-5.69033	60
	35	-21.48966	174
	30	-71.81466	324
	25	-44.48028	676
	20	-38.85776	1188
	15	-47.90485	2024
Currency pair	EntryThreshold	Average profit per trade	Trades
EURCADEURUSD	40	-5.69033	60
	35	-21.48966	174
	30	-71.81466	324
	25	-44.48028	676
	20	-38.85776	1188
	15	-47.90485	2024
Currency pair	EntryThreshold	Average profit per trade	Trades
EURNZDEURUSD	40	-31.92833	150
	35	-36.79160	326
	30	-37.54521	578
	25	-45.23803	938
	20	-51.56256	1424
	15	-74.19230	1354

3.2 Pair Selection

The optimal spread threshold is selected by 2 criteria, the highest average return per trade which is greater than zero and the number of trading records is greater than 200 records. But if the pair is not satisfying the 2 criteria, the pair is rejected and considered as non-potential pair.

The result from Table 3-1 clearly shows that the higher spread threshold is, the higher average return per trade is. But when the spread threshold goes too high, the average return per trade is not always positive at the 40-spread threshold. According to EURAUD vs EURCAD, for the number of trades, although the 35-spread threshold provides more average profit than 30-spread threshold, the number of trades is only 53 records meaning that, during 5 years from January 2010 to December 2014, the position is opened only 1 order a month. Therefore, the spread threshold is set 30. For the other pairs, the same rule is implemented in EURAUD vs EURCAD. Finally, there are 4 pairs of exchange rate which satisfy the criteria. The result is summarized in Table 3-2.

Table 3-2 Summary of the potential pairs and thresholds

Currency pair	Entry Threshold	Average profit per trade	Trades
EURAUDEURCAD	30	17.42340	282
EURAUDEURUSD	35	24.55435	322
EURGBPEURCAD	30	24.73401	352
EURGBPEURUSD	30	3.11846	358

3.3 DATA COLLECTION

After finished the pair selection, the parameters from previous section are set in strategy tester traded in In-sample price data and use 1 lot position size as usual.

The optimal spread threshold is set up in the strategy tester to trade and gather the trading data. When signal triggers the tester. The strategy opens the position and gathers the lagging indicator such as RSI, Correlation, Volatility, Position status (Short or Long leg1) and other crucial listed in the Table 3-3

Table 3-3 Summary of the factors collected in data collection period

Variable	Value	Function
Input features (or predicting features)		
MomLeg1[0]	RSI value of leg 1	Indicator of the momentum of asset in leg 1 in period 14
MomLeg2[0]	RSI value of leg 2	Indicator of the momentum of asset in leg 2 in period 14
VolatilityLeg1[0]	Volatility value of leg 1	measure the realized volatility of leg 1 in period 20
VolatilityLeg2[0]	Volatility value of leg 2	measure the realized volatility of leg 2 in period 20
VolatilityDiff	VolatilityLeg1[0] minus VolatilityLeg2[0]	measure the difference between leg 1 and leg 2
MAShortValue_Leg1[1] -MALongValue_Leg1[1]	Moving average period 12 minus Moving average period 26 of asset leg 1	measure the strength of the trend in asset leg 1
MAShortValue_Leg2[1] -MALongValue_Leg2[1]	Moving average period 12 minus Moving average period 26 of asset leg 2	measure the strength of the trend in asset leg 2
beta	Beta parameter OLS regression period 10,20,50,100	Measure the linear relation of the assets
Correlation	Pearson correlation period 14,20,50,100	measure the correlation of the asset
Type	0; short asset leg 1 and long asset leg 2 1; short asset leg 2 and long asset leg 1	indicate the type of position

Output features (or predicted features)		
P_L1	realized profit(loss) of leg 1 in dollar	
P_L2	realized profit(loss) of leg 2 in dollar	
Profit	Profit_leg1 + Profit_leg2	
Profit_L1	0; loss 1; profit	indicate whether leg 1 is profit or loss
Profit_L2	0; loss 1; profit	indicate whether leg 2 is profit or loss
Total_Profit	0; loss 1; profit	indicate whether pairs of assets are profit or loss

MomLeg1 and MomLeg2 (or RSI)

Mom, or Momentum, is measure by Relative Strength Index (RSI) which is well-known technical indicator in quantitative finance.

Volatility

According to [15], the volatility of exchange rate is measured by the square root of variance of logarithmic return of N period which is defined as

$$s^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})^2. \qquad (17)$$

Where, x_i are the logarithmic returns, $\overline{x}$ is the mean return in the sample and N is the sample size. In finance, it is difficult to measure the mean of return to calculate variance. So, the mean of variance is set to zero and the variance measured in the formula is not population, the formula is corrected to be

$$s^2 = \frac{1}{N-1}\sum_{i=1}^{N}{x_i}^2. \qquad (18)$$

From Jensen's inequality, it states that

$$. E(s) = E\left(\sqrt{s^2}\right) < \sqrt{E(s^2)} = \sqrt{\sigma^2} = \sigma \qquad (19)$$

Therefore, the unbiased estimator is needed to be corrected by following equation

$$\overline{s} = b(N)\sigma \quad . \tag{20}$$

Where the correction factor

$$. \, b(N) = \sqrt{\frac{2}{N}} \frac{\Gamma\left(\frac{N}{2}\right)}{\Gamma\left(\frac{N-1}{2}\right)} \tag{21}$$

So, s/b is an unbiased estimator of the population standard deviation

In our study, the N-period of hourly timeframe for weekly volatility is 120 which $b(N)$ is approximately 1. Therefore, the standard deviation formula is capable to be used as unbiased volatility estimator.

Volatility Difference

Volatility Difference is the simple difference of Volatility of Leg 1 period minus Volatility of Leg 1 period 20

$$\text{Vol\_Diff}_i = \text{VolVol\_leg1}_{i,20} - \text{Vol\_leg2}_{i,20} \tag{22}$$

MAShortValue_Leg1 - MALongValue_Leg1 and MAShortValue_Leg2 - MALongValue_Leg2

These lagging indicators are the difference Simple Moving Average in each leg. The short moving average means the period of the point of data is relatively lower than the long moving average, for example, Moving Average period 12 minus Moving average period is MAShortValue minus MALongValue, respectively. Leg1 and Leg2 represent the exchange rate in each side.

Beta

Beta, or Ordinary Least Square Beta, is the parameter in linear regression from least square method by the following equation

$$\hat{\beta} = \frac{\sum x_i y_i - \frac{1}{n}\sum x_i \sum y_i}{\sum x_i^2 - \frac{1}{n}(\sum x_i)^2} \tag{23}$$

where x_i is the open price in hourly period on Leg 1 at $t = i$

y_i is the open price in hourly period on Leg 2 at $t = i$

Correlation

Correlation or Pearson correlation coefficient is a parameter that measures linear correlation between two variables which is open price of Leg1 and Leg2 following by the equation

$$\mathbf{r} = \frac{\sum(\mathbf{x_i} - \bar{\mathbf{x}})(\mathbf{y_i} - \bar{\mathbf{y}})}{\sqrt{\sum(\mathbf{x_i} - \bar{\mathbf{x}})^2}\sqrt{\sum(\mathbf{y_i} - \bar{\mathbf{y}})^2}} \qquad (24)$$

where x_i is the open price in hourly period on Leg 1 at $t = i$

y_i is the open price in hourly period on Leg 2 at $t = i$

Type

This parameter indicates the type of the orders

0; short asset leg 1 and long asset leg 2

1; short asset leg 2 and long asset leg 1

P_L1, P_L2 and Profit

P_L1 is the realized profit/loss of Leg 1 position and of Leg 2 position, respectively. Profit is the summation of P_L1 and P_L2

Profit_L1, Profit_L2 and Total_Profit

These factors are the categorial parameter of the P_L1, P_L2 and Profit. 0 mean loss and 1 means profit.

3.4 Machine Learning Algorithm

After the data are gathered from tester, the performance is aimed to improve by applying the Machine Learning model. Researchers suggest that the volatility and correlation of pairs play an important role in pairs trading strategy. For example, [10] suggests that trader should implement volatility filter in the strategy. The authors also illustrate that the return per trade is significantly higher and frequency of trade is increased when the volatility is high. Also in [13], The authors disentangled that the correlation and volatility influence that return per trade and trading frequency. Our implementation is using Machine learning to learn the data and find the hidden relationship how do these factors influence the profitability. In general, the factors such as RSI, Volatility, Correlation, and others potential factors are “input”. The profit/loss is “output”. In our study, the term “Supervised classification” is used as the type of training process. After the training process, the ML algorithm can predict the profitability from the input at the time when the signal triggers.

Before elaborating the details, the task of machine learning is needed to be clarified carefully. From the data selection section, the lagging indicators were collected at the time when the trade position was being opened. When the spread crosses the zero, the trade position is closed, and the profit/loss is realized. If the trade wins, the profit is recorded as 1 and the loss is recorded as 0. According to the pairs trading strategy’s nature. The 2 trading positions are opened and closed at the same time. Therefore, 3 attributes are collected from one the trading result, profit/loss from leg 1, profit/loss from leg2 and total profit/loss. Profit_L1 plus Profit_L2 is Total_Profit. The machine learning algorithm aims to learn the input data which are lagging indicators and predict the profit/loss. Thus, the machine learning algorithms are design to predict profit/loss from leg 1, profit/loss from leg 2 and total profit/loss as stated before. predicting the total profit might not ensure it gives the better result. Predicting the individual leg of pairs trading might give better result. In conclusion, this thesis uses the lagging indicator data to predict the profit/loss from the trade by predicting individual leg and total leg. Figure 3-2 illustrates the outputs, the Profit_L1, Profit_L2 and Total_Profit, which machine learning algorithm predict.

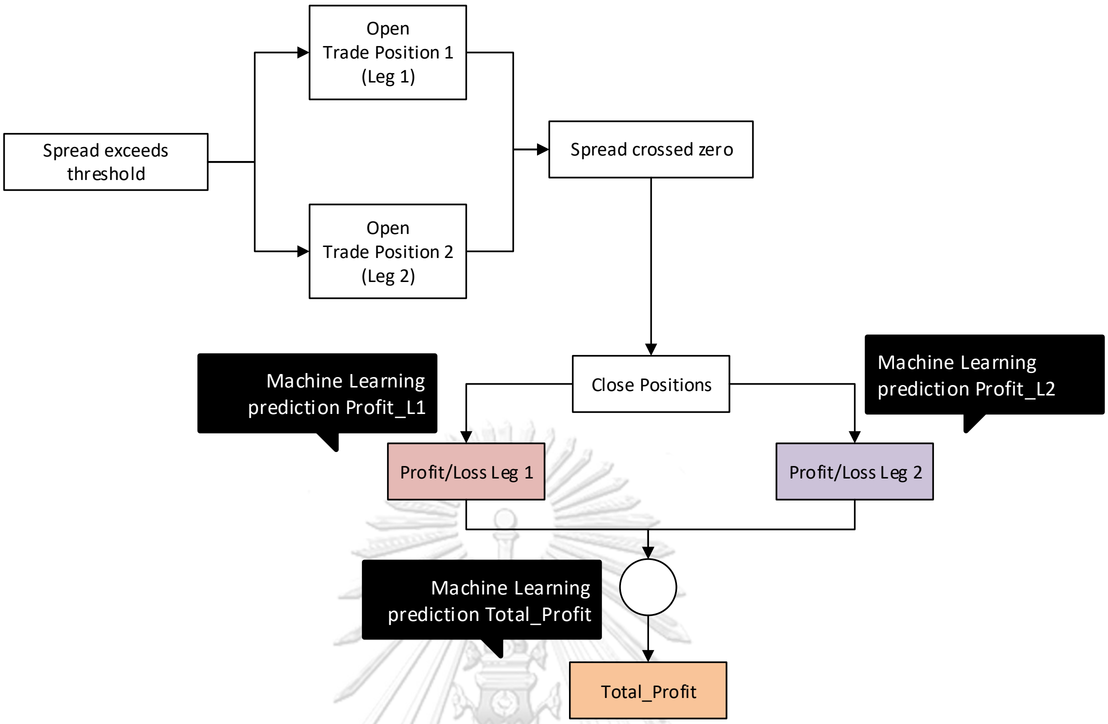

Figure 3-2 The identification of the outputs where machine learning algorithms predict

Artificial Neural Network, Logistic regression and XGBoost are applied to learn the data. This thesis aims to find the algorithms which can predict the outcome. Each algorithm has different distinctive points. Artificial Neural Network uses complex nodes and layers to construct the models and predicts the outcome, Logistic Regression uses simple Sigmoid equation and Log-loss to learn the data and XGBoost uses ensemble technique to create the model. The models with different techniques are studied used to learn the data. The expectation is that different models would yield the different performance.

Consecutive subsection explains the methods to create the machine learning model. Each model has different characteristic. The methods are slightly different but stay in similar scheme.

3.4.1 Feature selection

The feature selection is important process which hugely impacts the performance of the model. The process eliminates the irrelevant feature before used to train the model. It might eliminate some features which is anticipated that that they should have influenced the output. Figure 3-2 illustrate the process of feature selection.

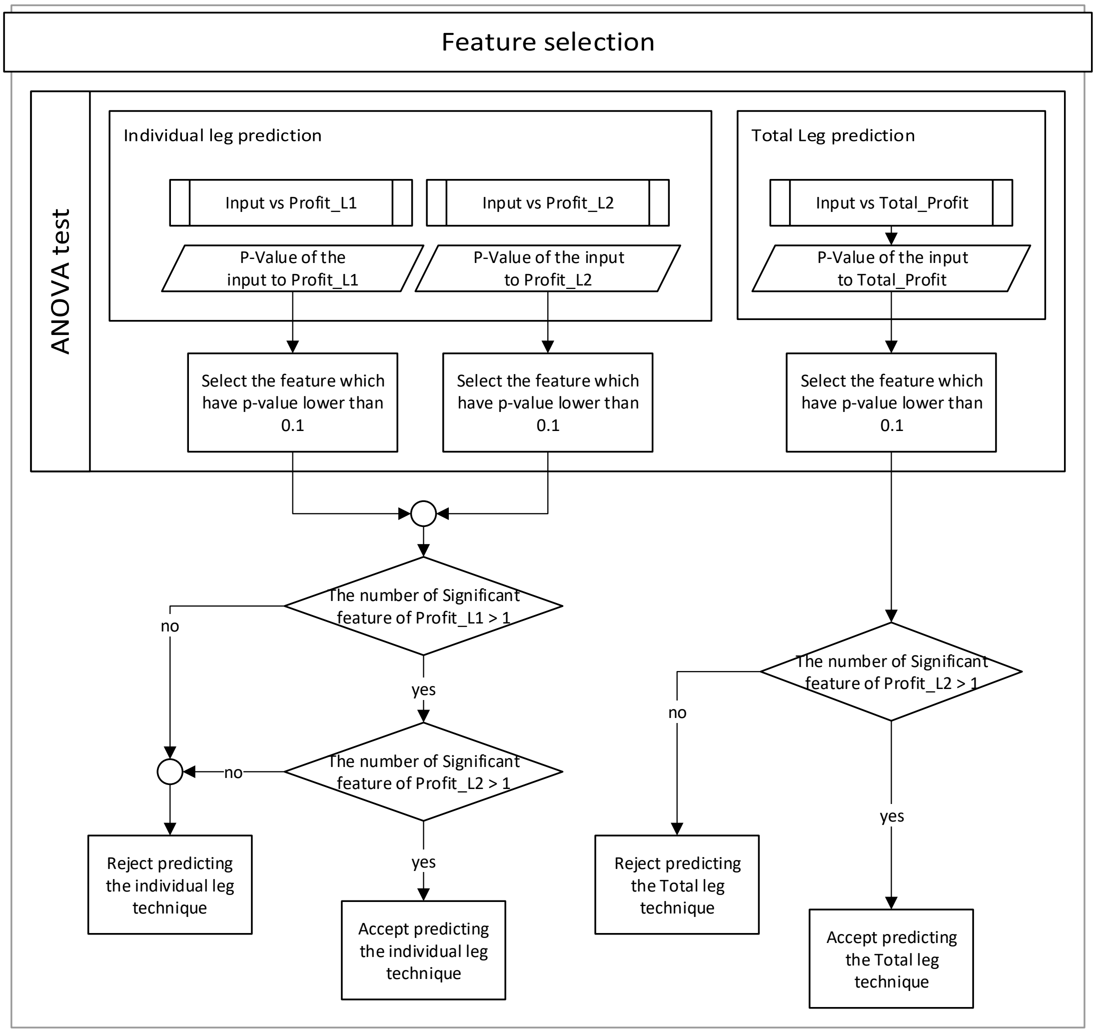

Figure 3-3 The process of feature selection

Prior to feature selection process, the output variables is separated into to 3 groups, Profit_L1, Profit_L1 and Total_Profit which are categorial variables. To be more meaningful, those 3 variables are binary classification consisting of 0 and 1, 0 and 1 mean loss and profit, respectively.

Thus, the features which might influence the profit and loss of the pairs trading strategy should be selected. ANOVA univariate test is implemented to select the features. The selected features are used to train and test the machine learning model.

ANOVA univariate test is used to measures the dependence of two variables. ANOVA assumes a linear relationship between the variables and the target, and the variables are normally distributed. ANOVA is compatible with continuous variables and binary target. The p-value of each feature from EURAUD vs EURCAD is calculated and ranked. The features which have p-value less than 0.1 are selected.

According to the Table 3-4, each outcome has different attributes which could influence the outcome regarding to ANOVA univariate test. For example, Profit_L1 of EURAUD-EURCAD, ANOVA univariate test shows that Correlation[20], VolatilityLeg1[0] and beta_10 have p-value less than 0.1 which could strongly influence the model. Therefore, only these features are selected to be used in the model and consider the other features as noise. Table 3-4,Table 3-5,Table 3-6 and Table 3-7 show the selected features from ANOVA univariate test in every selected pair.

Table 3-4 Summary of the features in EURAUDEURCAD which have p-value lower than 0.1

Pair	Profit_L1	
	feature	p-value
EURAUDEURCAD	Correlation[20]	0
	VolatilityLeg1[0]	0.06
	beta_10	0.1
	Profit_L2	
	feature	p-value
	VolatilityLeg2[0]	0.02
	VolatilityDiff	0.05
	Correlation[20]	0.1
	Total_Profit	
	feature	p-value
	Correlation[20]	0.01
	Type	0.01
	MomLeg2[0]	0.01
	MAShortValue_Leg2[1](5) -MALongValue_Leg2[1](10)	0.03
	MomLeg1[0]	0.03
	MAShortValue_Leg2[1] -MALongValue_Leg2[1]	0.06
	beta_10	0.07

Table 3-5 Summary of the features in EURAUDEURUSD which have p-value lower than 0.1

Pair	Profit_L1	
	feature	p-value
EURAUDEURUSD	MAShortValue_Leg1[1](50) -MALongValue_Leg1[1](100)	0.02
	MAShortValue_Leg2[1](50) -MALongValue_Leg2[1](100)	0.04
	beta_10	0.07
	Profit_L2	
	feature	p-value
	VolatilityDiff	0
	VolatilityLeg2[0]	0.01
	VolatilityLeg1[0]	0.01
	beta_10	0.1
	Total_Profit	
	feature	p-value
	VolatilityDiff	0.06
	VolatilityLeg2[0]	0.07

Table 3-6 Summary of the features in EURGBPEURCAD which have p-value lower than 0.1

Pair	Profit_L1	
	feature	p-value
EURGBPEURCAD		
	Profit_L2	
	feature	p-value
	VolatilityDiff	0
	VolatilityLeg2[0]	0.01
	VolatilityLeg1[0]	0.01
	beta_10	0.1
	Total_Profit	
	feature	p-value
	VolatilityDiff	0.06
	VolatilityLeg2[0]	0.07

Table 3-7 Summary of the features in EURGBPEURUSD which have p-value lower than 0.1

Pair	Profit_L1	
	feature	p-value
EURGBPEURUSD	Correlation[80]	0.09
	Profit_L2	
	feature	p-value
	MAShortValue_Leg1[1](50) -MALongValue_Leg1[1](100)	0.01
	Correlation	0.04
	Total_Profit	
	feature	p-value
	MAShortValue_Leg1[1](50) -MALongValue_Leg1[1](100)	0
	Correlation[40]	0.01
	VolatilityLeg2[0]	0.02
	MomLeg2[0]	0.04

The odd result from ANOVA univariate test shows that There is no feature which have p-value less than 0.1 in EURGBP vs EURCAD Profit_L1. In the other hand, any lagged features from EURGBP vs EURCAD does not significantly influence the outcome in ANOVA aspect. Thus, the prediction of the individual leg of EURGBP vs EURCAD is decided to be eliminated from Artificial Neural Network and Logistic

Regression, but XGBoost is still. The reason is that XGBoost does not require feature selection.

Another odd result is that only on feature is left in Total_Profit on EURAUDEURUSD, Profit_L2 on EURAUDEURUSD and Profit_L1 on EURGBPEURUSD. Therefore, the individual leg prediction on Artificial Neural Network and Logistic Regression for EURAUDEURUSD, EURGBPEURCAD and EURGBPEURUSD is not implemented.

The feature selection is applied to algorithms except XGBoost because the feature selection is built-in in XGBoost algorithm.

The feature selection eliminates a lot of features in each pair. If the feature selection eliminates the features and left 0 or only 1 feature, these pairs are ignored and not trained in machine learning algorithm. For example, in EURGBP vs EURCAD on Profit_L1, predicting the individual leg technique is ignored which means that if Profit_L1 is not capable to be predicted, Profit_L2 is not predicted as well and vis versa. Conversely, the elimination from feature selection does not impact XGBoost because feature selection is not necessary for XGBoost. Thus, every pair and both technique of prediction are still valid for XGBoost which is the strength of this algorithm. Table 3-8 shows the summary of the pairs which is valid after the feature selection.

Table 3-8 Summary of the valid pairs, algorithms and techniques after feature selection

	EURAUDEURCAD		EURAUDEURUSD		EURGBPEURCAD		EURGBPEURUSD	
	Individual leg	Total leg	Individual leg	Total leg	Individual leg	Total leg	Individual leg	Total leg
Artificial Neural Network	Yes	Yes	No	No	No	Yes	No	Yes
Logistic regression	Yes	Yes	No	No	No	Yes	No	Yes
XGBoost	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

3.4.2 Standardization

Differences in the scales across input variables may increase the difficulty of the problem being modeled. Standardization is another scaling technique where the values are centered around the mean with a unit standard deviation. This means that the mean of the attribute becomes zero and the resultant distribution has a unit standard deviation by following equation which is similar to z-score calculation.

$$z = \frac{x_i - \mu}{\sigma}$$

Where **z** **; standardized value**

x_i **; Attribute in the feature**

μ **; Average**

σ **; Standard deviation**

After the features were standardized, the algorithm yields the average (μ) and standard deviation (σ). Table 3-9 and Table 3-10 show that value of the average (μ) and standard deviation (σ) each feature. The standardization is implemented only in Artificial Neural Network.

Table 3-9 Summary of mean of features for standardization

features	EURAUDEURCAD	EURAUDEURUSD	EURGBPEURCAD	EURGBPEURUSD
	mean	mean	mean	mean
MomLeg1[0]	49.0242	52.3546	51.7974	52.5021
MomLeg2[0]	53.0122	48.0207	48.1612	48.7265
VolatilityLeg1[0]	0.00123180	0.00129220	0.00091588	0.00087028
VolatilityLeg2[0]	0.00099731	0.00116604	0.00106793	0.00089734
VolatilityDiff	0.00222911	0.00245825	0.00198380	0.00176762
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	-0.00003570	0.00019529	0.00005080	0.00001550
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.00030210	-0.00053646	-0.00020364	-0.00007000
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.00013748	0.00033216	0.00010174	0.00006180
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.00036493	-0.00012471	-0.00001880	-0.00007510
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.00021861	-0.00000855	0.00007010	0.00008210
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.00018696	-0.00010224	-0.00011342	-0.00001950
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.00020450	-0.00020085	-0.00007080	-0.00021610
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.00022726	-0.00025273	-0.00002950	-0.00040558
beta_10	-0.11962000	-0.11019000	0.15979800	0.19436800
beta_20	-0.04600730	-0.08235650	0.20874800	0.25410900
beta_50	-0.01251430	-0.07019570	0.34734400	0.38647300
beta_100	0.00685351	-0.03008760	0.39561500	0.43392000
Correlation	0.04177710	-0.28115500	0.02060120	0.03179560
Correlation[20]	0.13625700	-0.20684700	0.15152800	0.18199900
Correlation[40]	0.34950900	0.04682970	0.30285700	0.36191800
Correlation[80]	0.48893500	0.16808900	0.43469800	0.49675700
Correlation[100]	0.49162000	0.20334300	0.44535800	0.49647400
Type	0.44680900	0.55900600	0.55932200	0.54961800

Table 3-10 Summary of variance of features for standardization

features	EURAUDEURCAD	EURAUDEURUSD	EURGBPEURCAD	EURGBPEURUSD
	variance	variance	variance	variance
MomLeg1[0]	316.0580	387.4060	347.9510	363.0770
MomLeg2[0]	359.6670	445.6060	345.8470	313.2360
VolatilityLeg1[0]	0.00000024	0.00000027	0.00000013	0.00000012
VolatilityLeg2[0]	0.00000015	0.00000028	0.00000021	0.00000019
VolatilityDiff	0.00000045	0.00000078	0.00000044	0.00000042
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.00000570	0.00000654	0.00000104	0.00000092
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.00000592	0.00000815	0.00000394	0.00000305

MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.00000283	0.00000490	0.00000093	0.00000077
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.00000382	0.00000348	0.00000259	0.00000147
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.00000608	0.00000797	0.00000176	0.00000144
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.00000543	0.00000648	0.00000651	0.00000499
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.00002260	0.00002660	0.00000485	0.00000410
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.00002130	0.00002330	0.00002040	0.00001290
beta_10	0.20194600	0.13433100	0.31571800	0.43340500
beta_20	0.09461660	0.05972360	0.14139500	0.19503200
beta_50	0.02549440	0.02975440	0.06024700	0.07123340
beta_100	0.02191280	0.01688380	0.05497480	0.06154340
Correlation	0.26944800	0.24915400	0.21470100	0.28638600
Correlation[20]	0.20930400	0.19479400	0.20642700	0.19973600
Correlation[40]	0.16271900	0.23504500	0.23934900	0.20308900
Correlation[80]	0.13050600	0.24592300	0.19731200	0.16319600
Correlation[100]	0.14556700	0.22398100	0.19461800	0.15968900
Type	0.24717100	0.24651800	0.24648100	0.24753800

3.4.3 Hyper-parameter tuning

There are many complex mathematical equations and parameters work behind the machine algorithm which used to optimize the model itself to learn the training data. For practitioners, they need to search for the optimal and suitable parameters to construct the proper ANN model for the problems so-call hyper-parameter turning.

Artificial Neural Network

ANN is one of the well-known and powerful machine learning algorithms which is used for classification, regression, and clustering. ANN consists of different layers such as input layer, hidden layers, and output layer. Each of the layer organizes the units called neurons. The data from the outside world firstly go to input layer and processed by each connecting neuron which mimic the human brain, then the predicted answers to the output layer.

To maximize the performance of the Artificial Neural Network, the algorithm is needed to be fine-tuned. The hyper-parameters needed to be tuned are explained below.

The number of node and layer, Artificial neural networks have two main hyper-parameters that control the architecture or topology of the network: the number of layers and the number of nodes in each hidden layer. You must specify values for these parameters when configuring your network. A single-layer neural network can only be used to represent linearly separable functions. This means very simple problems where, say, the two classes in a classification problem can be neatly separated by a line. If your problem is relatively simple, perhaps a single layer network would be enough. In the thesis, the problem cannot be justified whether the relationship of the data is simple or complex. Therefore, the possible architecture of Artificial neural network is systematically created and searched for the best model.

Most problems that are interesting to be solved are not linearly separable. A Multilayer Perceptron can be used to represent convex regions. This means that in effect, they can learn to draw shapes around examples in some high-dimensional space that can separate and classify them, overcoming the limitation of linear separability.

According to [16], the number of nodes can be equal to the number of input features and the number of layers is set to be single layer. The reason is to prevent the overfitting. Setting up the number of node equal to the number of input features and the number of layer equal to one are too strictly. Thus, the multilayer neural network is experimented with different number of nodes. The number of node equal which is equal to the number of input feature is set as a baseline and the number of nodes is doubled as another structure.

For the number of layers, single layer is also set as the baseline and start increase the number of layers. If the number of node and layer are overwhelmed, the model may cause overfitting. 3 layers with double-from-input-feature node is adequate for the problem.

Optimizers are algorithms or methods used to change the attributes of your neural network such as weights and learning rate to reduce the losses. Optimization algorithms or strategies are responsible for reducing the losses and to provide the most accurate results possible. According to [17], Adaptive Moment Estimation, or Adam, is the high performance optimizer which work perfectly for Artificial Neural Network such as Straightforward to implement, Computationally efficient and Little memory requirements. Therefore, ADAM optimizer is used in the model.

Activation function, in hidden layer Rectified Linear Unit, or ReLu is used in the models. According to [2],linear units are nearly linear, they preserve many of the properties that make linear models easy to optimize with gradient-based methods. They also preserve many of the properties that make linear models generalize well. For output layer, Sigmoid activator is used in the models for classification problem.

Regularization, according to the layers and nodes, If the model configuration consists of abundant nodes and layers, it could cause overfitting problem. A single model can be used to simulate having many different network architectures by randomly dropping out nodes during training. This is called dropout and offers a very computationally cheap and remarkably effective regularization method to reduce overfitting and improve generalization error in deep neural networks of all kinds. Therefore, the model is optimized poorly. Regularization helps prevent the overfitting. In this thesis, “Dropout regularization” is implemented. According to [18], dropout rate is simply set to 0.5 which is nearly optimal for valid networks.

The set of hyper-parameters of Artificial Neural Network is summarized in Table 3-11. These combinations of hyper-parameters are tested in next section.

Table 3-11 Set of hyper-parameters to tune Artificial Neural Network

Artificial Neural Network	
Attribute	range
Layer(s)	{1,2,3}
Node(s)	{input_features x1, input_features x2, input_features x3}
Optimizer	Adam
Activation function	ReLu for hidden layers, Sigmoid for output layer
Dropout rate	{0.5}

Logistic regression

Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. The nature of target or dependent variable is dichotomous, which means there would be only two possible classes. Regularization is extremely important in logistic regression modeling. Without regularization, the asymptotic nature of logistic regression would keep driving loss towards 0 in high dimensions. Consequently, most logistic regression models use one of the following two strategies to dampen model complexity, L1 and L2 regularization

There are 3 hyper-parameters to be tuned up.

Regularization – The most common regularization consists of L1 and L2. Therefore, the best performance between L1 and L2 is investigated.

C parameter - Inverse regularization parameter - A control variable that retains strength modification of Regularization by being inversely positioned to the Lambda regulator. For the value of c parameter, the value is varied from 0.001 to 1000 exponentially.

Solver - Algorithm to use in the optimization problem. The 'newton-cg', 'sag', and 'lbfgs' solvers support only L2 regularization with primal formulation, or no regularization. The 'liblinear' solver supports both L1 and L2 regularization, with a dual formulation only for the L2 penalty. The Elastic-Net regularization is only supported by the 'saga' solver.

The set of hyper-parameters of Artificial Neural Network is summarized in Table 3-12. These combinations of hyper-parameters are tested in next section.

Table 3-12 Set of hyper-parameter to tune Logistic Regression

Logistic Regression	
Attribute	range
Regularization	{none, L1, L2}
C parameter	{0.001,0.01,0.1,1,10,100,1000}
Solver	{'liblinear' for L1 and L2 regularization,'lbfgs' for non-regularization}

XGBoost

XGBoost is a distributed gradient boosting library intended to be exceptionally proficient, adaptable, and minimized. It actualizes AI calculations under the Gradient Boosting system. XGBoost gives a parallel tree boosting that tackle scientific issues in a quick and precise. There are some parameters needed to be tuned to maximize the model performance.

booster – In this thesis gbtree-model is applied

eta or learning rate - Step size shrinkage utilized in update to protect overfitting. After each boosting step, we can straightforwardly get the weight of new features, and eta shrinks the feature weights to create the boosting procedure more stable.

min_child_weight - The minimum sum of instance weight required in a child. If the tree partition step shows in a leaf node with the sum of instance weight lower than min_child_weight, the building process will surrender further dividing.

min_child_weight is implemented to prevent over-fitting. Higher values protect a model from learning relations which cloud be exceptionally explicit to the specific example chose for a tree. In the other hand, higher values can generate under-fitting model.

max_depth [default=6] max_depth is implemented to prevent over-fitting as higher depth is grant model to learn relations very exceptionally explicit to a sample.

gamma [default=0] A node is split only when the consecutive split results a positive reduction in the loss function. Gamma controls the minimum loss reduction required to create the other split. Gammas create the algorithm conservatively. The values can change according to the loss function

subsample [default=1] Subsample proportion of the preparation cases. Setting it to 0.5 implies that XGBoost would randomly test half of the training data before developing trees. Subsample prevents overfitting.

colsample_bytree [default=1] is the subsample ratio of columns when creating each tree. Subsampling happens once for each tree developed.

objective - Logistic regression for binary classification uses 'binary:logistic' , returns predicted probability

The set of hyper-parameters of XGBoost is summarized in Table 3-13. These combinations of hyper-parameters are tested in next section.

Table 3-13 Set of hyper-parameter to tune XGBoost

XGBoost	
Attribute	range
Booster	{gbtree}
Eta	{0.01,0.05,0.1,0.15,0.2}
min_child_weight	{1,2,3,4,5,6,7,8,9}
Max_depth	{4,5,6,7,8,9}
Gamma	{0.0, 0.01, 0.001, 0.2, 0.002}
subsample	{0.6, 0.65, 0.7, 0.75, 0.8,1}
colsample_bytree	{0.6, 0.65, 0.7, 0.75, 0.8,1}
objective	binary:logistic

3.4.4 GridSearchCV

After the parameters in the model are defined, the value of the parameter is needed to be tuned to maximize the model performance. In the study, GridSearchCV is implemented. GridSearchCV is a library function that is a member of sklearn's model selection package. It helps to loop through predefined hyper-parameters and fit your estimator (model) on your training set. So, in the end, you can select the best parameters from the listed hyper-parameters.

In every set of hyper-parameters is tested by k-Fold cross validation illustrated in Figure 3-4 to measure the performance of the model. K-Fold cross validation divides the training data into k parts, folds 1 part as test data and use the others k-1 as training data which means that the k-Fold validation can generate k different test sets from single data set. This method can obtain total k score matrix which is enough to evaluate the model performance

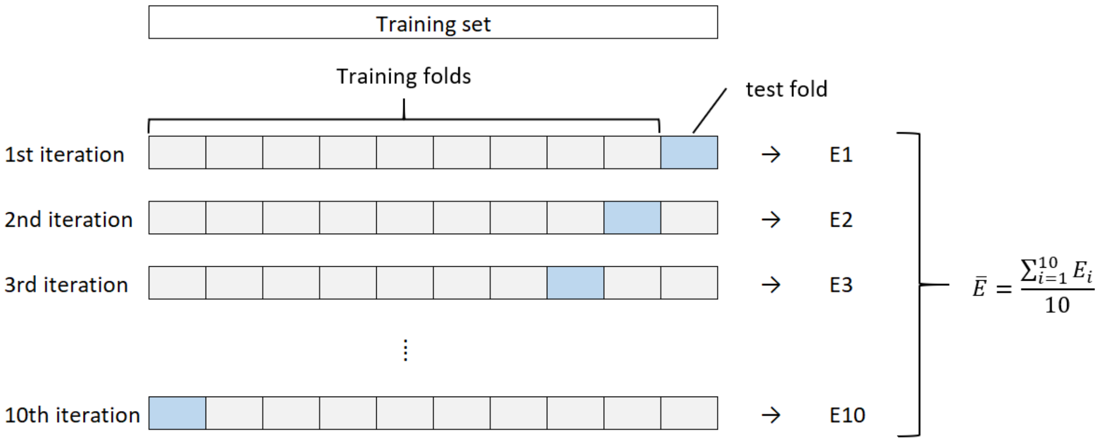

Figure 3-4 Demonstration of k-Fold Cross Validation

Every single set of hyper-parameters provide the different outcome. The performance of the set of Hyper-parameters can be evaluated by the performance matrix. The important matrixes are accuracy, precision, recall and F1 score. The k-Fold validation is applied for every set of hyper-parameters, taking the average of k iterations, repeating the process until the matrix score is maximized. In practical, the best set of hyper-parameters cannot be justified, but the set of hyper-parameters which yields maximum f1 score is aimed to be found

We use the In-sample data from previous section to train the machine learning model and evaluate the performance from score matrix by k-Fold cross validation. We simply set k=5 which is enough to generalize the model [19]

The score which can be used to evaluate the performance of the algorithm is F1-score. Due to our study, machine learning algorithm is used to predict the output whether the signal is profit or not. If the algorithm predicts that the outcome is 1 (0), it means the signal is profitable (not profitable). The confusion matrix is constructed and calculated the evaluation scores which is accuracy, precision, f1 score and recall.

In this thesis, ML models can predict whether the trading signals are profitable or not. F1-score is implemented when the False Positives and False Negatives are important. F1 score is the weighted average of recall and precision. In recall aspect, if the recall is low, it means that the algorithm fails to catch the profitable signal. Therefore, the machine learning model will reject a lot of signal. Traders would miss the opportunity to make profit. In precision aspect, if the precision is low, it means that the algorithm accepts a lot of false signals, causing huge losses. Thus, maximum f1 score help balance the advantages of recall and precision.

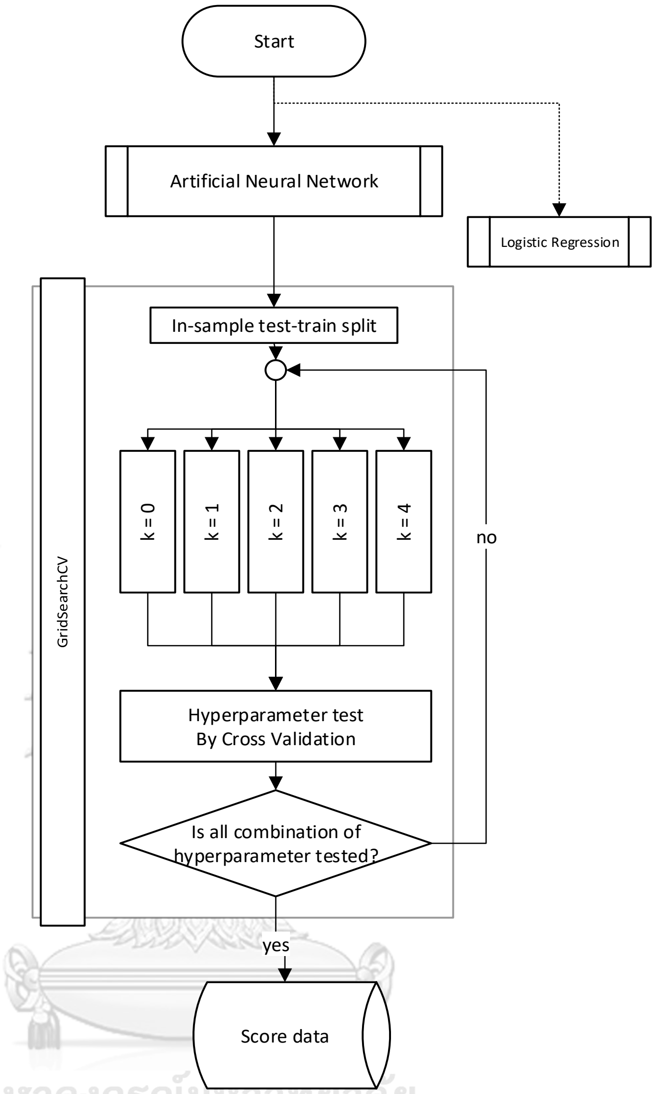

Figure 3-5 The process of GridSearchCV on Artificial Neural Network and Logistic Regression

For Artificial Neural Network and Logistic regression, the process of GridSearchCV illustrated in Figure 3-5 is simple. The process splits the data into 5 groups due to k=5 in the setup. The combination of hyper-parameters is tested by cross validation process. Then, the process is looped until all combinations is tested. Finally, every set of hyper-parameters tested and looped in GridSearchCV. Then, these data are used in next section to select the best model configuration.

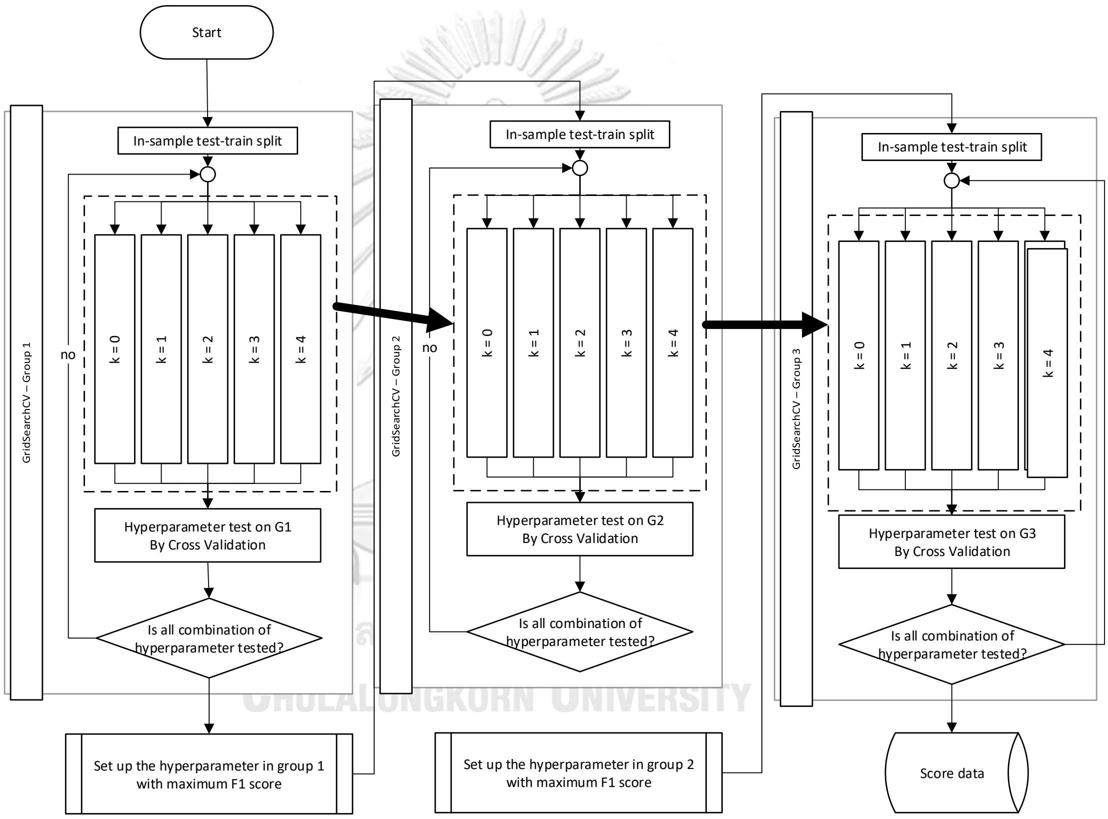

Figure 3-6 The process of GridSearchCV on XGBoost

For XGBoost, the process is similar to the Artificial Neural Network and Logistic Regression. The XGBoost algorithm does not require standardization and feature selection. All features are capable to be used as inputs in the training process. For Hyper-parameter tuning, the valid sets of the Hyper-parameter for XGBoost are more than 3,000 combinations. GridSearchCV is applied, it will consume a lot of time. A set of hyper-parameters consumes approximately 10 minutes, the whole algorithm takes almost a month. Therefore, GridSearchCV is not implemented directly to all combination of hyper-parameters in XGBoost. The process of tuning the XGBoost hyper-parameters is illustrated in Figure 3-6. The Hyper-parameter is tuned set by set. Some of the Hyper-parameters are grouped and tuned by GridSearchCV to maximize the performance of the model. Then, these optimized Hyper-parameters is used to tune the consecutive set of Hyper-parameters. The technique shrinks a lot of time in GridSearchCV. The trade-off of the technique is that the performance of the model might not achieve the maximum performance due to the ignorance of some set of Hyper-parameters. However, the result from maintaining the whole set of Hyper-parameters is not significantly different from this technique. Therefore, this technique is used to tune the model. The groups the hyper-parameter is shown in Table 3-14.

Table 3-14 Groups of hyper-parameters used to tune XGBoost

	Attribute	range
Group 1	learning_rate	{0.01,0.05,0.1,0.15,0.2}
	min_child_weight	{1,2,3,4,5,6,7,8,9}
	Max_depth	{4,5,6,7,8,9}
Group 2	subsample	{0.6, 0.65, 0.7, 0.75, 0.8,1}
	colsample_bytree	{0.6, 0.65, 0.7, 0.75, 0.8,1}
Group 3	gamma	{0.0, 0.01, 0.001, 0.2, 0.002}

The whole step of this technique in EURAUD vs EURCAD pair is demonstrated in this section. Figure 3-7 graphically shows the step of XGBoost Hyper-parameter tuning.

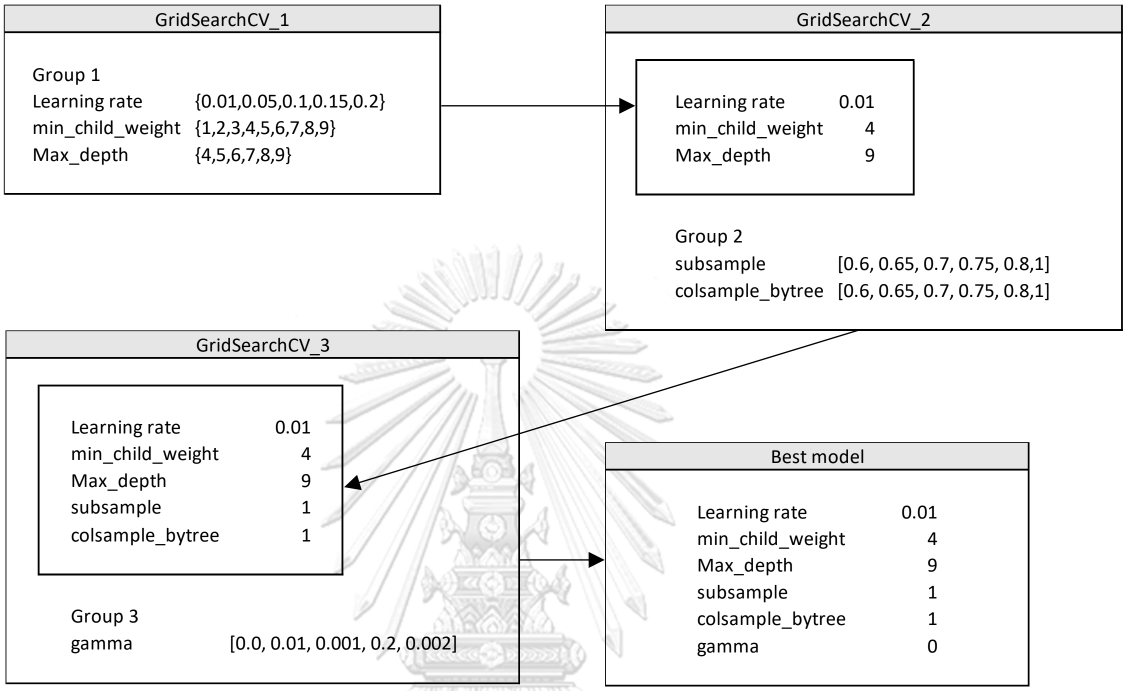

Figure 3-7 The result of GridSearchCV to tune XGBoost in groups

First set of the Hyper-parameter consists of learning_rate, max_depth and min_child_weight. The pair of EURAUD vs EURCAD by predicting Total_Profit is exemplified in this section. GridSearchCV is applied to the valid Hyper-parameters by maximizing the f1 score. The result of first step is learning_rate = 0.01, max_depth = 4 and min_child_weight = 9 with average f1 test score = 78.38%, the f1 score is quite satisfied in first round of tuning. According to the result, there is not necessary to fine-tune the whole combination of Hyper-parameter to achieve the maximum score which could be only 1-5% higher but consumes a lot of time.

Next, these results are used as fixed values to tune the next combination, colsample_bytree and subsample. The result from GridSearchCV is colsample_bytree = 0.6 and subsample = 0.6 The average test f1 score in second round is 78.96%.

Last set of hyper-parameters is tuned by GridSearchCV as well. GridSearchCV shows the best gamma is 0.0 which the average test f1 score last round is 78.96%.

Thus, the model configuration of predicting the profitability of EURAUD-EURCAD by total profit feature is derived. The same rules are applied in the others as same as stated before.

3.5 MODEL SELECTION AND EVALUATION

After optimizing the parameter in GridSearchCV is finished, the model selection and evaluation are clarified in this section. The maximum average score is selected and used to predict the out-of-sample data. Before selecting the hyper-parameter, the result from GridSearchCV is analyzed thoroughly.

The trade result is separated into 3 groups, Profit_L1, Profit_L2 and Total_Profit. The result from individual prediction and total prediction is expected that they would win each other. However, the result from GridSearchCV shows that there is not significantly different. Thus, both prediction style should be studied further.

To select the set of hyper-parameters, the best average F1-score from report of GridSearchCV is simply selected. The maximum F1-score is assumed to represent the most generalized model. Therefore, the set of the hyper-parameter could perform the best performance in out-of-sample data.

Finally, the best F1-score on each exchange rate pair is selected. It implies that there is one algorithm which is suitable and performs the best. This algorithm is used in the out-of-sample test.

Artificial Neural Network

For Artificial Neural Network, the results seem straightforward, the GridSearchCV is run on the combination of hyper-parameter stated in section 3.4.3. The model with the highest average f1 test score is selected. Table 3-15 shows the best f1 score from GridSearchCV.

Table 3-15 Best combination of hyper-parameter of Artificial Neural Network

	EURAUDEURCAD	EURAUDEURCAD	EURAUDEURCAD	EURGBPEURCAD	EURGBPEURUSD
output	Profit_L1	Profit_L2	Total_Profit	Total_Profit	Total_Profit
Layer	1	3	3	1	3
Node	2	1	1	1	1
matrix	'test_f1'	'test_f1'	'test_f1'	'test_f1'	'test_f1'
k1	0.706	0.682	0.852	0.846	0.683
k2	0.813	0.698	0.668	0.727	0.667
k3	0.765	0.667	0.802	0.7	0.667
k4	0.645	0.667	0.802	0.72	0.733
k5	0.667	0.667	0.828	0.636	0.7
average f1 score	0.719	0.676	0.790	0.726	0.69

Logistic Regression

Some results of the Logistic Regression are odd. For example, In the model with C=0.001 and L2 regularization in EURAUD vs EURCAD predicting Profit_L1 which result is stated in Table 3-16 giving average 100% of recall score, meaning the model doesn't have ability to classify the data. Although these model parameters give the highest average f1 score, these parameters are eliminated. This type of the hyper-parameter is anticipated to be underfitting.

Table 3-16 The odd result from cross validation of the set of hyper-parameters in Logistic Regression with 100% average recall

		C=0.001 Regularization = L1
Total	'split0_test_recall':	1
	'split1_test_recall':	1
	'split2_test_recall':	1
	'split3_test_recall':	1
	'split4_test_recall':	1
	'mean_test_recall':	1
	'split0_train_recall':	1
	'split1_train_recall':	1
	'split2_train_recall':	1
	'split3_train_recall':	1
	'split4_train_recall':	1
	'mean_train_recall':	1

The solution is to select the consecutive highest Hyper-parameter which does not give 100% recall. According to GridSearchCV result, the best hyper-parameter which does not contain 100% average test recall. Therefore, the next model Hyper-parameter which have max f1 score is C=0.001 and L2 regularization. The average f1 score from cross validation 78.55%. The rules mentioned before are applied through all pairs. Table 3-17 summarized the best f1 score of hyper-parameters from GridSearchCV.

Table 3-17 Best combination of hyper-parameter of Logistic Regression

	EURAUDEURCAD	EURAUDEURCAD	EURAUDEURCAD	EURGBPEURCAD	EURGBPEURUSD
output	Profit_L1	Profit_L2	Total_Profit	Total_Profit	Total_Profit
C	1	0.1	1	1000	1000
Regularization	L2	L2	L2	L1	L1
k1	0.688	0.682	0.837	0.737	0.62
k2	0.774	0.7	0.837	0.737	0.73
k3	0.727	0.667	0.7	0.706	0.589
k4	0.706	0.667	0.773	0.75	0.739
k5	0.625	0.667	0.78	0.745	0.623
average f1 score	0.704	0.676	0.786	0.735	0.67

XGBoost

For XGBoost, the process in GridSearchCV completely selected the best set of hyper-parameters. Table 3-18 summarized the best hyper-parameter of XGBoost.

Table 3-18 Best combination of hyper-parameter of XGBoost

	EURAUDEURCAD			EURAUDEURUSD			EURGBPEURCAD			EURGBPEURUSD		
	L1	L2	Total	L1	L2	Total	L1	L2	Total	L1	L2	Total
learning_rate	0.01	0.001	0.01	0.15	0.01	0.01	0.01	0.15	0.2	0.01	0.01	0.01
max_depth	4	4	4	4	4	4	8	7	9	4	6	4
min_child_weight	3	7	9	5	9	9	2	2	1	2	1	8
subsample	1	1	1	0.75	0.6	0.6	1	1	1	1	1	1
colsample_bytree	1	0.6	1	0.7	0.6	0.6	0.6	0.8	1	0.7	1	1
gamma	0	0	0	0	0	0	0	0.2	0.2	0.1	0	0
average f1 score	0.634	0.597	0.774	0.542	0.807	0.794	0.729	0.633	0.662	0.653	0.631	0.635

When the F1-scores were gathered, the only one algorithm and technique is selected on each pair to select the best model to trade in out-of-sample data. The model is needed to be thoroughly picked. Table 3-19 summarized the F1-score.

Table 3-19 Summary of F1-score from algorithms on every pair

		ANN	LR	XGB
EURAUDEURCAD	Profit_L1	0.719	0.7039697	0.634
	Profit_L2	0.676	0.6763636	0.597
	Total_Profit	0.79	0.7855267	0.77
EURAUDEURUSD	Profit_L1			0.542
	Profit_L2			0.807
	Total_Profit			0.794
EURGBPEURCAD	Profit_L1			0.729
	Profit_L2			0.633
	Total_Profit	0.726	0.7349329	0.662
EURGBPEURUSD	Profit_L1			0.653
	Profit_L2			0.631
	Total_Profit	0.69	0.67	0.635

EURAUD-EURCAD

The F1-scores from predicting the Total_Profit beat the F1-scores from Profit_L1 and Profit_L2. Therefore, the method that is selected to predict the profitability is Total_Profit method. Next, the best decision is to select the model which yields the highest F1-score. The model is expected to perform the prediction on out-of-sample data similarly to in-sample data. The Artificial Neural Network gives the highest F1-score. Thus, the Artificial Neural Network is selected to predict the profitability of EURAUD-EURCAD

EURAUD-EURUSD

Only XGBoost is capable to predict the profitability of EURAUD-EURUSD due to feature selection. Therefore, either individual prediction or total prediction is selected. The highest F1-score is Profit_L2. However, other leg, Profit_L1 prediction yields only 0.542. Therefore, the performance of prediction might be dropped. For total prediction, The F1-score is 0.794 which is very satisfied. Thus, for EURAUD-EURUSD, XGBoost prediction total profit is selected.

EURGBP-EURAUD

The performance of these algorithms is similar to EURAUD-EURUSD. The highest F1-score is Logistic Regression for total prediction. Therefore, Logistic Regression predicting total profit is selected.

EURGBP-EURUSD

The performance of these algorithms is similar to EURAUD-EURUSD and EURGBP-EURAUD. The highest F1-score is Artificial Neural Network for total prediction. Therefore, Artificial Neural Network predicting total profit is selected.

3.4 OUT-OF-SAMPLE TEST

After training the model is finished, the model architecture and model parameter of ML algorithm are extracted. Then, the out-of-sample data is tested from January 2015 to December 2018

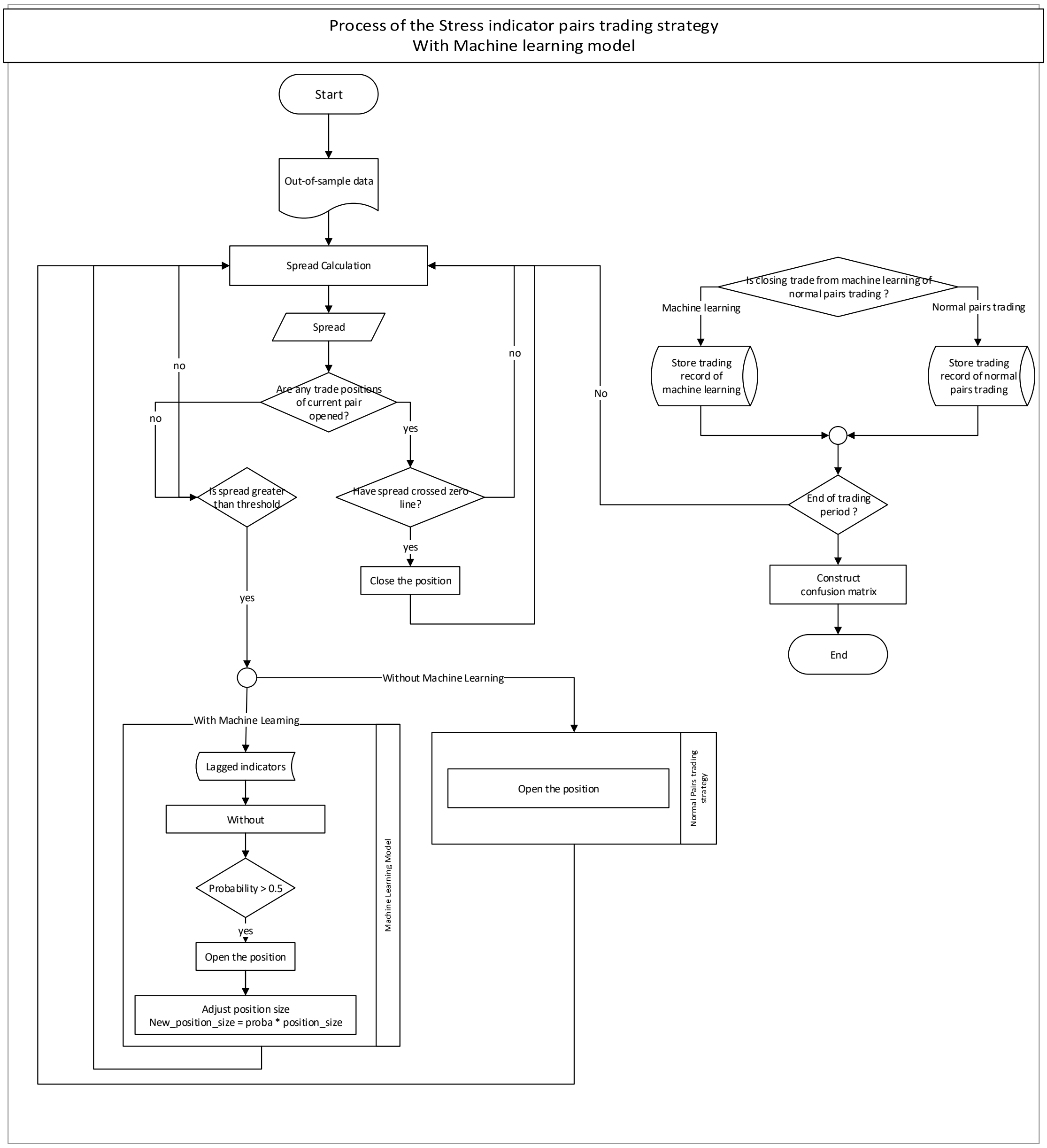

Figure 3-8 The process of the stress indicator pairs trading strategy with machine learning model

Machine Learning algorithm implementation

ML algorithm is implemented to trade from January 2015 to December 2018. The step of implementation shown in Figure 3-8 is slightly different from normal pairs trading strategy. First step, machine learning algorithm is traded and evaluated by the performance of prediction in out-of-sample data by confusion matrix. The lagging indicators from the signal is put in the machine learning algorithm. Then, the algorithm predicts the profitability in terms of probability. If the probability is greater than 0.5, it means the signal has potential to make money and vis versa. All the machine learning algorithm is tested and collected the performance in terms of confusion matrix.

For example, from EURAUD-EURCAD, the Artificial Neural Network require 7 features, Correlation[20], Type, MomLeg2[0], MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10), MomLeg1[0], MAShortValue_Leg2[1]-MALongValue_Leg2[1] and beta_10 to predict the profitability of Total_Profit. At the moment when the signal triggers, these 7 features are standardized and put in the Artificial Neural Network model created in section 3.5. The model calculates the probability from 0 to 1. If the probability is greater than 0.5, the trade position is open and vis versa. Then, performance of the predicting model is evaluated by comparing to the normal pairs trading strategy. The normal pairs trading strategy trades all of signal but the machine learning algorithms filter some signals. Therefore, the confusion matrix can evaluate the performance that the negative signals are actual profit or loss and also the positive signals are profit or loss as well.

Table 3-20 shows the summary of the confusion matrix of the out-of-sample test data. The results show that the machine leaning algorithms work quite well on predict the profitability.

Table 3-20 Summary of confusion matrix from out-of-sample data

Confusion Matrix	EURAUD EURCAD		EURAUD EURUSD		EURGBP EURCAD		EURGBP EURUSD	
Artificial Neural Network	15	27					12	53
	13	42					16	82
Logistic Regression					36	22		
					55	36		
XGBoost			9	31				
			18	55				

Table 3-21 summarized the accuracy, precision, recall and F1 score. Most of the algorithm predicts with more than 50% accuracy and 50% f1 score which is close to the scores in k-fold Cross Validation. However, only confusion matrix cannot exactly tell the actual performance. The total profit in cash from trading system is the real performance. Next section, profit/loss and cumulative profit are studied further to justify whether the machine learning can perform better than solely pairs trading strategy.

Table 3-21 Summary of scores from out-of-sample data

		EURAUD EURCAD	EURAUD EURUSD	EURGBP EURCAD	EURGBP EURUSD
Artificial Neural Network	Accuracy	0.587628866			0.576687117
	Precision	0.582089552			0.59375
	Recall	0.709090909			0.969387755
	F1	0.639344262			0.736434109
Logistic Regression	Accuracy			0.483221477	
	Precision			0.620689655	
	Recall			0.395604396	
	F1			0.483221477	
XGBoost	Accuracy		0.566371681		
	Precision		0.887096774		
	Recall		0.753424658		
	F1		0.814814815		

4 PROFIT EVALUATION

When the signal triggers the strategy to open the position, the ML algorithm predicts the signal and provides the probability of the profitability from 0 to 1. If the probability is less than 0.5, the strategy will not open the position. Otherwise, the strategy will adjust the position size by multiplying the probability to position size to reduce the risk exposure based on probability.

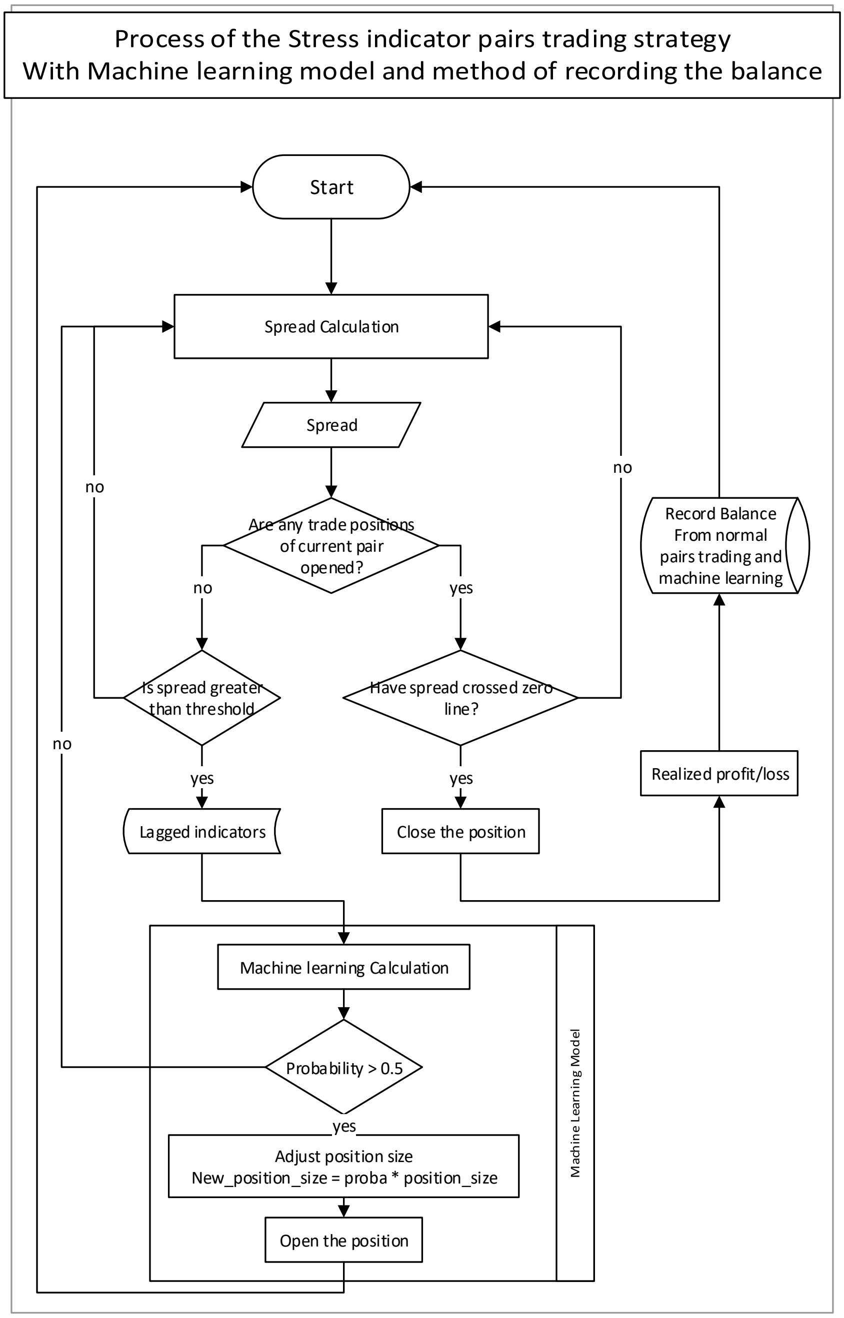

Figure 4-1 The process of pairs trading strategy with machine learning recorded in cumulative profit

For example, EURAUD vs EURCAD with Total_Profit outcome is being predicted by Artificial neural Network. The first signal from starting point is triggered by spread = 33. At the time the trade is opening. The ANN requires 7 parameters regarding to the model tuning in section 3.5. Next, 7 required parameters are standardized. The ANN model will predict the profitability in terms of probability.

This section does not evaluate the correctness of classification. This section aims to evaluate the profit/loss in cash whether the balance of the portfolio is improved after implementing the machine learning algorithms. The process is similar to out-of-sample testing. The difference is that this section records the outcome in cumulative profit or balance curve. The process is illustrated in Figure 4-1

The rule to trade with the strategy and ML filter together is that when the signal trigger the strategy, there are 2 valid situations.

- If the ML filter predicts the probability is equal or less than 0.5, the strategy will not open the trade. Next, the strategy will wait until the spread across the zero line and spread diverges more than 2σ again (wait the next cycle) then the signal can trigger the strategy again.
- If the ML filter predicts the probability is greater than 0.5, the strategy opens the position and adjust the lot size of each leg by multiplying the lot size by the probability from ML algorithm.

Figure 4-1 shows the trading result by predicting Total_Profit of the EURAUD vs EURCAD using Artificial Neural Network on out-of-sample data. For simplicity, the profit is proportionated by probability, In the other words, the profit is multiplied by the probability from Machine Learning algorithm to investigate the effect of prediction, not necessary to run the trade again.

Table 4-1 shows the trading records using machine learning versus not using machine learning

Table 4-1 Trading record of pairs trading strategy with and without machine learning

	Profit	y_pred	y_filter	Profit_w_ML	Cumulative_profit	Cumulative_profit With ML
0	-38.27	0.999998	0.999998	-38.2699235	-38.27	-38.26992346
1	380.1	1	1	380.1	341.83	-38.26992346
2	501.48	0.477751	0	0	843.31	-190.9099235
3	-152.64	1	1	-152.64	690.67	-190.9099235
4	-239.81	0.477751	0	0	450.86	102.4900765
5	293.4	1	1	293.4	744.26	1162.670077
6	1060.18	1	1	1060.18	1804.44	1207.900077
7	45.23	1	1	45.23	1849.67	1216.630077
8	8.73	1	1	8.73	1858.4	1347.290077
9	130.66	1	1	130.66	1989.06	1394.470077
10	47.18	1	1	47.18	2036.24	1094.800077
11	-299.67	1	1	-299.67	1736.57	1094.800077
12	-338.82	0.477751	0	0	1397.75	595.6800765
13	-499.12	1	1	-499.12	898.63	595.6800765
14	-156.04	0.477751	0	0	742.59	509.8100765
15	-85.87	1	1	-85.87	656.72	672.0900765
16	162.28	1	1	162.28	819	672.0900765
17	182.97	0.477751	0	0	1001.97	500.424404
18	-292.75	0.58639	0.58639	-171.665673	709.22	500.424404
19	-787.08	0.477751	0	0	-77.86	500.424404
20	-1073.39	0.477751	0	0	-1151.25	700.844404
21	200.42	1	1	200.42	-950.83	603.594404
22	-97.25	1	1	-97.25	-1048.08	603.594404
23	14.84	0.477751	0	0	-1033.24	603.594404
24	-172.73	0.477751	0	0	-1205.97	603.594404
25	-425.83	0.477751	0	0	-1631.8	700.174404
26	96.58	1	1	96.58	-1535.22	977.744404
27	277.57	1	1	277.57	-1257.65	1515.894404
28	538.15	1	1	538.15	-719.5	1515.894404
29	-631.55	0.477751	0	0	-1351.05	1515.894404
30	22.15	0.477751	0	0	-1328.9	1515.894404
31	423.06	0.477751	0	0	-905.84	1201.004404
32	-314.89	1	1	-314.89	-1220.73	1065.474404
33	-135.53	1	1	-135.53	-1356.26	1432.354404
34	366.88	1	1	366.88	-989.38	705.7257406

35	-930.61	0.780809	0.780809	-726.628663	-1919.99	1058.165036
36	352.44	0.999998	0.999998	352.439295	-1567.55	835.8850357
37	-222.28	1	1	-222.28	-1789.83	835.8850357
38	-187.86	0.477751	0	0	-1977.69	1032.615036
39	196.73	1	1	196.73	-1780.96	1032.615036
40	17.87	0.477751	0	0	-1763.09	557.8750357
41	-474.74	1	1	-474.74	-2237.83	456.7650357
42	-101.11	1	1	-101.11	-2338.94	761.2650357
43	304.5	1	1	304.5	-2034.44	761.2650357
44	346.79	0.477751	0	0	-1687.65	371.4850357
45	-389.78	1	1	-389.78	-2077.43	740.5150357
46	369.03	1	1	369.03	-1708.4	420.4750357
47	-320.04	1	1	-320.04	-2028.44	420.4750357
48	-277.36	0.477751	0	0	-2305.8	684.5350357
49	264.06	1	1	264.06	-2041.74	2500.355036
50	1815.82	1	1	1815.82	-225.92	2500.355036
51	463.37	0.477751	0	0	237.45	2500.355036
52	-284.98	0.477751	0	0	-47.53	2781.415036
53	281.06	1	1	281.06	233.53	2832.575036
54	51.16	1	1	51.16	284.69	2832.575036
55	430.56	0.477751	0	0	715.25	2786.435036
56	-46.14	1	1	-46.14	669.11	3439.015036
57	652.58	1	1	652.58	1321.69	3629.675036
58	190.66	1	1	190.66	1512.35	3641.305036
59	11.63	1	1	11.63	1523.98	3641.305036
60	492.78	0.477751	0	0	2016.76	3030.805036
61	-610.5	1	1	-610.5	1406.26	2794.925036
62	-235.88	1	1	-235.88	1170.38	3036.465036
63	241.54	1	1	241.54	1411.92	2831.044034
64	-294.75	0.696933	0.696933	-205.421002	1117.17	3026.794034
65	195.75	1	1	195.75	1312.92	3026.794034
66	230.44	0.477751	0	0	1543.36	3200.294034
67	173.5	1	1	173.5	1716.86	3507.964034
68	307.67	1	1	307.67	2024.53	3018.624034
69	-489.34	1	1	-489.34	1535.19	2910.194034
70	-108.43	1	1	-108.43	1426.76	3320.264034
71	410.07	1	1	410.07	1836.83	3380.044034
72	59.78	1	1	59.78	1896.61	3422.894034

73	42.85	1	1	42.85	1939.46	3496.034034
74	73.14	1	1	73.14	2012.6	3327.824034
75	-168.21	1	1	-168.21	1844.39	3327.824034
76	397.9	0.477751	0	0	2242.29	3646.714034
77	318.89	1	1	318.89	2561.18	3494.704034
78	-152.01	1	1	-152.01	2409.17	3494.704034
79	-1369.58	0.477751	0	0	1039.59	3682.244034
80	187.54	1	1	187.54	1227.13	3682.244034
81	301.67	0.477751	0	0	1528.8	3485.654034
82	-196.59	1	1	-196.59	1332.21	3485.654034
83	-176.96	0.477751	0	0	1155.25	3743.944034
84	258.29	1	1	258.29	1413.54	3907.134034
85	163.19	1	1	163.19	1576.73	4245.814034
86	338.68	1	1	338.68	1915.41	4245.814034
87	-199.66	0.477751	0	0	1715.75	4245.814034
88	-790.39	0.477751	0	0	925.36	4266.744034
89	20.93	1	1	20.93	946.29	4500.344034
90	233.6	1	1	233.6	1179.89	4716.964034
91	216.62	1	1	216.62	1396.51	4072.434034
92	-644.53	1	1	-644.53	751.98	3400.744034
93	-671.69	1	1	-671.69	80.29	3455.104034
94	54.36	1	1	54.36	134.65	3405.814034
95	-49.29	1	1	-49.29	85.36	3637.624034
96	231.81	1	1	231.81	317.17	3637.624034

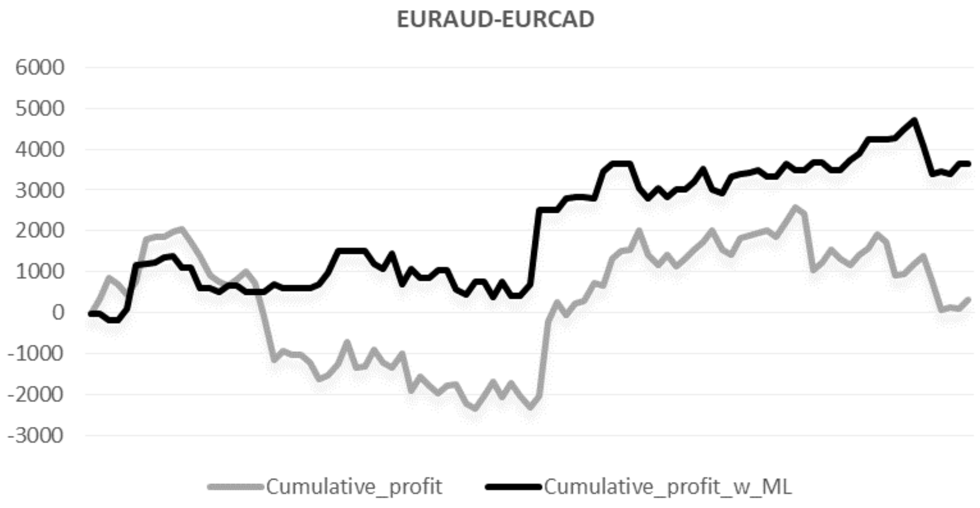

Figure 4-2 Cumulative profit of EURAUD-EURCAD pairs trading strategy with and without machine learning

According to the result, the final balance from using Artificial Neural Network is $3637 which is more than 10 times greater than normal pairs trading strategy. Next, All Machine learning and all technique are applied to EURAUD-EURCAD pair to depict the balance curves. Figure 4-2 shows the balance curves by using the Machine learning with normal pairs trading strategy.

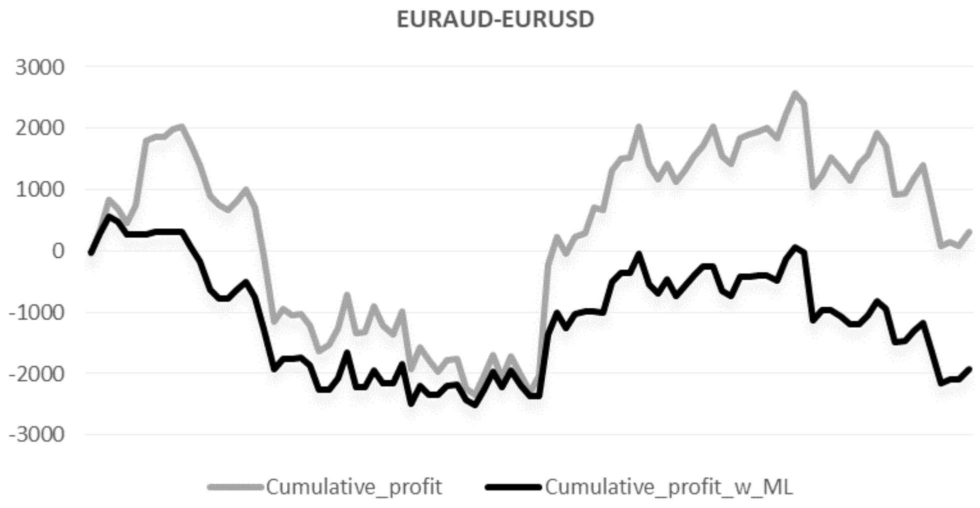

Figure 4-3 Cumulative profit of EURAUD-EURUSD pairs trading strategy with and without machine learning

According to the balance curve of EURAUD-EURUSD shown in Figure 4-2, the balance curve from using XGBoost predicting the total_Profit feature is lower than the balance curve of normal pairs trading strategy. Although the F1-score from the confusion matrix is 0.815 which is very satisfactory, the balance curve and final profit are very poor.

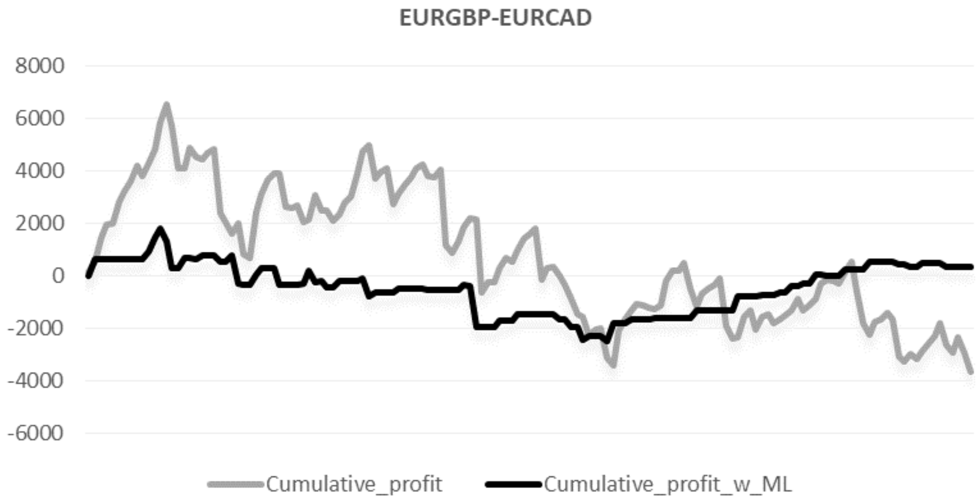

Figure 4-4 Cumulative profit of EURGBP-EURCAD pairs trading strategy with and without machine learning

According to the balance curve of EURGBP-EURAUD shown in Figure 4-4, Logistic Regression yields 0.772 of F1-score which is very high. However, the balance curve from using Logistic Regression on predicting Total_profit is worse than normal pairs trading strategy in the first half of trading. In the latter half, the balance curve gradually cumulates the profit and wins beats the normal pairs trading strategy at the final cumulative profit.

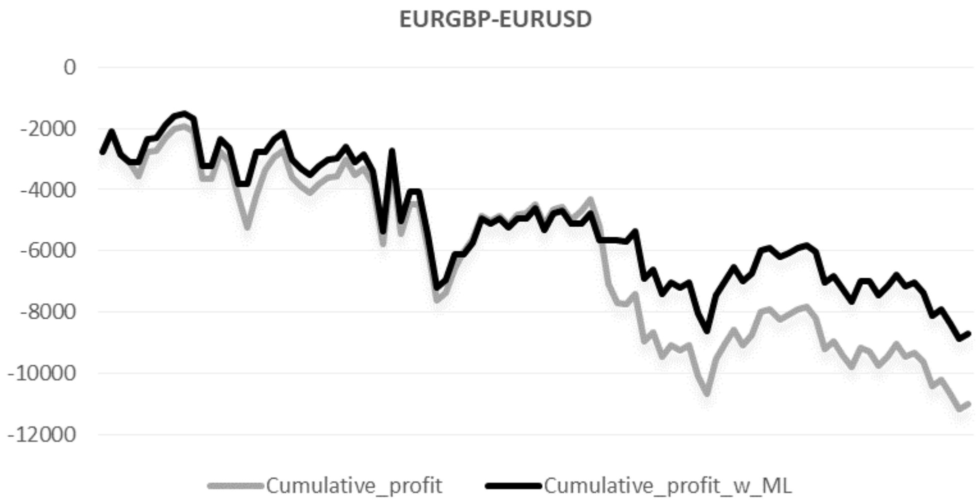

Figure 4-5 Cumulative profit of EURGBP-EURUSD pairs trading strategy with and without machine learning

According to the balance curve of EURAGBPEURUSD shown in Figure 4-5, Artificial Neural Network yields 0.736 of F1-score which is very high. The balance curve from using Artificial Neural Network on predicting Total_profit is vague. The balance curve at the first half is not apparently improved. The balance curve from using machine learning is gradually improved. Although the final balance is loss, the loss is lower than normal pairs trading strategy.

5 RESULT

The result of this thesis is to answer the question that whether the Machine learning algorithm help improve the trading system or not. Regarding to previous section, if the traders focus in the score matrix, the scores are quite satisfied. The algorithms have potential to classified whether the signals are profitable or not. However, in profit aspect, the balance curves from the out-of-sample trading are different. Some algorithms do not generate profit even the scores are more than 60%. Thus, the profit/loss records are statistically evaluated in this section to answer the question

5.1 BALANCE COMPARISON

Regarding to profit expected return test, the expected return from implementing the machine learning is not significantly greater than zero and average return from implementing the machine learning is not significantly greater and the average return from normal pairs trading strategy. However, the evaluation of these methods may be too harsh. The reason is that the unexpected situations may occur which affect the fundamental of the exchange rate such as the consensus of interest rate, GDP announcement, Declaration of unemployment rate, etc. Therefore, the past parameters such as spread threshold and period of RSI might not be profitable in the future. Thus, the machine learning could not improve significantly. Another method to test the improvement is to compare the balance at every point of time. When the profit/loss is realized, the balance is increased/decreased. If the machine learning helps improve the trade, the balance from using machine learning should be greater than the balance from normal pairs trading strategy. The statistical test used to evaluate the balance is paired t-test. The parameter which used to imply the performance of machine learning is the distance or difference between the balance using machine learning and the balance with normal pairs trading strategy.

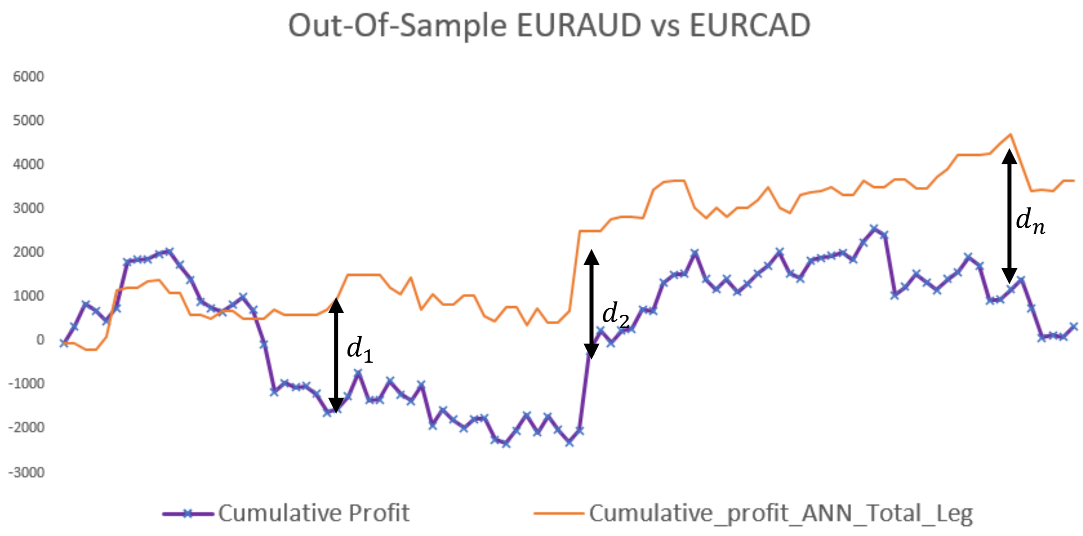

Figure *5-1* Distance on balance curve of normal pairs trading strategy and the strategy with machine learning

Figure 5-1 illustrate the distance or the difference of balance from normal pairs trading strategy and the strategy with Artificial Neural Network on EURAUD-EURCAD. If the machine learning helps improve the trading performance, the average of the distance is significantly greater than zero. This statement is translated into hypothesis test.

$$H_0: \mu_{\bar{d}} \leq 0$$

$$H_a: \mu_{\bar{d}} > 0 \quad \text{at significant level } \alpha = 0.05$$

Table 5-1 summarized the p-value from paired t-test.

Table 5-1 Summary of t-value and p-value of paired t-test

EURAUDEURCAD	T-Value	P-Value	Result
ANN_EURAUDEURCAD_Total	12.52	0.000	Improved
EURAUDEURUSD	**T-Value**	**P-Value**	**Result**
XGB_EURAUDEURUSD_Total	-25.1	1.000	Failed
EURGBPEURCAD	**T-Value**	**P-Value**	**Result**
LR_EURGBPEURCAD_Total	-6.32	1.000	Failed
EURGBPEURUSD	**T-Value**	**P-Value**	**Result**
ANN_EURGBPEURUSD_Total	16.43	0.000	Improved

According to Table 5-1, Artificial Neural Network works very well in EURAUD-EURCAD and EURGBP-EURUSD. The balances from using Artificial Neural Network beats the balance from normal pairs trading strategy. Although the final balance in EURGBP-EURUSD is loss, the balance from Artificial Neural Network is significantly greater than the balance from normal pairs trading strategy. It implies that the machine learning help reduce drawdown and loss.

For EURAUD-EURUSD, the algorithm does not improve trading performance. Although the F1-score from out-of-sample is very high. The XGBoost is not significantly capable to win the normal pairs trading strategy in EURAUD-EURCAD. Similarly, the result from EURGBP-EURAUD is not improved, but the final balance from using machine learning is greater normal pairs trading strategy.

The factor which influences the prediction is the size of profit and loss. According to the methodology, Artificial Neural Network, Logistic Regression and XGBoost are very generalized regarding to the in-sample and out-of-sample score. Although the F1-score from the Artificial Neural Network ,Logistic Regression and XGBoost are very satisfactory, the payoff from correct prediction is lower than the loss from wrong prediction causing the loss poor performance in cumulative profit.

5.2 CONCLUSION

The implementation of machine learning algorithm is satisfactory in some conditions. According to the in-sample score in k-fold cross validation, the machine learning algorithm predicting score very satisfied which means that the performance of the machine learning is capable to classify the profitable signals based on lagging indicators.

The algorithm of each pairs is selected by the best F1-score to trade the out-of-sample data. The performance of the out-of-sample is expected to be similar to k-Fold cross validation score in GridSearchCV. Thus, the algorithms which have the highest F1-sccore on each pair are selected. In summary, Artificial Neural Network is applied in EURAUD-EURCAD and EURGBP-EURUSD, XGBoost for EURAUD-EURUSD and Logistic Regression for EURGBP-EURAUD. Next, the algorithms are applied to out-of-sample data. The result shows that the F!-score on out-of-sample data persist to in-sample score. For example, Artificial Neural Network on predicting EURAUD-EURCAD Total_leg, the F1 score from cross validation is 0.771 and the out-of-sample score is 0.64. Therefore, it can be concluded that the machine learning has capability to classify the signal from lagging indicator.

In the profit aspect, the results conclude that best algorithm and the best pairs of exchange rates are Artificial Neural Network on EURAUD-EURCAD. For EURAUD-EURUSD and EURGBP-EURCAD, the best models from selecting in in-sample data, which are XGBoost on EURAUD-EURUSD and Logistic Regression on EURGBP-EURAUD, do not significantly improve the trading performance regarding to the balance comparison. For Artificial Neural Network on EURGBP-EURUSD, Although the balance comparison shows that the balance from using Artificial Neural Network is significantly improved the trading performance, the cumulative profit is loss. The reason is the potential of the pair EURGBP-EURUSD which has mean reversion behavior is faded over time. The EURGBP-EURUSD is not persistent to mispricing and equilibrium behavior. Finally, the algorithm

Artificial Neural Network is selected and proved that the algorithm is capable to predict the profitability of EURAUD-EURCAD.

5.3 LIMITATION OF THE RESEARCH

1. Most of the features has high dimensional of the parameter to fine-tuned. The stress indicator which is selected in the thesis is Relative Strength Index with period 14. The period of the stress indicator is the parameter that can be tuned to adjust the frequency of the trade. It affects the profitability in the pairs selecting selection. However, If the period of the RSI is considered as the parameter, the data which are needed to handle is overwhelmed. Another limitation is computer speed, most algorithm consumed a lot of time such as the GridSearchCV in Artificial Neural Network and XGBoost. The inconvenience resource causes the ability to tune up the model.
2. For XGBoost, Applying the GridSearchCV in every combination of hyper-parameter consumes a lot of time. Therefore, this thesis cannot perform all combinations. The set of hyperparameter from GridSearchCV in this thesis might not yield the best F1-score compared to all combinations.

5.4 FUTURE WORK

In this thesis, there are a lot of room for improvement. There are many parameters in this thesis to be tune. For example, RSI period, the period of RSI causes the sensitivity and profitability of the exchange rate pairs. The period of RSI could change the non-profitability pairs into profitability pairs

In machine learning aspect, there are many hyperparameter to be tune but not mentioned in this thesis such as alpha, L1 and L2 regularization in XGBoost, optimizer, batch_size and n_epochs in Artificial Neural Network.

The classifying threshold in classification problem can be adjusted. In this thesis, threshold 0.5 is used. The threshold affects the sensitivity of classification causing the performance of the machine learning algorithms. In the future work, this value should be adjusted and analyzed.

As a result, the machine learning model helps the pairs trading strategy to improve trading performance. In the future work, researchers can extend the methodology to other assets such as stocks, ETF, Cryptocurrency, etc. Another work is the feature engineering, regarding to the result, features hugely impact the performance of the models. Thus, the researchers should study the feature extraction methodology to improve the model.

6 APPENDIX

6.1 APPENDIX 1 ANOVA UNIVARIATE TEST RESULT

Table A- 1 ANOVA p-value on Profit_L1 EURAUDEURCAD

Profit_L1

EURAUDEURCAD	
	Value
Correlation[20]	0.00
VolatilityLeg1[0]	0.06
beta_10	0.10
Correlation	0.11
VolatilityDiff	0.16
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.16
beta_20	0.18
Hour	0.19
MomLeg1[0]	0.20
Type	0.24
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.26
Correlation[40]	0.28
Duration	0.29
beta_50	0.31
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.36
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.40
MomLeg2[0]	0.44
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.58
beta_100	0.59
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.77
Correlation[80]	0.85
Correlation[100]	0.87
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.88
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.91
VolatilityLeg2[0]	0.97

Table A- 2ANOVA p-value on Profit_L2 EURAUDEURCAD

Profit_L2

EURAUDEURCAD	
	Value
VolatilityLeg2[0]	0.02
VolatilityDiff	0.05
Correlation[20]	0.10
Correlation	0.10
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.23
Hour	0.24
Correlation[40]	0.24
Duration	0.33
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.33
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.36
VolatilityLeg1[0]	0.42
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.46
beta_10	0.48
MomLeg1[0]	0.61
beta_50	0.61
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.64
beta_100	0.68
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.69
MomLeg2[0]	0.73
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.77
Type	0.78
Correlation[80]	0.81
beta_20	0.83
Correlation[100]	0.89
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.97

Table A- 3 ANOVA p-value on Total_Profit EURAUDEURCAD

Total_Profit

EURAUDEURCAD	
	Value
Correlation[20]	0.01
Type	0.01
MomLeg2[0]	0.01
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.03
MomLeg1[0]	0.03
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.06
beta_10	0.07
beta_20	0.13
VolatilityLeg1[0]	0.15
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.16
Duration	0.17
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.22
beta_50	0.33
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.33
VolatilityDiff	0.41
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.52
Correlation	0.55
Correlation[40]	0.56
Hour	0.59
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.62
Correlation[100]	0.64
VolatilityLeg2[0]	0.69
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.79
beta_100	0.94
Correlation[80]	0.97

Table A- 4 ANOVA p-value on Profit_L1 EURAUDEURUSD

Profit_L1

EURAUDEURUSD	
	Value
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.02
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.04
beta_10	0.07
Correlation[20]	0.11
Correlation[40]	0.12
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.16
MomLeg1[0]	0.26
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.27
beta_100	0.32
Type	0.33
Duration	0.34
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.35
beta_50	0.41
MomLeg2[0]	0.42
VolatilityLeg1[0]	0.54
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.60
Hour	0.61
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.65
beta_20	0.70
VolatilityDiff	0.78
Correlation	0.88
Correlation[100]	0.88
VolatilityLeg2[0]	0.91
Correlation[80]	0.94
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.96

Table A- 5 ANOVA p-value on Profit_L2 EURAUDEURUSD

Profit_L2

EURAUDEURUSD	
	Value
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.04
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.12
Hour	0.18
Correlation[80]	0.31
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.41
Correlation	0.42
beta_20	0.43
Type	0.49
Duration	0.50
Correlation[40]	0.52
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.53
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.54
MomLeg2[0]	0.55
VolatilityLeg2[0]	0.73
beta_100	0.76
MomLeg1[0]	0.79
beta_50	0.79
beta_10	0.81
Correlation[20]	0.82
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.84
VolatilityDiff	0.88
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.89
Correlation[100]	0.93
VolatilityLeg1[0]	0.93
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	1.00

Table A- 6 ANOVA p-value on Total_Profit EURAUDEURUSD

Total_Profit

EURAUDEURUSD	
	Value
Correlation[40]	0.03
Correlation[80]	0.16
MomLeg2[0]	0.18
beta_50	0.25
Type	0.28
MomLeg1[0]	0.32
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.34
Correlation[20]	0.35
VolatilityLeg2[0]	0.37
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.37
VolatilityDiff	0.39
Duration	0.48
Correlation[100]	0.52
beta_100	0.56
VolatilityLeg1[0]	0.58
Hour	0.62
Correlation	0.63
beta_20	0.65
beta_10	0.68
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.71
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.79
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.86
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.89
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.97
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	1.00

Table A- 7 ANOVA p-value on Profit_L1 EURGBPEURCAD

Profit_L1

EURGBPEURCAD	
	Value
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.21
Correlation[100]	0.24
Correlation[80]	0.29
Correlation	0.29
beta_50	0.32
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.33
beta_20	0.38
Duration	0.45
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.45
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.46
Hour	0.46
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.46
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.55
Correlation[20]	0.63
Correlation[40]	0.69
beta_10	0.71
MomLeg1[0]	0.75
VolatilityLeg2[0]	0.76
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.79
Type	0.80
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.82
VolatilityDiff	0.83
beta_100	0.95
MomLeg2[0]	0.97
VolatilityLeg1[0]	0.99

Table A- 8 ANOVA p-value on Profit_L2 EURGBPEURCAD

Profit_L2

EURGBPEURCAD	
	Value
VolatilityDiff	0.00
VolatilityLeg2[0]	0.01
VolatilityLeg1[0]	0.01
beta_10	0.10
beta_20	0.14
beta_50	0.14
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.16
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.19
MomLeg1[0]	0.29
Duration	0.31
Correlation[20]	0.47
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.47
Correlation	0.49
Type	0.59
Hour	0.64
beta_100	0.67
Correlation[80]	0.71
Correlation[100]	0.78
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.81
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.82
Correlation[40]	0.82
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.83
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.89
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.92
MomLeg2[0]	0.95

Table A- 9 ANOVA p-value on Total_Profit EURGBPEURCAD

Total_Profit

EURGBPEURCAD	
	Value
VolatilityDiff	0.06
VolatilityLeg2[0]	0.07
Correlation	0.14
beta_20	0.17
VolatilityLeg1[0]	0.23
beta_10	0.30
Correlation[40]	0.41
Duration	0.41
MomLeg1[0]	0.42
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.48
Hour	0.51
Type	0.58
Correlation[80]	0.64
beta_100	0.65
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.75
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.75
MomLeg2[0]	0.79
Correlation[100]	0.82
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.83
beta_50	0.89
Correlation[20]	0.92
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.93
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.94
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.94
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.99

Table A- 10 ANOVA p-value on Profit_L1 EURGBPEURUSD

Profit_L1

EURGBPEURUSD	
	Value
Correlation[80]	0.09
Correlation[100]	0.13
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.14
Correlation	0.14
Correlation[20]	0.15
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.16
Hour	0.20
Correlation[40]	0.29
Duration	0.30
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.31
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.35
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.39
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.44
beta_10	0.46
VolatilityDiff	0.57
Type	0.57
MomLeg1[0]	0.60
beta_20	0.61
VolatilityLeg2[0]	0.61
MomLeg2[0]	0.63
VolatilityLeg1[0]	0.67
beta_100	0.68
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.82
beta_50	0.95
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.97

Table A- 11 ANOVA p-value on Profit_L2 EURGBPEURUSD

Profit_L2

EURGBPEURUSD	
	Value
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.01
Correlation	0.04
VolatilityLeg1[0]	0.13
Correlation[20]	0.28
Duration	0.31
beta_20	0.36
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.39
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.42
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.42
beta_50	0.42
beta_10	0.44
beta_100	0.48
MomLeg1[0]	0.52
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.56
Correlation[80]	0.57
VolatilityLeg2[0]	0.59
Hour	0.63
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.65
VolatilityDiff	0.65
Type	0.66
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.66
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.69
Correlation[100]	0.70
Correlation[40]	0.77
MomLeg2[0]	0.89

Table A- 12 ANOVA p-value on Total_Profit EURGBPEURUSD

Total_Profit

EURGBPEURUSD	
	Value
MAShortValue_Leg1[1](50)-MALongValue_Leg1[1](100)	0.00
Correlation[40]	0.01
VolatilityLeg2[0]	0.02
MomLeg2[0]	0.04
VolatilityDiff	0.10
Type	0.11
beta_50	0.12
MAShortValue_Leg2[1](5)-MALongValue_Leg2[1](10)	0.13
Correlation	0.15
MAShortValue_Leg1[1](5)-MALongValue_Leg1[1](10)	0.20
MAShortValue_Leg2[1](50)-MALongValue_Leg2[1](100)	0.22
Correlation[20]	0.22
beta_100	0.23
Correlation[80]	0.25
MomLeg1[0]	0.25
Duration	0.31
beta_20	0.32
Correlation[100]	0.42
MAShortValue_Leg2[1]-MALongValue_Leg2[1]	0.62
MAShortValue_Leg1[1](30)-MALongValue_Leg1[1](50)	0.63
beta_10	0.64
MAShortValue_Leg2[1](30)-MALongValue_Leg2[1](50)	0.70
MAShortValue_Leg1[1]-MALongValue_Leg1[1]	0.81
VolatilityLeg1[0]	0.86
Hour	0.94

6.2 APPENDIX B GRIDSEARHCV RESULTS

Table B- 13 GridSearchCV result of Artificial Neural Network

Pairs	output	batch	epoch	Layer	Node	matrix	k1	k2	k3	k4	k5	average
EURAUDEURCAD	Profit_L1	5	1000	1	1	'train_accuracy'	0.705	0.637	0.673	0.681	0.655	0.670
EURAUDEURCAD	Profit_L1	5	1000	1	1	'test_precision'	0.571	0.722	0.684	0.625	0.611	0.643
EURAUDEURCAD	Profit_L1	5	1000	1	1	'test_recall'	0.800	0.867	0.867	0.667	0.733	0.787
EURAUDEURCAD	Profit_L1	5	1000	1	1	'test_f1'	0.667	0.788	0.765	0.645	0.667	0.706
EURAUDEURCAD	Profit_L1	5	1000	1	2	'train_accuracy'	0.688	0.664	0.655	0.699	0.664	0.674
EURAUDEURCAD	Profit_L1	5	1000	1	2	'test_precision'	0.632	0.765	0.684	0.625	0.611	0.663
EURAUDEURCAD	Profit_L1	5	1000	1	2	'test_recall'	0.800	0.867	0.867	0.667	0.733	0.787
EURAUDEURCAD	Profit_L1	5	1000	1	2	'test_f1'	0.706	0.813	0.765	0.645	0.667	0.719
EURAUDEURCAD	Profit_L1	5	1000	1	3	'train_accuracy'	0.688	0.673	0.664	0.681	0.673	0.676
EURAUDEURCAD	Profit_L1	5	1000	1	3	'test_precision'	0.632	0.750	0.684	0.625	0.611	0.660
EURAUDEURCAD	Profit_L1	5	1000	1	3	'test_recall'	0.800	0.800	0.867	0.667	0.733	0.773
EURAUDEURCAD	Profit_L1	5	1000	1	3	'test_f1'	0.706	0.774	0.765	0.645	0.667	0.711
EURAUDEURCAD	Profit_L1	5	2000	2	1	'train_accuracy'	0.679	0.690	0.681	0.726	0.708	0.697
EURAUDEURCAD	Profit_L1	5	2000	2	1	'test_precision'	0.647	0.688	0.692	0.647	0.588	0.652
EURAUDEURCAD	Profit_L1	5	2000	2	1	'test_recall'	0.733	0.733	0.600	0.733	0.667	0.693
EURAUDEURCAD	Profit_L1	5	2000	2	1	'test_f1'	0.688	0.710	0.643	0.688	0.625	0.671
EURAUDEURCAD	Profit_L1	5	2000	2	2	'train_accuracy'	0.732	0.664	0.699	0.708	0.690	0.699
EURAUDEURCAD	Profit_L1	5	2000	2	2	'test_precision'	0.571	0.733	0.667	0.625	0.588	0.637
EURAUDEURCAD	Profit_L1	5	2000	2	2	'test_recall'	0.800	0.733	0.933	0.667	0.667	0.760
EURAUDEURCAD	Profit_L1	5	2000	2	2	'test_f1'	0.667	0.733	0.778	0.645	0.625	0.690
EURAUDEURCAD	Profit_L1	5	2000	2	3	'train_accuracy'	0.732	0.708	0.664	0.726	0.735	0.713
EURAUDEURCAD	Profit_L1	5	2000	2	3	'test_precision'	0.571	0.722	0.600	0.647	0.588	0.626
EURAUDEURCAD	Profit_L1	5	2000	2	3	'test_recall'	0.800	0.867	1.000	0.733	0.667	0.813
EURAUDEURCAD	Profit_L1	5	2000	2	3	'test_f1'	0.667	0.788	0.750	0.688	0.625	0.703
EURAUDEURCAD	Profit_L1	5	3000	3	1	'train_accuracy'	0.536	0.531	0.531	0.531	0.531	0.532
EURAUDEURCAD	Profit_L1	5	3000	3	1	'test_precision'	0.517	0.536	0.536	0.536	0.536	0.532
EURAUDEURCAD	Profit_L1	5	3000	3	1	'test_recall'	1.000	1.000	1.000	1.000	1.000	1.000
EURAUDEURCAD	Profit_L1	5	3000	3	1	'test_f1'	0.682	0.698	0.698	0.698	0.698	0.695
EURAUDEURCAD	Profit_L1	5	3000	3	2	'train_accuracy'	0.732	0.531	0.531	0.726	0.726	0.649
EURAUDEURCAD	Profit_L1	5	3000	3	2	'test_precision'	0.571	0.536	0.536	0.647	0.588	0.576
EURAUDEURCAD	Profit_L1	5	3000	3	2	'test_recall'	0.800	1.000	1.000	0.733	0.667	0.840
EURAUDEURCAD	Profit_L1	5	3000	3	2	'test_f1'	0.667	0.698	0.698	0.688	0.625	0.675
EURAUDEURCAD	Profit_L1	5	3000	3	3	'train_accuracy'	0.723	0.690	0.699	0.735	0.531	0.676
EURAUDEURCAD	Profit_L1	5	3000	3	3	'test_precision'	0.571	0.722	0.706	0.647	0.536	0.636
EURAUDEURCAD	Profit_L1	5	3000	3	3	'test_recall'	0.800	0.867	0.800	0.733	1.000	0.840
EURAUDEURCAD	Profit_L1	5	3000	3	3	'test_f1'	0.667	0.788	0.750	0.688	0.698	0.718
EURAUDEURCAD	Profit_L2	5	1000	1	1	'train_accuracy'	0.661	0.664	0.628	0.655	0.628	0.647

EURAUDEURCAD	Profit_L2	5	1000	1	1	'test_precision'	0.522	0.667	0.625	0.467	0.500	0.556
EURAUDEURCAD	Profit_L2	5	1000	1	1	'test_recall'	0.800	0.800	0.714	0.500	0.571	0.677
EURAUDEURCAD	Profit_L2	5	1000	1	1	'test_f1'	0.632	0.727	0.667	0.483	0.533	0.608
EURAUDEURCAD	Profit_L2	5	1000	1	2	'train_accuracy'	0.670	0.628	0.628	0.664	0.628	0.644
EURAUDEURCAD	Profit_L2	5	1000	1	2	'test_precision'	0.522	0.577	0.667	0.500	0.500	0.553
EURAUDEURCAD	Profit_L2	5	1000	1	2	'test_recall'	0.800	1.000	0.429	0.500	0.500	0.646
EURAUDEURCAD	Profit_L2	5	1000	1	2	'test_f1'	0.632	0.732	0.522	0.500	0.500	0.577
EURAUDEURCAD	Profit_L2	5	1000	1	3	'train_accuracy'	0.670	0.646	0.655	0.655	0.646	0.654
EURAUDEURCAD	Profit_L2	5	1000	1	3	'test_precision'	0.545	0.600	0.667	0.467	0.571	0.570
EURAUDEURCAD	Profit_L2	5	1000	1	3	'test_recall'	0.800	1.000	0.429	0.500	0.571	0.660
EURAUDEURCAD	Profit_L2	5	1000	1	3	'test_f1'	0.649	0.750	0.522	0.483	0.571	0.595
EURAUDEURCAD	Profit_L2	5	2000	2	1	'train_accuracy'	0.670	0.673	0.513	0.619	0.655	0.626
EURAUDEURCAD	Profit_L2	5	2000	2	1	'test_precision'	0.522	0.619	0.500	0.571	0.462	0.535
EURAUDEURCAD	Profit_L2	5	2000	2	1	'test_recall'	0.800	0.867	1.000	0.857	0.429	0.790
EURAUDEURCAD	Profit_L2	5	2000	2	1	'test_f1'	0.632	0.722	0.667	0.686	0.444	0.630
EURAUDEURCAD	Profit_L2	5	2000	2	2	'train_accuracy'	0.670	0.637	0.673	0.681	0.690	0.670
EURAUDEURCAD	Profit_L2	5	2000	2	2	'test_precision'	0.500	0.632	0.750	0.538	0.563	0.597
EURAUDEURCAD	Profit_L2	5	2000	2	2	'test_recall'	0.867	0.800	0.429	0.500	0.643	0.648
EURAUDEURCAD	Profit_L2	5	2000	2	2	'test_f1'	0.634	0.706	0.545	0.519	0.600	0.601
EURAUDEURCAD	Profit_L2	5	2000	2	3	'train_accuracy'	0.670	0.664	0.708	0.655	0.743	0.688
EURAUDEURCAD	Profit_L2	5	2000	2	3	'test_precision'	0.522	0.619	0.667	0.471	0.563	0.568
EURAUDEURCAD	Profit_L2	5	2000	2	3	'test_recall'	0.800	0.867	0.429	0.571	0.643	0.662
EURAUDEURCAD	Profit_L2	5	2000	2	3	'test_f1'	0.632	0.722	0.522	0.516	0.600	0.598
EURAUDEURCAD	Profit_L2	5	3000	3	1	'train_accuracy'	0.509	0.504	0.513	0.513	0.513	0.511
EURAUDEURCAD	Profit_L2	5	3000	3	1	'test_precision'	0.517	0.536	0.500	0.500	0.500	0.511
EURAUDEURCAD	Profit_L2	5	3000	3	1	'test_recall'	1.000	1.000	1.000	1.000	1.000	1.000
EURAUDEURCAD	Profit_L2	5	3000	3	1	'test_f1'	0.682	0.698	0.667	0.667	0.667	0.676
EURAUDEURCAD	Profit_L2	5	3000	3	2	'train_accuracy'	0.509	0.504	0.690	0.513	0.673	0.578
EURAUDEURCAD	Profit_L2	5	3000	3	2	'test_precision'	0.517	0.536	0.750	0.500	0.500	0.561
EURAUDEURCAD	Profit_L2	5	3000	3	2	'test_recall'	1.000	1.000	0.214	1.000	0.429	0.729
EURAUDEURCAD	Profit_L2	5	3000	3	2	'test_f1'	0.682	0.698	0.333	0.667	0.462	0.568
EURAUDEURCAD	Profit_L2	5	3000	3	3	'train_accuracy'	0.688	0.743	0.690	0.655	0.743	0.704
EURAUDEURCAD	Profit_L2	5	3000	3	3	'test_precision'	0.550	0.538	1.000	0.600	0.550	0.648
EURAUDEURCAD	Profit_L2	5	3000	3	3	'test_recall'	0.733	0.467	0.214	0.214	0.786	0.483
EURAUDEURCAD	Profit_L2	5	3000	3	3	'test_f1'	0.629	0.500	0.353	0.316	0.647	0.489
EURAUDEURCAD	Total_Profit	5	1000	1	1	'train_accuracy'	0.723	0.805	0.726	0.752	0.752	0.752
EURAUDEURCAD	Total_Profit	5	1000	1	1	'test_precision'	0.704	0.579	0.600	0.696	0.700	0.656
EURAUDEURCAD	Total_Profit	5	1000	1	1	'test_recall'	1.000	0.611	0.833	0.889	0.737	0.814
EURAUDEURCAD	Total_Profit	5	1000	1	1	'test_f1'	0.826	0.595	0.698	0.780	0.718	0.723
EURAUDEURCAD	Total_Profit	5	1000	1	2	'train_accuracy'	0.723	0.841	0.779	0.788	0.788	0.784
EURAUDEURCAD	Total_Profit	5	1000	1	2	'test_precision'	0.727	0.591	0.625	0.727	0.700	0.674
EURAUDEURCAD	Total_Profit	5	1000	1	2	'test_recall'	0.842	0.722	0.833	0.889	0.737	0.805

EURAUDEURCAD	Total_Profit	5	1000	1	2	'test_f1'	0.780	0.650	0.714	0.800	0.718	0.733
EURAUDEURCAD	Total_Profit	5	1000	1	3	'train_accuracy'	0.777	0.814	0.761	0.796	0.841	0.798
EURAUDEURCAD	Total_Profit	5	1000	1	3	'test_precision'	0.739	0.625	0.591	0.708	0.737	0.680
EURAUDEURCAD	Total_Profit	5	1000	1	3	'test_recall'	0.895	0.833	0.722	0.944	0.737	0.826
EURAUDEURCAD	Total_Profit	5	1000	1	3	'test_f1'	0.810	0.714	0.650	0.810	0.737	0.744
EURAUDEURCAD	Total_Profit	5	2000	2	1	'train_accuracy'	0.750	0.832	0.743	0.823	0.752	0.780
EURAUDEURCAD	Total_Profit	5	2000	2	1	'test_precision'	0.704	0.632	0.615	0.682	0.727	0.672
EURAUDEURCAD	Total_Profit	5	2000	2	1	'test_recall'	1.000	0.667	0.889	0.833	0.842	0.846
EURAUDEURCAD	Total_Profit	5	2000	2	1	'test_f1'	0.826	0.649	0.727	0.750	0.780	0.746
EURAUDEURCAD	Total_Profit	5	2000	2	2	'train_accuracy'	0.804	0.858	0.850	0.796	0.850	0.832
EURAUDEURCAD	Total_Profit	5	2000	2	2	'test_precision'	0.708	0.667	0.706	0.696	0.737	0.703
EURAUDEURCAD	Total_Profit	5	2000	2	2	'test_recall'	0.895	0.778	0.667	0.889	0.737	0.793
EURAUDEURCAD	Total_Profit	5	2000	2	2	'test_f1'	0.791	0.718	0.686	0.780	0.737	0.742
EURAUDEURCAD	Total_Profit	5	2000	2	3	'train_accuracy'	0.875	0.894	0.841	0.929	0.823	0.872
EURAUDEURCAD	Total_Profit	5	2000	2	3	'test_precision'	0.783	0.619	0.591	0.727	0.696	0.683
EURAUDEURCAD	Total_Profit	5	2000	2	3	'test_recall'	0.947	0.722	0.722	0.889	0.842	0.825
EURAUDEURCAD	Total_Profit	5	2000	2	3	'test_f1'	0.857	0.667	0.650	0.800	0.762	0.747
EURAUDEURCAD	Total_Profit	5	3000	3	1	'train_accuracy'	0.696	0.841	0.655	0.655	0.646	0.699
EURAUDEURCAD	Total_Profit	5	3000	3	1	'test_precision'	0.882	0.632	0.643	0.643	0.679	0.696
EURAUDEURCAD	Total_Profit	5	3000	3	1	'test_recall'	0.789	0.667	1.000	1.000	1.000	0.891
EURAUDEURCAD	Total_Profit	5	3000	3	1	'test_f1'	0.852	0.668	0.802	0.802	0.828	0.790
EURAUDEURCAD	Total_Profit	5	3000	3	2	'train_accuracy'	0.839	0.858	0.903	0.841	0.832	0.855
EURAUDEURCAD	Total_Profit	5	3000	3	2	'test_precision'	0.778	0.667	0.571	0.737	0.765	0.703
EURAUDEURCAD	Total_Profit	5	3000	3	2	'test_recall'	0.737	0.778	0.444	0.778	0.684	0.684
EURAUDEURCAD	Total_Profit	5	3000	3	2	'test_f1'	0.757	0.718	0.500	0.757	0.722	0.691
EURAUDEURCAD	Total_Profit	5	3000	3	3	'train_accuracy'	0.902	0.929	0.903	0.920	0.929	0.917
EURAUDEURCAD	Total_Profit	5	3000	3	3	'test_precision'	0.765	0.667	0.684	0.650	0.700	0.693
EURAUDEURCAD	Total_Profit	5	3000	3	3	'test_recall'	0.684	0.778	0.722	0.722	0.737	0.729
EURAUDEURCAD	Total_Profit	5	3000	3	3	'test_f1'	0.722	0.718	0.703	0.684	0.718	0.709
EURGBPEURCAD	Total_Profit	5	1000	1	1	'train_accuracy'	0.766	0.734	0.766	0.766	0.766	0.759
EURGBPEURCAD	Total_Profit	5	1000	1	1	'test_precision'	0.733	0.727	0.700	0.600	0.583	0.669
EURGBPEURCAD	Total_Profit	5	1000	1	1	'test_recall'	1.000	0.727	0.700	0.900	0.700	0.805
EURGBPEURCAD	Total_Profit	5	1000	1	1	'test_f1'	0.846	0.727	0.700	0.720	0.636	0.726
EURGBPEURCAD	Total_Profit	5	1000	1	2	'train_accuracy'	0.844	0.875	0.797	0.797	0.781	0.819
EURGBPEURCAD	Total_Profit	5	1000	1	2	'test_precision'	0.714	0.600	0.700	0.643	0.583	0.648
EURGBPEURCAD	Total_Profit	5	1000	1	2	'test_recall'	0.909	0.545	0.700	0.900	0.700	0.751
EURGBPEURCAD	Total_Profit	5	1000	1	2	'test_f1'	0.800	0.571	0.700	0.750	0.636	0.692
EURGBPEURCAD	Total_Profit	5	1000	1	3	'train_accuracy'	0.891	0.891	0.859	0.797	0.781	0.844
EURGBPEURCAD	Total_Profit	5	1000	1	3	'test_precision'	0.667	0.583	0.700	0.643	0.700	0.659
EURGBPEURCAD	Total_Profit	5	1000	1	3	'test_recall'	0.545	0.636	0.700	0.900	0.700	0.696
EURGBPEURCAD	Total_Profit	5	1000	1	3	'test_f1'	0.600	0.609	0.700	0.750	0.700	0.672
EURGBPEURCAD	Total_Profit	5	2000	2	1	'train_accuracy'	0.813	0.875	0.766	0.813	0.813	0.816

EURGBPEURCAD	Total_Profit	5	2000	2	1	'test_precision'	0.750	0.714	0.875	0.643	0.583	0.713
EURGBPEURCAD	Total_Profit	5	2000	2	1	'test_recall'	0.818	0.455	0.700	0.900	0.700	0.715
EURGBPEURCAD	Total_Profit	5	2000	2	1	'test_f1'	0.783	0.556	0.778	0.750	0.636	0.700
EURGBPEURCAD	Total_Profit	5	2000	2	2	'train_accuracy'	0.781	0.891	0.875	0.875	0.906	0.866
EURGBPEURCAD	Total_Profit	5	2000	2	2	'test_precision'	0.727	0.800	0.778	0.600	0.714	0.724
EURGBPEURCAD	Total_Profit	5	2000	2	2	'test_recall'	0.727	0.727	0.700	0.900	0.500	0.711
EURGBPEURCAD	Total_Profit	5	2000	2	2	'test_f1'	0.727	0.762	0.737	0.720	0.588	0.707
EURGBPEURCAD	Total_Profit	5	2000	2	3	'train_accuracy'	0.891	0.969	0.969	0.922	0.859	0.922
EURGBPEURCAD	Total_Profit	5	2000	2	3	'test_precision'	0.778	0.545	0.778	0.750	0.636	0.697
EURGBPEURCAD	Total_Profit	5	2000	2	3	'test_recall'	0.636	0.545	0.700	0.600	0.700	0.636
EURGBPEURCAD	Total_Profit	5	2000	2	3	'test_f1'	0.700	0.545	0.737	0.667	0.667	0.663
EURGBPEURCAD	Total_Profit	5	3000	3	1	'train_accuracy'	0.750	0.828	0.828	0.781	0.875	0.813
EURGBPEURCAD	Total_Profit	5	3000	3	1	'test_precision'	0.778	0.700	0.857	0.800	0.750	0.777
EURGBPEURCAD	Total_Profit	5	3000	3	1	'test_recall'	0.636	0.636	0.600	0.400	0.600	0.575
EURGBPEURCAD	Total_Profit	5	3000	3	1	'test_f1'	0.700	0.667	0.706	0.533	0.667	0.655
EURGBPEURCAD	Total_Profit	5	3000	3	2	'train_accuracy'	0.813	0.906	0.891	0.859	0.906	0.875
EURGBPEURCAD	Total_Profit	5	3000	3	2	'test_precision'	0.778	0.600	0.778	0.714	0.636	0.701
EURGBPEURCAD	Total_Profit	5	3000	3	2	'test_recall'	0.636	0.545	0.700	0.500	0.700	0.616
EURGBPEURCAD	Total_Profit	5	3000	3	2	'test_f1'	0.700	0.571	0.737	0.588	0.667	0.653
EURGBPEURCAD	Total_Profit	5	3000	3	3	'train_accuracy'	0.875	0.969	0.938	0.953	0.891	0.925
EURGBPEURCAD	Total_Profit	5	3000	3	3	'test_precision'	0.692	0.700	0.714	0.750	0.600	0.691
EURGBPEURCAD	Total_Profit	5	3000	3	3	'test_recall'	0.818	0.636	0.500	0.600	0.600	0.631
EURGBPEURCAD	Total_Profit	5	3000	3	3	'test_f1'	0.750	0.667	0.588	0.667	0.600	0.654
EURGBPEURUSD	Total_Profit	5	1000	1	1	'train_accuracy'	0.702	0.714	0.724	0.743	0.752	0.727
EURGBPEURUSD	Total_Profit	5	1000	1	1	'test_precision'	0.750	0.591	0.600	0.727	0.615	0.657
EURGBPEURUSD	Total_Profit	5	1000	1	1	'test_recall'	0.643	1.000	0.462	0.571	0.571	0.649
EURGBPEURUSD	Total_Profit	5	1000	1	1	'test_f1'	0.692	0.743	0.522	0.640	0.593	0.638
EURGBPEURUSD	Total_Profit	5	1000	1	2	'train_accuracy'	0.692	0.695	0.733	0.752	0.771	0.729
EURGBPEURUSD	Total_Profit	5	1000	1	2	'test_precision'	0.727	0.591	0.571	0.750	0.643	0.656
EURGBPEURUSD	Total_Profit	5	1000	1	2	'test_recall'	0.571	1.000	0.308	0.643	0.643	0.633
EURGBPEURUSD	Total_Profit	5	1000	1	2	'test_f1'	0.640	0.743	0.400	0.692	0.643	0.624
EURGBPEURUSD	Total_Profit	5	1000	1	3	'train_accuracy'	0.750	0.743	0.781	0.743	0.781	0.760
EURGBPEURUSD	Total_Profit	5	1000	1	3	'test_precision'	0.750	0.650	0.571	0.667	0.643	0.656
EURGBPEURUSD	Total_Profit	5	1000	1	3	'test_recall'	0.429	1.000	0.308	0.857	0.643	0.647
EURGBPEURUSD	Total_Profit	5	1000	1	3	'test_f1'	0.545	0.788	0.400	0.750	0.643	0.625
EURGBPEURUSD	Total_Profit	5	2000	2	1	'train_accuracy'	0.731	0.695	0.667	0.724	0.762	0.716
EURGBPEURUSD	Total_Profit	5	2000	2	1	'test_precision'	0.750	0.688	0.500	0.632	0.667	0.647
EURGBPEURUSD	Total_Profit	5	2000	2	1	'test_recall'	0.429	0.846	0.154	0.857	0.714	0.600
EURGBPEURUSD	Total_Profit	5	2000	2	1	'test_f1'	0.545	0.759	0.235	0.727	0.690	0.591
EURGBPEURUSD	Total_Profit	5	2000	2	2	'train_accuracy'	0.760	0.762	0.733	0.810	0.743	0.761
EURGBPEURUSD	Total_Profit	5	2000	2	2	'test_precision'	0.643	0.600	0.667	0.600	0.643	0.630
EURGBPEURUSD	Total_Profit	5	2000	2	2	'test_recall'	0.643	0.923	0.615	0.429	0.643	0.651

EURGBPEURUSD	Total_Profit	5	2000	2	2	'test_f1'	0.643	0.727	0.640	0.500	0.643	0.631
EURGBPEURUSD	Total_Profit	5	2000	2	3	'train_accuracy'	0.731	0.790	0.752	0.800	0.762	0.767
EURGBPEURUSD	Total_Profit	5	2000	2	3	'test_precision'	0.857	0.600	0.571	0.571	0.643	0.649
EURGBPEURUSD	Total_Profit	5	2000	2	3	'test_recall'	0.429	0.923	0.308	0.571	0.643	0.575
EURGBPEURUSD	Total_Profit	5	2000	2	3	'test_f1'	0.571	0.727	0.400	0.571	0.643	0.583
EURGBPEURUSD	Total_Profit	5	3000	3	1	'train_accuracy'	0.519	0.524	0.752	0.714	0.514	0.605
EURGBPEURUSD	Total_Profit	5	3000	3	1	'test_precision'	0.519	0.500	0.727	0.688	0.538	0.594
EURGBPEURUSD	Total_Profit	5	3000	3	1	'test_recall'	1.000	1.000	0.615	0.786	1.000	0.880
EURGBPEURUSD	Total_Profit	5	3000	3	1	'test_f1'	0.683	0.667	0.667	0.733	0.700	0.690
EURGBPEURUSD	Total_Profit	5	3000	3	2	'train_accuracy'	0.731	0.752	0.705	0.781	0.790	0.752
EURGBPEURUSD	Total_Profit	5	3000	3	2	'test_precision'	0.857	0.667	0.500	0.714	0.643	0.676
EURGBPEURUSD	Total_Profit	5	3000	3	2	'test_recall'	0.429	0.923	0.154	0.714	0.643	0.573
EURGBPEURUSD	Total_Profit	5	3000	3	2	'test_f1'	0.571	0.774	0.235	0.714	0.643	0.588
EURGBPEURUSD	Total_Profit	5	3000	3	3	'train_accuracy'	0.740	0.714	0.743	0.829	0.790	0.763
EURGBPEURUSD	Total_Profit	5	3000	3	3	'test_precision'	0.667	0.647	0.667	0.727	0.625	0.667
EURGBPEURUSD	Total_Profit	5	3000	3	3	'test_recall'	0.286	0.846	0.462	0.571	0.714	0.576
EURGBPEURUSD	Total_Profit	5	3000	3	3	'test_f1'	0.400	0.733	0.545	0.640	0.667	0.597

Table B- 14 GridSearchCV result of EURAUDEURCAD

		l1							l2						
	LR_EURAUD _EURCAD	0.001	0.01	0.1	1	10	100	1000	0.001	0.01	0.1	1	10	100	1000
Total	'split0_test_recall':	1.000	1.000	1.000	0.947	0.947	0.947	0.947	1.000	1.000	0.947	0.947	0.947	0.947	0.947
	'split1_test_recall':	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split2_test_recall':	1.000	1.000	0.889	0.778	0.778	0.778	0.778	0.944	0.944	0.833	0.778	0.778	0.778	0.778
	'split3_test_recall':	1.000	1.000	1.000	0.944	0.889	0.889	0.833	1.000	1.000	1.000	0.944	0.889	0.889	0.889
	'split4_test_recall':	1.000	1.000	0.895	0.842	0.789	0.789	0.737	0.895	0.842	0.842	0.842	0.789	0.789	0.789
	'mean_test_recall':	1.000	1.000	0.957	0.902	0.881	0.881	0.859	0.968	0.957	0.925	0.902	0.881	0.881	0.881
	'split0_train_recall':	1.000	1.000	1.000	0.918	0.918	0.932	0.890	1.000	0.986	0.986	0.945	0.932	0.932	0.932
	'split1_train_recall':	1.000	1.000	0.959	0.892	0.865	0.878	0.865	1.000	1.000	0.946	0.892	0.878	0.878	0.878
	'split2_train_recall':	1.000	1.000	0.959	0.892	0.878	0.878	0.878	1.000	1.000	0.932	0.892	0.878	0.878	0.878
	'split3_train_recall':	1.000	1.000	1.000	0.919	0.932	0.905	0.905	1.000	1.000	0.986	0.932	0.932	0.892	0.905
	'split4_train_recall':	1.000	1.000	0.973	0.904	0.918	0.904	0.904	0.945	0.890	0.918	0.904	0.918	0.904	0.890
	'mean_train_recall':	1.000	1.000	0.978	0.905	0.902	0.900	0.889	0.989	0.975	0.954	0.913	0.908	0.897	0.897
	'split0_test_precision':	0.655	0.655	0.655	0.720	0.750	0.750	0.750	0.655	0.655	0.643	0.750	0.750	0.750	0.750
	'split1_test_precision':	0.643	0.643	0.667	0.720	0.720	0.720	0.750	0.643	0.643	0.667	0.720	0.720	0.720	0.720
	'split2_test_precision':	0.643	0.643	0.640	0.636	0.636	0.667	0.667	0.654	0.654	0.652	0.636	0.667	0.667	0.667
	'split3_test_precision':	0.643	0.643	0.643	0.654	0.640	0.640	0.682	0.643	0.643	0.643	0.654	0.640	0.640	0.640
	'split4_test_precision':	0.679	0.679	0.680	0.727	0.750	0.714	0.667	0.708	0.696	0.727	0.727	0.714	0.714	0.714
	'mean_test_precision':	0.652	0.652	0.657	0.691	0.699	0.698	0.703	0.661	0.658	0.666	0.697	0.698	0.698	0.698
	'split0_train_precision':	0.652	0.652	0.652	0.677	0.698	0.708	0.691	0.652	0.661	0.667	0.690	0.708	0.708	0.708
	'split1_train_precision':	0.655	0.655	0.657	0.702	0.711	0.730	0.727	0.661	0.661	0.686	0.710	0.714	0.730	0.730
	'split2_train_precision':	0.655	0.655	0.664	0.742	0.722	0.739	0.739	0.655	0.667	0.726	0.725	0.722	0.722	0.739
	'split3_train_precision':	0.655	0.655	0.655	0.708	0.750	0.744	0.736	0.655	0.655	0.658	0.726	0.742	0.750	0.744
	'split4_train_precision':	0.646	0.646	0.651	0.710	0.713	0.717	0.725	0.657	0.663	0.691	0.710	0.713	0.717	0.714
	'mean_train_precision':	0.652	0.652	0.656	0.708	0.719	0.728	0.724	0.656	0.661	0.686	0.712	0.720	0.726	0.727
	'split0_test_f1':	0.792	0.792	0.792	0.818	0.837	0.837	0.837	0.792	0.792	0.766	0.837	0.837	0.837	0.837
	'split1_test_f1':	0.783	0.783	0.800	0.837	0.837	0.837	0.857	0.783	0.783	0.800	0.837	0.837	0.837	0.837
	'split2_test_f1':	0.783	0.783	0.744	0.700	0.700	0.718	0.718	0.773	0.773	0.732	0.700	0.718	0.718	0.718
	'split3_test_f1':	0.783	0.783	0.783	0.773	0.744	0.744	0.750	0.783	0.783	0.783	0.773	0.744	0.744	0.744
	'split4_test_f1':	0.809	0.809	0.773	0.780	0.769	0.750	0.700	0.791	0.762	0.780	0.780	0.750	0.750	0.750
	'mean_test_f1':	0.790	0.790	0.778	0.782	0.778	0.777	0.772	0.784	0.778	0.772	0.786	0.777	0.777	0.777
	'split0_train_f1':	0.789	0.789	0.789	0.779	0.793	0.805	0.778	0.789	0.791	0.796	0.798	0.805	0.805	0.805
	'split1_train_f1':	0.791	0.791	0.780	0.786	0.780	0.798	0.790	0.796	0.796	0.795	0.790	0.788	0.798	0.798
	'split2_train_f1':	0.791	0.791	0.785	0.810	0.793	0.802	0.802	0.791	0.800	0.817	0.800	0.793	0.793	0.802
	'split3_train_f1':	0.791	0.791	0.791	0.800	0.831	0.817	0.812	0.791	0.791	0.789	0.817	0.826	0.815	0.817
	'split4_train_f1':	0.785	0.785	0.780	0.795	0.802	0.800	0.805	0.775	0.760	0.788	0.795	0.802	0.800	0.793
	'mean_train_f1':	0.790	0.790	0.785	0.794	0.800	0.804	0.798	0.789	0.788	0.797	0.800	0.803	0.802	0.803
	'split0_test_accuracy':	0.655	0.655	0.655	0.724	0.759	0.759	0.759	0.655	0.655	0.621	0.759	0.759	0.759	0.759
	'split1_test_accuracy':	0.643	0.643	0.679	0.750	0.750	0.750	0.786	0.643	0.643	0.679	0.750	0.750	0.750	0.750
	'split2_test_accuracy':	0.643	0.643	0.607	0.571	0.571	0.607	0.607	0.643	0.643	0.607	0.571	0.607	0.607	0.607
	'split3_test_accuracy':	0.643	0.643	0.643	0.643	0.607	0.607	0.643	0.643	0.643	0.643	0.643	0.607	0.607	0.607

	'split4_test_accuracy':	0.679	0.679	0.643	0.679	0.679	0.643	0.571	0.679	0.643	0.679	0.679	0.643	0.643	0.643
	'mean_test_accuracy':	0.652	0.652	0.645	0.673	0.673	0.673	0.673	0.652	0.645	0.646	0.680	0.673	0.673	0.673
	'split0_train_accuracy':	0.652	0.652	0.652	0.661	0.688	0.705	0.670	0.652	0.661	0.670	0.688	0.705	0.705	0.705
	'split1_train_accuracy':	0.655	0.655	0.646	0.681	0.681	0.708	0.699	0.664	0.664	0.681	0.690	0.690	0.708	0.708
	'split2_train_accuracy':	0.655	0.655	0.655	0.726	0.699	0.717	0.717	0.655	0.673	0.726	0.708	0.699	0.699	0.717
	'split3_train_accuracy':	0.655	0.655	0.655	0.699	0.752	0.735	0.726	0.655	0.655	0.655	0.726	0.743	0.735	0.735
	'split4_train_accuracy':	0.646	0.646	0.646	0.699	0.708	0.708	0.717	0.646	0.637	0.681	0.699	0.708	0.708	0.699
	'mean_train_accuracy':	0.652	0.652	0.651	0.693	0.706	0.715	0.706	0.654	0.658	0.683	0.702	0.709	0.711	0.713
		l1							l2						
	LR_EURAUD_EURCAD	0.001	0.01	0.1	1	10	100	1000	0.001	0.01	0.1	1	10	100	1000
L1	'split0_test_recall':	0.000	0.000	0.000	0.733	0.733	0.733	0.733	1.000	1.000	0.800	0.733	0.733	0.733	0.733
	'split1_test_recall':	0.000	0.000	0.000	0.800	0.800	0.800	0.733	1.000	1.000	0.867	0.800	0.800	0.800	0.800
	'split2_test_recall':	0.000	0.000	0.000	0.800	0.800	0.800	0.867	1.000	1.000	0.867	0.800	0.800	0.800	0.800
	'split3_test_recall':	0.000	0.000	0.000	0.733	0.733	0.733	0.667	1.000	1.000	0.800	0.800	0.733	0.733	0.733
	'split4_test_recall':	0.000	0.000	0.000	0.600	0.600	0.600	0.667	1.000	1.000	0.667	0.667	0.600	0.600	0.600
	'mean_test_recall':	0.000	0.000	0.000	0.733	0.733	0.733	0.733	1.000	1.000	0.800	0.760	0.733	0.733	0.733
	'split0_train_recall':	0.000	0.000	0.000	0.783	0.800	0.800	0.750	1.000	1.000	0.833	0.800	0.800	0.800	0.783
	'split1_train_recall':	0.000	0.000	0.000	0.733	0.733	0.733	0.733	1.000	1.000	0.833	0.750	0.750	0.733	0.733
	'split2_train_recall':	0.000	0.000	0.000	0.750	0.750	0.750	0.733	1.000	1.000	0.800	0.767	0.750	0.750	0.750
	'split3_train_recall':	0.000	0.000	0.000	0.733	0.767	0.767	0.767	1.000	1.000	0.800	0.783	0.783	0.767	0.767
	'split4_train_recall':	0.000	0.000	0.000	0.717	0.733	0.733	0.700	1.000	1.000	0.817	0.733	0.733	0.733	0.733
	'mean_train_recall':	0.000	0.000	0.000	0.743	0.757	0.757	0.737	1.000	1.000	0.817	0.767	0.763	0.757	0.753
	'split0_test_precision':	0.000	0.000	0.000	0.023	0.025	0.025	0.022	0.517	0.517	0.571	0.647	0.647	0.647	0.647
	'split1_test_precision':	0.000	0.000	0.000	0.647	0.647	0.647	0.733	0.536	0.536	0.765	0.750	0.750	0.750	0.750
	'split2_test_precision':	0.000	0.000	0.000	0.750	0.750	0.750	0.733	0.536	0.536	0.650	0.667	0.667	0.667	0.667
	'split3_test_precision':	0.000	0.000	0.000	0.706	0.667	0.667	0.684	0.536	0.536	0.600	0.632	0.611	0.611	0.611
	'split4_test_precision':	0.000	0.000	0.000	0.611	0.611	0.611	0.625	0.536	0.536	0.556	0.588	0.563	0.563	0.563
	'mean_test_precision':	0.000	0.000	0.000	0.600	0.563	0.563	0.588	0.532	0.532	0.628	0.657	0.647	0.647	0.647
	'split0_train_precision':	0.000	0.000	0.000	0.663	0.647	0.647	0.673	0.536	0.536	0.641	0.658	0.658	0.667	0.662
	'split1_train_precision':	0.000	0.000	0.000	0.629	0.620	0.620	0.657	0.531	0.531	0.610	0.616	0.625	0.620	0.620
	'split2_train_precision':	0.000	0.000	0.000	0.662	0.652	0.652	0.667	0.531	0.531	0.632	0.648	0.652	0.652	0.652
	'split3_train_precision':	0.000	0.000	0.000	0.677	0.687	0.687	0.697	0.531	0.531	0.649	0.691	0.691	0.687	0.687
	'split4_train_precision':	0.000	0.000	0.000	0.652	0.667	0.667	0.689	0.531	0.531	0.662	0.657	0.667	0.667	0.667
	'mean_train_precision':	0.000	0.000	0.000	0.658	0.658	0.658	0.676	0.532	0.532	0.639	0.654	0.659	0.658	0.657
	'split0_test_f1':	0.000	0.000	0.000	0.688	0.688	0.688	0.733	0.682	0.682	0.667	0.688	0.688	0.688	0.688
	'split1_test_f1':	0.000	0.000	0.000	0.774	0.774	0.774	0.733	0.698	0.698	0.813	0.774	0.774	0.774	0.774
	'split2_test_f1':	0.000	0.000	0.000	0.750	0.727	0.727	0.765	0.698	0.698	0.743	0.727	0.727	0.727	0.727
	'split3_test_f1':	0.000	0.000	0.000	0.667	0.667	0.667	0.645	0.698	0.698	0.686	0.706	0.667	0.667	0.667
	'split4_test_f1':	0.000	0.000	0.000	0.600	0.581	0.581	0.625	0.698	0.698	0.606	0.625	0.581	0.581	0.581
	'mean_test_f1':	0.000	0.000	0.000	0.696	0.687	0.687	0.700	0.695	0.695	0.703	0.704	0.687	0.687	0.687
	'split0_train_f1':	0.000	0.000	0.000	0.723	0.727	0.727	0.709	0.698	0.698	0.725	0.722	0.722	0.727	0.718
	'split1_train_f1':	0.000	0.000	0.000	0.677	0.672	0.672	0.693	0.694	0.694	0.704	0.677	0.682	0.672	0.672
	'split2_train_f1':	0.000	0.000	0.000	0.703	0.698	0.698	0.698	0.694	0.694	0.706	0.702	0.698	0.698	0.698

	'split3_train_f1':	0.000	0.000	0.000	0.704	0.724	0.724	0.730	0.694	0.694	0.716	0.734	0.734	0.724	0.724
	'split4_train_f1':	0.000	0.000	0.000	0.683	0.698	0.698	0.694	0.694	0.694	0.731	0.693	0.698	0.698	0.698
	'mean_train_f1':	0.000	0.000	0.000	0.698	0.704	0.704	0.705	0.694	0.694	0.717	0.706	0.707	0.704	0.702
	'split0_test_accuracy':	0.483	0.483	0.483	0.655	0.655	0.655	0.724	0.517	0.517	0.586	0.655	0.655	0.655	0.655
	'split1_test_accuracy':	0.464	0.464	0.464	0.750	0.750	0.750	0.714	0.536	0.536	0.786	0.750	0.750	0.750	0.750
	'split2_test_accuracy':	0.464	0.464	0.464	0.714	0.679	0.679	0.714	0.536	0.536	0.679	0.679	0.679	0.679	0.679
	'split3_test_accuracy':	0.464	0.464	0.464	0.607	0.607	0.607	0.607	0.536	0.536	0.607	0.643	0.607	0.607	0.607
	'split4_test_accuracy':	0.464	0.464	0.464	0.571	0.536	0.536	0.571	0.536	0.536	0.536	0.571	0.536	0.536	0.536
	'mean_test_accuracy':	0.468	0.468	0.468	0.660	0.645	0.645	0.666	0.532	0.532	0.639	0.660	0.645	0.645	0.645
	'split0_train_accuracy':	0.464	0.464	0.464	0.679	0.679	0.679	0.670	0.536	0.536	0.661	0.670	0.670	0.679	0.670
	'split1_train_accuracy':	0.469	0.469	0.469	0.628	0.619	0.619	0.655	0.531	0.531	0.628	0.619	0.628	0.619	0.619
	'split2_train_accuracy':	0.469	0.469	0.469	0.664	0.655	0.655	0.664	0.531	0.531	0.646	0.655	0.655	0.655	0.655
	'split3_train_accuracy':	0.469	0.469	0.469	0.673	0.690	0.690	0.699	0.531	0.531	0.664	0.699	0.699	0.690	0.690
	'split4_train_accuracy':	0.469	0.469	0.469	0.646	0.664	0.664	0.673	0.531	0.531	0.681	0.655	0.664	0.664	0.664
	'mean_train_accuracy':	0.468	0.468	0.468	0.658	0.661	0.661	0.672	0.532	0.532	0.656	0.660	0.663	0.661	0.660
		l1							l2						
	LR_EURAUD_EURCAD	0.001	0.01	0.1	1	10	100	1000	0.001	0.01	0.1	1	10	100	1000
L2	'split0_test_recall':	0.000	0.000	0.000	0.600	0.533	0.533	0.667	1.000	1.000	0.667	0.533	0.533	0.533	0.533
	'split1_test_recall':	0.000	0.000	0.000	0.533	0.533	0.533	0.600	1.000	0.933	0.533	0.533	0.533	0.533	0.533
	'split2_test_recall':	0.000	0.000	0.000	0.643	0.643	0.643	0.357	1.000	1.000	0.643	0.643	0.643	0.643	0.643
	'split3_test_recall':	0.000	0.000	0.000	0.571	0.571	0.571	0.571	1.000	1.000	0.571	0.571	0.571	0.571	0.571
	'split4_test_recall':	0.000	0.000	0.000	0.714	0.714	0.714	0.429	1.000	1.000	0.857	0.714	0.714	0.714	0.714
	'mean_test_recall':	0.000	0.000	0.000	0.612	0.599	0.599	0.525	1.000	0.987	0.654	0.599	0.599	0.599	0.599
	'split0_train_recall':	0.000	0.000	0.000	0.614	0.596	0.596	0.649	1.000	1.000	0.614	0.596	0.596	0.596	0.596
	'split1_train_recall':	0.000	0.000	0.000	0.632	0.596	0.596	0.579	1.000	0.825	0.596	0.596	0.596	0.596	0.596
	'split2_train_recall':	0.000	0.000	0.000	0.603	0.603	0.586	0.603	1.000	1.000	0.707	0.603	0.586	0.586	0.603
	'split3_train_recall':	0.000	0.000	0.000	0.621	0.586	0.586	0.603	1.000	1.000	0.655	0.586	0.586	0.586	0.586
	'split4_train_recall':	0.000	0.000	0.000	0.586	0.586	0.569	0.638	1.000	1.000	0.655	0.586	0.569	0.569	0.586
	'mean_train_recall':	0.000	0.000	0.000	0.611	0.594	0.587	0.615	1.000	0.965	0.646	0.594	0.587	0.587	0.594
	'split0_test_precision':	0.000	0.000	0.000	0.563	0.571	0.571	0.526	0.517	0.517	0.588	0.571	0.571	0.571	0.571
	'split1_test_precision':	0.000	0.000	0.000	0.533	0.615	0.615	0.600	0.536	0.560	0.571	0.615	0.615	0.615	0.615
	'split2_test_precision':	0.000	0.000	0.000	0.600	0.643	0.643	0.556	0.500	0.500	0.563	0.643	0.643	0.643	0.643
	'split3_test_precision':	0.000	0.000	0.000	0.500	0.533	0.533	0.471	0.500	0.500	0.471	0.533	0.533	0.533	0.533
	'split4_test_precision':	0.000	0.000	0.000	0.588	0.588	0.588	0.500	0.500	0.500	0.571	0.588	0.588	0.588	0.588
	'mean_test_precision':	0.000	0.000	0.000	0.557	0.590	0.590	0.530	0.511	0.515	0.553	0.590	0.590	0.590	0.590
	'split0_train_precision':	0.000	0.000	0.000	0.547	0.586	0.596	0.617	0.509	0.509	0.547	0.586	0.596	0.596	0.586
	'split1_train_precision':	0.000	0.000	0.000	0.571	0.596	0.596	0.579	0.504	0.485	0.586	0.596	0.596	0.596	0.596
	'split2_train_precision':	0.000	0.000	0.000	0.538	0.547	0.548	0.593	0.513	0.513	0.532	0.538	0.548	0.548	0.556
	'split3_train_precision':	0.000	0.000	0.000	0.563	0.586	0.586	0.614	0.513	0.513	0.576	0.576	0.586	0.586	0.586
	'split4_train_precision':	0.000	0.000	0.000	0.540	0.557	0.559	0.627	0.513	0.513	0.528	0.548	0.559	0.559	0.567
	'mean_train_precision':	0.000	0.000	0.000	0.552	0.575	0.577	0.606	0.511	0.507	0.554	0.569	0.577	0.577	0.578
	'split0_test_f1':	0.000	0.000	0.000	0.581	0.552	0.552	0.588	0.682	0.682	0.625	0.552	0.552	0.552	0.552
	'split1_test_f1':	0.000	0.000	0.000	0.533	0.571	0.571	0.600	0.698	0.700	0.552	0.571	0.571	0.571	0.571

	'split2_test_f1':	0.000	0.000	0.000	0.621	0.643	0.643	0.435	0.667	0.667	0.600	0.643	0.643	0.643	0.643
	'split3_test_f1':	0.000	0.000	0.000	0.533	0.552	0.552	0.516	0.667	0.667	0.516	0.552	0.552	0.552	0.552
	'split4_test_f1':	0.000	0.000	0.000	0.645	0.645	0.645	0.462	0.667	0.667	0.686	0.645	0.645	0.645	0.645
	'mean_test_f1':	0.000	0.000	0.000	0.583	0.593	0.593	0.520	0.676	0.676	0.596	0.593	0.593	0.593	0.593
	'split0_train_f1':	0.000	0.000	0.000	0.579	0.591	0.596	0.632	0.675	0.675	0.579	0.591	0.596	0.596	0.591
	'split1_train_f1':	0.000	0.000	0.000	0.600	0.596	0.596	0.579	0.671	0.610	0.591	0.596	0.596	0.596	0.596
	'split2_train_f1':	0.000	0.000	0.000	0.569	0.574	0.567	0.598	0.678	0.678	0.607	0.569	0.567	0.567	0.579
	'split3_train_f1':	0.000	0.000	0.000	0.590	0.586	0.586	0.609	0.678	0.678	0.613	0.581	0.586	0.586	0.586
	'split4_train_f1':	0.000	0.000	0.000	0.562	0.571	0.564	0.632	0.678	0.678	0.585	0.567	0.564	0.564	0.576
	'mean_train_f1':	0.000	0.000	0.000	0.580	0.584	0.582	0.610	0.676	0.664	0.595	0.581	0.582	0.582	0.586
	'split0_test_accuracy':	0.483	0.483	0.483	0.552	0.552	0.552	0.517	0.517	0.517	0.586	0.552	0.552	0.552	0.552
	'split1_test_accuracy':	0.464	0.464	0.464	0.500	0.571	0.571	0.571	0.536	0.571	0.536	0.571	0.571	0.571	0.571
	'split2_test_accuracy':	0.500	0.500	0.500	0.607	0.643	0.643	0.536	0.500	0.500	0.571	0.643	0.643	0.643	0.643
	'split3_test_accuracy':	0.500	0.500	0.500	0.500	0.536	0.536	0.464	0.500	0.500	0.464	0.536	0.536	0.536	0.536
	'split4_test_accuracy':	0.500	0.500	0.500	0.607	0.607	0.607	0.500	0.500	0.500	0.607	0.607	0.607	0.607	0.607
	'mean_test_accuracy':	0.489	0.489	0.489	0.553	0.582	0.582	0.518	0.511	0.518	0.553	0.582	0.582	0.582	0.582
	'split0_train_accuracy':	0.491	0.491	0.491	0.545	0.580	0.589	0.616	0.509	0.509	0.545	0.580	0.589	0.589	0.580
	'split1_train_accuracy':	0.496	0.496	0.496	0.575	0.593	0.593	0.575	0.504	0.469	0.584	0.593	0.593	0.593	0.593
	'split2_train_accuracy':	0.487	0.487	0.487	0.531	0.540	0.540	0.584	0.513	0.513	0.531	0.531	0.540	0.540	0.549
	'split3_train_accuracy':	0.487	0.487	0.487	0.558	0.575	0.575	0.602	0.513	0.513	0.575	0.566	0.575	0.575	0.575
	'split4_train_accuracy':	0.487	0.487	0.487	0.531	0.549	0.549	0.619	0.513	0.513	0.522	0.540	0.549	0.549	0.558
	'mean_train_accuracy':	0.489	0.489	0.489	0.548	0.567	0.569	0.599	0.511	0.504	0.551	0.562	0.569	0.569	0.571

Table B- 15 GridSearchCV result of LR Total_Profit EURGBPEURCAD

		l1							l2						
	EURGBPEURCAD	0.001	0.01	0.1	1	10	100	1000	0.001	0.01	0.1	1	10	100	1000
Total	'split0_test_recall':	0.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split1_test_recall':	0.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split2_test_recall':	0.000	0.000	1.000	1.000	1.000	1.000	0.857	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split3_test_recall':	0.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split4_test_recall':	0.000	0.000	1.000	1.000	1.000	1.000	0.950	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'mean_test_recall':	0.000	0.000	1.000	1.000	1.000	1.000	0.961	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split0_train_recall':	0.000	0.000	1.000	1.000	1.000	1.000	0.988	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split1_train_recall':	0.000	0.000	1.000	1.000	1.000	1.000	0.964	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split2_train_recall':	0.000	0.000	1.000	1.000	1.000	1.000	0.904	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split3_train_recall':	0.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split4_train_recall':	0.000	0.000	1.000	1.000	1.000	1.000	0.952	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'mean_train_recall':	0.000	0.000	1.000	1.000	1.000	1.000	0.962	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	'split0_test_precision':	0.000	0.000	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583
	'split1_test_precision':	0.000	0.000	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583
	'split2_test_precision':	0.000	0.000	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600
	'split3_test_precision':	0.000	0.000	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600
	'split4_test_precision':	0.000	0.000	0.571	0.571	0.571	0.571	0.613	0.571	0.571	0.571	0.571	0.571	0.571	0.571
	'mean_test_precision':	0.000	0.000	0.588	0.588	0.588	0.588	0.596	0.588	0.588	0.588	0.588	0.588	0.588	0.588
	'split0_train_precision':	0.000	0.000	0.589	0.589	0.589	0.589	0.586	0.589	0.589	0.589	0.589	0.589	0.589	0.589
	'split1_train_precision':	0.000	0.000	0.589	0.589	0.589	0.589	0.602	0.589	0.589	0.589	0.589	0.589	0.589	0.589
	'split2_train_precision':	0.000	0.000	0.585	0.585	0.585	0.585	0.600	0.585	0.585	0.585	0.585	0.585	0.585	0.585
	'split3_train_precision':	0.000	0.000	0.585	0.585	0.585	0.585	0.585	0.585	0.585	0.585	0.585	0.585	0.585	0.585
	'split4_train_precision':	0.000	0.000	0.592	0.592	0.592	0.592	0.597	0.592	0.592	0.592	0.592	0.592	0.592	0.592
	'mean_train_precision':	0.000	0.000	0.588	0.588	0.588	0.588	0.594	0.588	0.588	0.588	0.588	0.588	0.588	0.588
	'split0_test_f1':	0.000	0.000	0.737	0.737	0.737	0.737	0.737	0.737	0.737	0.737	0.737	0.737	0.737	0.737
	'split1_test_f1':	0.000	0.000	0.737	0.737	0.737	0.737	0.737	0.737	0.737	0.737	0.737	0.737	0.737	0.737
	'split2_test_f1':	0.000	0.000	0.750	0.750	0.750	0.750	0.706	0.750	0.750	0.750	0.750	0.750	0.750	0.750
	'split3_test_f1':	0.000	0.000	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750
	'split4_test_f1':	0.000	0.000	0.727	0.727	0.727	0.727	0.745	0.727	0.727	0.727	0.727	0.727	0.727	0.727
	'mean_test_f1':	0.000	0.000	0.740	0.740	0.740	0.740	0.735	0.740	0.740	0.740	0.740	0.740	0.740	0.740
	'split0_train_f1':	0.000	0.000	0.741	0.741	0.741	0.741	0.735	0.741	0.741	0.741	0.741	0.741	0.741	0.741
	'split1_train_f1':	0.000	0.000	0.741	0.741	0.741	0.741	0.741	0.741	0.741	0.741	0.741	0.741	0.741	0.741
	'split2_train_f1':	0.000	0.000	0.738	0.738	0.738	0.738	0.721	0.738	0.738	0.738	0.738	0.738	0.738	0.738
	'split3_train_f1':	0.000	0.000	0.738	0.738	0.738	0.738	0.738	0.738	0.738	0.738	0.738	0.738	0.738	0.738
	'split4_train_f1':	0.000	0.000	0.743	0.743	0.743	0.743	0.734	0.743	0.743	0.743	0.743	0.743	0.743	0.743
	'mean_train_f1':	0.000	0.000	0.740	0.740	0.740	0.740	0.734	0.740	0.740	0.740	0.740	0.740	0.740	0.740
	'split0_test_accuracy':	0.417	0.417	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583
	'split1_test_accuracy':	0.417	0.417	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583	0.583
	'split2_test_accuracy':	0.400	0.400	0.600	0.600	0.600	0.600	0.571	0.600	0.600	0.600	0.600	0.600	0.600	0.600
	'split3_test_accuracy':	0.400	0.400	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600	0.600

	'split4_test_accuracy':	0.429	0.429	0.571	0.571	0.571	0.571	0.629	0.571	0.571	0.571	0.571	0.571	0.571	0.571
	'mean_test_accuracy':	0.412	0.412	0.588	0.588	0.588	0.588	0.593	0.588	0.588	0.588	0.588	0.588	0.588	0.588
	'split0_train_accuracy':	0.411	0.411	0.589	0.589	0.589	0.589	0.582	0.589	0.589	0.589	0.589	0.589	0.589	0.589
	'split1_train_accuracy':	0.411	0.411	0.589	0.589	0.589	0.589	0.603	0.589	0.589	0.589	0.589	0.589	0.589	0.589
	'split2_train_accuracy':	0.415	0.415	0.585	0.585	0.585	0.585	0.592	0.585	0.585	0.585	0.585	0.585	0.585	0.585
	'split3_train_accuracy':	0.415	0.415	0.585	0.585	0.585	0.585	0.585	0.585	0.585	0.585	0.585	0.585	0.585	0.585
	'split4_train_accuracy':	0.408	0.408	0.592	0.592	0.592	0.592	0.592	0.592	0.592	0.592	0.592	0.592	0.592	0.592
	'mean_train_accuracy':	0.412	0.412	0.588	0.588	0.588	0.588	0.590	0.588	0.588	0.588	0.588	0.588	0.588	0.588

Table B- 16 GridSearchCV result of LR EURGBPEURUSD

		l1							l2						
	EURGBPEURUSD	0.001	0.01	0.1	1	10	100	1000	0.001	0.01	0.1	1	10	100	1000
Total	'split0_test_recall':	0.000	0.000	0.000	0.643	0.643	0.571	0.643	0.714	0.714	0.786	0.714	0.643	0.643	0.643
	'split1_test_recall':	0.000	0.000	0.000	0.462	0.462	0.769	0.923	0.462	0.462	0.462	0.462	0.462	0.462	0.462
	'split2_test_recall':	0.000	0.000	0.000	0.615	0.692	0.692	0.538	0.538	0.538	0.615	0.615	0.692	0.692	0.692
	'split3_test_recall':	0.000	0.000	0.000	0.786	0.786	0.714	0.786	0.643	0.643	0.714	0.786	0.786	0.786	0.786
	'split4_test_recall':	0.000	0.000	0.000	0.857	0.857	0.714	0.643	0.714	0.714	0.786	0.786	0.857	0.857	0.786
	'mean_test_recall':	0.000	0.000	0.000	0.673	0.688	0.692	0.707	0.614	0.614	0.673	0.673	0.688	0.688	0.674
	'split0_train_recall':	0.000	0.000	0.000	0.722	0.722	0.722	0.722	0.593	0.611	0.648	0.722	0.722	0.722	0.722
	'split1_train_recall':	0.000	0.000	0.000	0.691	0.691	0.745	0.745	0.655	0.655	0.691	0.691	0.691	0.691	0.709
	'split2_train_recall':	0.000	0.000	0.000	0.709	0.709	0.745	0.727	0.636	0.636	0.691	0.673	0.709	0.709	0.709
	'split3_train_recall':	0.000	0.000	0.000	0.685	0.722	0.704	0.648	0.611	0.611	0.648	0.685	0.704	0.704	0.722
	'split4_train_recall':	0.000	0.000	0.000	0.704	0.704	0.722	0.704	0.593	0.593	0.667	0.741	0.704	0.704	0.722
	'mean_train_recall':	0.000	0.000	0.000	0.702	0.710	0.728	0.709	0.617	0.621	0.669	0.702	0.706	0.706	0.717
	'split0_test_precision':	0.000	0.000	0.000	0.692	0.692	0.727	0.750	0.556	0.588	0.647	0.714	0.692	0.692	0.692
	'split1_test_precision':	0.000	0.000	0.000	0.462	0.462	0.556	0.632	0.500	0.500	0.462	0.500	0.462	0.462	0.462
	'split2_test_precision':	0.000	0.000	0.000	0.667	0.692	0.643	0.700	0.500	0.500	0.615	0.667	0.692	0.692	0.643
	'split3_test_precision':	0.000	0.000	0.000	0.733	0.688	0.769	0.733	0.643	0.643	0.769	0.688	0.688	0.688	0.688
	'split4_test_precision':	0.000	0.000	0.000	0.600	0.632	0.667	0.643	0.588	0.588	0.647	0.647	0.632	0.632	0.611
	'mean_test_precision':	0.000	0.000	0.000	0.631	0.633	0.672	0.692	0.557	0.564	0.628	0.643	0.633	0.633	0.619
	'split0_train_precision':	0.000	0.000	0.000	0.661	0.661	0.661	0.661	0.552	0.559	0.614	0.650	0.650	0.650	0.639
	'split1_train_precision':	0.000	0.000	0.000	0.655	0.667	0.683	0.695	0.590	0.590	0.633	0.691	0.667	0.667	0.684
	'split2_train_precision':	0.000	0.000	0.000	0.650	0.650	0.707	0.702	0.574	0.574	0.613	0.617	0.629	0.639	0.639
	'split3_train_precision':	0.000	0.000	0.000	0.638	0.650	0.667	0.686	0.550	0.550	0.583	0.627	0.633	0.644	0.639
	'split4_train_precision':	0.000	0.000	0.000	0.667	0.655	0.672	0.679	0.561	0.552	0.610	0.667	0.644	0.644	0.650
	'mean_train_precision':	0.000	0.000	0.000	0.654	0.657	0.678	0.685	0.565	0.565	0.611	0.650	0.645	0.649	0.650
	'split0_test_f1':	0.000	0.000	0.000	0.667	0.667	0.640	0.62	0.625	0.645	0.710	0.714	0.667	0.667	0.667
	'split1_test_f1':	0.000	0.000	0.000	0.462	0.462	0.645	0.730	0.480	0.480	0.462	0.480	0.462	0.462	0.462
	'split2_test_f1':	0.000	0.000	0.000	0.640	0.692	0.667	0.589	0.519	0.519	0.615	0.640	0.692	0.692	0.667
	'split3_test_f1':	0.000	0.000	0.000	0.759	0.733	0.741	0.739	0.643	0.643	0.741	0.733	0.733	0.733	0.733
	'split4_test_f1':	0.000	0.000	0.000	0.706	0.727	0.690	0.623	0.645	0.645	0.710	0.710	0.727	0.727	0.688
	'mean_test_f1':	0.000	0.000	0.000	0.647	0.656	0.676	0.670	0.582	0.586	0.647	0.655	0.656	0.656	0.643
	'split0_train_f1':	0.000	0.000	0.000	0.690	0.690	0.690	0.690	0.571	0.584	0.631	0.684	0.684	0.684	0.678
	'split1_train_f1':	0.000	0.000	0.000	0.673	0.679	0.713	0.719	0.621	0.621	0.661	0.691	0.679	0.679	0.696
	'split2_train_f1':	0.000	0.000	0.000	0.678	0.678	0.726	0.714	0.603	0.603	0.650	0.643	0.667	0.672	0.672
	'split3_train_f1':	0.000	0.000	0.000	0.661	0.684	0.685	0.667	0.579	0.579	0.614	0.655	0.667	0.673	0.678
	'split4_train_f1':	0.000	0.000	0.000	0.685	0.679	0.696	0.691	0.577	0.571	0.637	0.702	0.673	0.673	0.684
	'mean_train_f1':	0.000	0.000	0.000	0.677	0.682	0.702	0.696	0.590	0.592	0.638	0.675	0.674	0.676	0.682
	'split0_test_accuracy':	0.481	0.481	0.481	0.667	0.667	0.667	0.704	0.556	0.593	0.667	0.704	0.667	0.667	0.667
	'split1_test_accuracy':	0.500	0.500	0.500	0.462	0.462	0.577	0.692	0.500	0.500	0.462	0.500	0.462	0.462	0.462
	'split2_test_accuracy':	0.500	0.500	0.500	0.654	0.692	0.654	0.654	0.500	0.500	0.615	0.654	0.692	0.692	0.654
	'split3_test_accuracy':	0.462	0.462	0.462	0.731	0.692	0.731	0.731	0.615	0.615	0.731	0.692	0.692	0.692	0.692

	'split4_test_accuracy':	0.462	0.462	0.462	0.615	0.654	0.654	0.615	0.577	0.577	0.654	0.654	0.654	0.654	0.615
	'mean_test_accuracy':	0.481	0.481	0.481	0.626	0.633	0.656	0.679	0.550	0.557	0.626	0.641	0.633	0.633	0.618
	'split0_train_accuracy':	0.481	0.481	0.481	0.663	0.663	0.663	0.663	0.538	0.548	0.606	0.654	0.654	0.654	0.644
	'split1_train_accuracy':	0.476	0.476	0.476	0.648	0.657	0.686	0.695	0.581	0.581	0.629	0.676	0.657	0.657	0.676
	'split2_train_accuracy':	0.476	0.476	0.476	0.648	0.648	0.705	0.695	0.562	0.562	0.610	0.610	0.629	0.638	0.638
	'split3_train_accuracy':	0.486	0.486	0.486	0.638	0.657	0.667	0.667	0.543	0.543	0.581	0.629	0.638	0.648	0.648
	'split4_train_accuracy':	0.486	0.486	0.486	0.667	0.657	0.676	0.676	0.552	0.543	0.610	0.676	0.648	0.648	0.657
	'mean_train_accuracy':	0.481	0.481	0.481	0.653	0.657	0.679	0.679	0.555	0.555	0.607	0.649	0.645	0.649	0.653

6.4 APPENDIX C PROFIT RECORD

Table C- 17 Profit record from ANN_EURAUDEURCAD_Total_Leg

	Profit	y_pred	y_filter	Profit_w_ML	Cumulative_profit	Cumulative_profit _w_ML
0	-38.27	0.999998	0.999998	-38.2699235	-38.27	-38.26992346
1	380.1	1	1	380.1	341.83	-38.26992346
2	501.48	0.477751	0	0	843.31	-190.9099235
3	-152.64	1	1	-152.64	690.67	-190.9099235
4	-239.81	0.477751	0	0	450.86	102.4900765
5	293.4	1	1	293.4	744.26	1162.670077
6	1060.18	1	1	1060.18	1804.44	1207.900077
7	45.23	1	1	45.23	1849.67	1216.630077
8	8.73	1	1	8.73	1858.4	1347.290077
9	130.66	1	1	130.66	1989.06	1394.470077
10	47.18	1	1	47.18	2036.24	1094.800077
11	-299.67	1	1	-299.67	1736.57	1094.800077
12	-338.82	0.477751	0	0	1397.75	595.6800765
13	-499.12	1	1	-499.12	898.63	595.6800765
14	-156.04	0.477751	0	0	742.59	509.8100765
15	-85.87	1	1	-85.87	656.72	672.0900765
16	162.28	1	1	162.28	819	672.0900765
17	182.97	0.477751	0	0	1001.97	500.424404
18	-292.75	0.58639	0.58639	-171.665673	709.22	500.424404
19	-787.08	0.477751	0	0	-77.86	500.424404
20	-1073.39	0.477751	0	0	-1151.25	700.844404
21	200.42	1	1	200.42	-950.83	603.594404
22	-97.25	1	1	-97.25	-1048.08	603.594404
23	14.84	0.477751	0	0	-1033.24	603.594404
24	-172.73	0.477751	0	0	-1205.97	603.594404
25	-425.83	0.477751	0	0	-1631.8	700.174404
26	96.58	1	1	96.58	-1535.22	977.744404
27	277.57	1	1	277.57	-1257.65	1515.894404
28	538.15	1	1	538.15	-719.5	1515.894404
29	-631.55	0.477751	0	0	-1351.05	1515.894404
30	22.15	0.477751	0	0	-1328.9	1515.894404
31	423.06	0.477751	0	0	-905.84	1201.004404
32	-314.89	1	1	-314.89	-1220.73	1065.474404
33	-135.53	1	1	-135.53	-1356.26	1432.354404
34	366.88	1	1	366.88	-989.38	705.7257406
35	-930.61	0.780809	0.780809	-726.628663	-1919.99	1058.165036
36	352.44	0.999998	0.999998	352.439295	-1567.55	835.8850357
37	-222.28	1	1	-222.28	-1789.83	835.8850357
38	-187.86	0.477751	0	0	-1977.69	1032.615036

39	196.73	1	1	196.73	-1780.96	1032.615036
40	17.87	0.477751	0	0	-1763.09	557.8750357
41	-474.74	1	1	-474.74	-2237.83	456.7650357
42	-101.11	1	1	-101.11	-2338.94	761.2650357
43	304.5	1	1	304.5	-2034.44	761.2650357
44	346.79	0.477751	0	0	-1687.65	371.4850357
45	-389.78	1	1	-389.78	-2077.43	740.5150357
46	369.03	1	1	369.03	-1708.4	420.4750357
47	-320.04	1	1	-320.04	-2028.44	420.4750357
48	-277.36	0.477751	0	0	-2305.8	684.5350357
49	264.06	1	1	264.06	-2041.74	2500.355036
50	1815.82	1	1	1815.82	-225.92	2500.355036
51	463.37	0.477751	0	0	237.45	2500.355036
52	-284.98	0.477751	0	0	-47.53	2781.415036
53	281.06	1	1	281.06	233.53	2832.575036
54	51.16	1	1	51.16	284.69	2832.575036
55	430.56	0.477751	0	0	715.25	2786.435036
56	-46.14	1	1	-46.14	669.11	3439.015036
57	652.58	1	1	652.58	1321.69	3629.675036
58	190.66	1	1	190.66	1512.35	3641.305036
59	11.63	1	1	11.63	1523.98	3641.305036
60	492.78	0.477751	0	0	2016.76	3030.805036
61	-610.5	1	1	-610.5	1406.26	2794.925036
62	-235.88	1	1	-235.88	1170.38	3036.465036
63	241.54	1	1	241.54	1411.92	2831.044034
64	-294.75	0.696933	0.696933	-205.421002	1117.17	3026.794034
65	195.75	1	1	195.75	1312.92	3026.794034
66	230.44	0.477751	0	0	1543.36	3200.294034
67	173.5	1	1	173.5	1716.86	3507.964034
68	307.67	1	1	307.67	2024.53	3018.624034
69	-489.34	1	1	-489.34	1535.19	2910.194034
70	-108.43	1	1	-108.43	1426.76	3320.264034
71	410.07	1	1	410.07	1836.83	3380.044034
72	59.78	1	1	59.78	1896.61	3422.894034
73	42.85	1	1	42.85	1939.46	3496.034034
74	73.14	1	1	73.14	2012.6	3327.824034
75	-168.21	1	1	-168.21	1844.39	3327.824034
76	397.9	0.477751	0	0	2242.29	3646.714034
77	318.89	1	1	318.89	2561.18	3494.704034
78	-152.01	1	1	-152.01	2409.17	3494.704034
79	-1369.58	0.477751	0	0	1039.59	3682.244034
80	187.54	1	1	187.54	1227.13	3682.244034
81	301.67	0.477751	0	0	1528.8	3485.654034
82	-196.59	1	1	-196.59	1332.21	3485.654034

83	-176.96	0.477751	0	0	1155.25	3743.944034
84	258.29	1	1	258.29	1413.54	3907.134034
85	163.19	1	1	163.19	1576.73	4245.814034
86	338.68	1	1	338.68	1915.41	4245.814034
87	-199.66	0.477751	0	0	1715.75	4245.814034
88	-790.39	0.477751	0	0	925.36	4266.744034
89	20.93	1	1	20.93	946.29	4500.344034
90	233.6	1	1	233.6	1179.89	4716.964034
91	216.62	1	1	216.62	1396.51	4072.434034
92	-644.53	1	1	-644.53	751.98	3400.744034
93	-671.69	1	1	-671.69	80.29	3455.104034
94	54.36	1	1	54.36	134.65	3405.814034
95	-49.29	1	1	-49.29	85.36	3637.624034
96	231.81	1	1	231.81	317.17	3637.624034

Table C- 18 Profit record of ANN_EURGBPEURUSD_Total_Leg

	Profit	y_pred	y_filter	Profit_w_ML	Cumulative_profit	Cumulative_profit _w_ML
0	-2770.88	1.00	1.00	-2770.88	-2770.88	-2770.88
1	673.46	1.00	1.00	673.46	-2097.42	-2097.42
2	-774.39	1.00	1.00	-774.39	-2871.81	-2871.81
3	-250.37	1.00	1.00	-250.37	-3122.18	-3122.18
4	-426.15	0.44	0.00	0.00	-3548.33	-3122.18
5	780.07	1.00	1.00	780.07	-2768.26	-2342.11
6	37.77	1.00	1.00	37.77	-2730.49	-2304.34
7	416.05	1.00	1.00	416.05	-2314.44	-1888.29
8	308.83	1.00	1.00	308.83	-2005.61	-1579.46
9	78.51	1.00	1.00	78.47	-1927.10	-1500.99
10	-182.00	1.00	1.00	-182.00	-2109.10	-1682.99
11	-1529.49	1.00	1.00	-1529.49	-3638.59	-3212.48
12	-17.34	1.00	1.00	-17.34	-3655.93	-3229.82
13	867.75	1.00	1.00	867.75	-2788.18	-2362.07
14	-294.45	1.00	1.00	-294.45	-3082.63	-2656.52
15	-1145.39	1.00	1.00	-1145.37	-4228.02	-3801.88
16	-1010.24	0.44	0.00	0.00	-5238.26	-3801.88
17	1031.62	1.00	1.00	1031.62	-4206.64	-2770.26
18	854.78	0.44	0.00	0.00	-3351.86	-2770.26
19	435.81	1.00	1.00	435.81	-2916.05	-2334.45
20	210.71	1.00	1.00	210.71	-2705.34	-2123.74
21	-889.27	0.99	0.99	-881.45	-3594.61	-3005.19
22	-314.03	1.00	1.00	-314.03	-3908.64	-3319.22
23	-210.97	1.00	1.00	-210.97	-4119.61	-3530.19
24	287.07	1.00	1.00	287.07	-3832.54	-3243.12
25	214.53	1.00	1.00	214.47	-3618.01	-3028.65
26	48.16	1.00	1.00	48.16	-3569.85	-2980.49
27	537.17	0.70	0.70	378.43	-3032.68	-2602.07
28	-503.15	1.00	1.00	-503.15	-3535.83	-3105.22
29	239.59	1.00	1.00	239.59	-3296.24	-2865.63
30	-536.76	1.00	1.00	-536.76	-3833.00	-3402.39
31	-1940.98	1.00	1.00	-1940.98	-5773.98	-5343.37
32	2626.63	1.00	1.00	2626.63	-3147.35	-2716.74
33	-2293.17	1.00	1.00	-2293.17	-5440.52	-5009.91
34	940.10	1.00	1.00	940.10	-4500.42	-4069.81
35	14.56	1.00	1.00	14.56	-4485.86	-4055.25
36	-1438.31	1.00	1.00	-1438.31	-5924.17	-5493.56
37	-1709.08	1.00	1.00	-1709.08	-7633.25	-7202.64
38	257.59	1.00	1.00	257.59	-7375.66	-6945.05
39	816.69	1.00	1.00	816.69	-6558.97	-6128.36
40	525.01	0.44	0.00	0.00	-6033.96	-6128.36

41	400.61	1.00	1.00	400.61	-5633.35	-5727.75
42	770.99	1.00	1.00	770.99	-4862.36	-4956.76
43	-154.16	1.00	1.00	-154.16	-5016.52	-5110.92
44	162.00	1.00	1.00	162.00	-4854.52	-4948.92
45	-288.09	1.00	1.00	-288.09	-5142.61	-5237.01
46	311.57	1.00	1.00	311.57	-4831.04	-4925.44
47	52.38	0.44	0.00	0.00	-4778.66	-4925.44
48	298.39	1.00	1.00	298.39	-4480.27	-4627.05
49	-680.25	1.00	1.00	-680.25	-5160.52	-5307.30
50	529.45	1.00	1.00	529.45	-4631.07	-4777.85
51	77.11	0.98	0.98	75.81	-4553.96	-4702.03
52	-422.63	1.00	1.00	-422.63	-4976.59	-5124.66
53	300.54	0.44	0.00	0.00	-4676.05	-5124.66
54	357.05	1.00	1.00	357.05	-4319.00	-4767.61
55	-871.06	1.00	1.00	-871.06	-5190.06	-5638.67
56	-1898.16	0.44	0.00	0.00	-7088.22	-5638.67
57	-595.43	0.44	0.00	0.00	-7683.65	-5638.67
58	-67.31	0.97	0.97	-65.29	-7750.96	-5703.96
59	359.89	1.00	1.00	359.89	-7391.07	-5344.07
60	-1558.61	1.00	1.00	-1558.61	-8949.68	-6902.68
61	293.00	1.00	1.00	293.00	-8656.68	-6609.68
62	-804.40	1.00	1.00	-804.40	-9461.08	-7414.08
63	391.11	1.00	1.00	391.11	-9069.97	-7022.97
64	-195.86	1.00	1.00	-195.86	-9265.83	-7218.83
65	197.11	1.00	1.00	197.11	-9068.72	-7021.72
66	-1025.32	0.99	0.99	-1014.63	-10094.04	-8036.35
67	-572.25	1.00	1.00	-572.25	-10666.29	-8608.60
68	1134.86	1.00	1.00	1134.84	-9531.43	-7473.76
69	499.34	1.00	1.00	499.34	-9032.09	-6974.42
70	456.77	1.00	1.00	456.77	-8575.32	-6517.65
71	-489.94	1.00	1.00	-489.94	-9065.26	-7007.59
72	310.95	0.86	0.86	266.67	-8754.31	-6740.92
73	765.94	1.00	1.00	765.94	-7988.37	-5974.98
74	63.85	1.00	1.00	63.85	-7924.52	-5911.13
75	-305.99	1.00	1.00	-305.99	-8230.51	-6217.12
76	137.67	1.00	1.00	137.67	-8092.84	-6079.45
77	181.10	1.00	1.00	181.07	-7911.74	-5898.38
78	81.96	1.00	1.00	81.96	-7829.78	-5816.42
79	-384.98	0.61	0.61	-235.12	-8214.76	-6051.54
80	-1001.36	1.00	1.00	-1001.36	-9216.12	-7052.90
81	238.19	1.00	1.00	238.19	-8977.93	-6814.71
82	-439.96	1.00	1.00	-439.96	-9417.89	-7254.67
83	-391.62	1.00	1.00	-391.62	-9809.51	-7646.29
84	656.62	1.00	1.00	656.62	-9152.89	-6989.67

85	-130.65	0.44	0.00	0.00	-9283.54	-6989.67
86	-458.31	1.00	1.00	-458.31	-9741.85	-7447.98
87	268.99	1.00	1.00	268.99	-9472.86	-7178.99
88	412.04	1.00	1.00	412.04	-9060.82	-6766.95
89	-390.10	1.00	1.00	-390.10	-9450.92	-7157.05
90	125.84	1.00	1.00	125.84	-9325.08	-7031.21
91	-319.37	1.00	1.00	-319.37	-9644.45	-7350.57
92	-783.60	1.00	1.00	-783.60	-10428.05	-8134.17
93	204.89	1.00	1.00	204.85	-10223.16	-7929.33
94	-442.69	1.00	1.00	-442.69	-10665.85	-8372.02
95	-510.17	1.00	1.00	-510.17	-11176.02	-8882.19
96	171.62	1.00	1.00	171.62	-11004.40	-8710.57
97	654.48	1.00	1.00	654.48	-10349.92	-8056.09
98	318.68	1.00	1.00	318.68	-10031.24	-7737.41
99	166.81	0.44	0.00	0.00	-9864.43	-7737.41
100	-351.72	1.00	1.00	-351.72	-10216.15	-8089.13
101	332.14	0.68	0.68	226.58	-9884.01	-7862.55
102	40.45	1.00	1.00	40.45	-9843.56	-7822.10
103	1018.30	1.00	1.00	1018.30	-8825.26	-6803.80
104	-20.15	0.99	0.99	-19.94	-8845.41	-6823.74
105	124.16	1.00	1.00	124.12	-8721.25	-6699.62
106	162.22	1.00	1.00	162.22	-8559.03	-6537.40
107	-554.31	1.00	1.00	-554.31	-9113.34	-7091.71
108	125.64	1.00	1.00	125.64	-8987.70	-6966.07
109	288.83	0.88	0.88	253.48	-8698.87	-6712.59
110	-1438.39	0.44	0.00	0.00	-10137.26	-6712.59
111	102.10	0.95	0.95	96.90	-10035.16	-6615.69
112	311.30	0.44	0.00	0.00	-9723.86	-6615.69
113	-1620.93	0.44	0.00	0.00	-11344.79	-6615.69
114	1196.10	1.00	1.00	1196.10	-10148.69	-5419.59
115	265.22	1.00	1.00	265.22	-9883.47	-5154.37
116	829.47	0.44	0.00	0.00	-9054.00	-5154.37
117	-894.45	0.44	0.00	0.00	-9948.45	-5154.37
118	80.77	0.69	0.69	56.11	-9867.68	-5098.26
119	480.72	1.00	1.00	480.72	-9386.96	-4617.54
120	388.37	1.00	1.00	388.37	-8998.59	-4229.17
121	553.89	1.00	1.00	553.89	-8444.70	-3675.28
122	38.40	1.00	1.00	38.40	-8406.30	-3636.88
123	-727.79	0.44	0.00	0.00	-9134.09	-3636.88
124	-50.43	1.00	1.00	-50.43	-9184.52	-3687.31
125	419.06	0.46	0.00	0.00	-8765.46	-3687.31
126	-1211.85	1.00	1.00	-1211.85	-9977.31	-4899.16
127	-1288.50	1.00	1.00	-1288.50	-11265.81	-6187.66
128	948.11	0.99	0.99	942.72	-10317.70	-5244.94

129	207.32	1.00	1.00	207.32	-10110.38	-5037.63
130	750.16	0.44	0.00	0.00	-9360.22	-5037.63
131	618.50	0.67	0.67	411.90	-8741.72	-4625.73
132	312.36	0.44	0.00	0.00	-8429.36	-4625.73
133	-1027.22	0.44	0.00	0.00	-9456.58	-4625.73
134	-467.16	1.00	1.00	-467.16	-9923.74	-5092.89
135	181.52	0.44	0.00	0.00	-9742.22	-5092.89
136	239.29	1.00	1.00	239.29	-9502.93	-4853.60
137	247.81	1.00	1.00	247.81	-9255.12	-4605.79
138	58.63	1.00	1.00	58.63	-9196.49	-4547.16
139	-186.14	1.00	1.00	-186.12	-9382.63	-4733.28
140	232.47	1.00	1.00	232.47	-9150.16	-4500.81
141	221.13	1.00	1.00	221.13	-8929.03	-4279.68
142	73.97	0.44	0.00	0.00	-8855.06	-4279.68
143	505.78	0.44	0.00	0.00	-8349.28	-4279.68
144	191.95	1.00	1.00	191.95	-8157.33	-4087.73
145	139.14	1.00	1.00	139.14	-8018.19	-3948.59
146	519.10	0.96	0.96	497.82	-7499.09	-3450.77
147	-345.62	0.95	0.95	-328.57	-7844.71	-3779.34
148	609.62	1.00	1.00	609.62	-7235.09	-3169.72
149	61.85	1.00	1.00	61.64	-7173.24	-3108.09
150	184.21	0.50	0.00	0.00	-6989.03	-3108.09
151	-431.94	0.44	0.00	0.00	-7420.97	-3108.09
152	-67.10	0.44	0.00	0.00	-7488.07	-3108.09
153	-1088.77	1.00	1.00	-1088.77	-8576.84	-4196.86
154	-919.09	1.00	1.00	-919.09	-9495.93	-5115.95
155	492.83	0.44	0.00	0.00	-9003.10	-5115.95
156	544.46	1.00	1.00	544.46	-8458.64	-4571.49
157	315.73	1.00	1.00	315.73	-8142.91	-4255.76
158	-75.19	0.69	0.69	-52.14	-8218.10	-4307.90
159	-233.41	1.00	1.00	-233.41	-8451.51	-4541.31
160	847.11	1.00	1.00	844.98	-7604.40	-3696.33
161	33.03	0.44	0.00	0.00	-7571.37	-3696.33
162	393.19	1.00	1.00	392.23	-7178.18	-3304.10

Table C- 19 Profit record of XGB_EURAUDEURUSD_Total

	Total_Profit	y_pred	y_filter	Profit_w_ML	Cumulative_profit	Cumulative_profit _w_ML
0	709.08	0.55	1.00	709.08	709.08	709.08
1	886.77	0.87	0.87	772.05	1595.85	1481.13
2	254.11	0.27	0.00	0.00	1849.96	1481.13
3	248.06	0.67	0.67	166.14	2098.02	1647.28
4	-847.82	0.38	0.00	0.00	1250.20	1647.28
5	794.45	0.51	0.51	406.17	2044.65	2053.45
6	-53.28	0.66	0.66	-35.01	1991.37	2018.43
7	-1062.19	0.41	0.00	0.00	929.18	2018.43
8	331.28	0.61	0.61	201.76	1260.46	2220.20
9	537.25	0.31	0.00	0.00	1797.71	2220.20
10	1203.80	0.73	0.73	875.89	3001.51	3096.09
11	342.54	0.54	0.54	184.52	3344.05	3280.61
12	574.88	0.51	0.51	295.67	3918.93	3576.28
13	218.28	0.68	0.68	147.45	4137.21	3723.72
14	-204.50	0.70	0.70	-142.88	3932.71	3580.84
15	-582.99	0.69	0.69	-403.54	3349.72	3177.31
16	514.81	0.57	0.57	295.22	3864.53	3472.53
17	160.83	0.55	0.55	88.44	4025.36	3560.97
18	340.20	0.71	0.71	241.88	4365.56	3802.85
19	-943.22	0.72	0.72	-675.80	3422.34	3127.05
20	-319.58	0.47	0.00	0.00	3102.76	3127.05
21	476.91	0.65	0.65	309.38	3579.67	3436.43
22	-278.73	0.44	0.00	0.00	3300.94	3436.43
23	-410.63	0.72	0.72	-297.33	2890.31	3139.09
24	69.12	0.60	0.60	41.23	2959.43	3180.32
25	-429.91	0.74	0.74	-316.88	2529.52	2863.45
26	-1164.61	0.66	0.66	-766.28	1364.91	2097.17
27	-512.12	0.79	0.79	-402.49	852.79	1694.68
28	-225.21	0.77	0.77	-174.44	627.58	1520.24
29	196.74	0.46	0.00	0.00	824.32	1520.24
30	2120.21	0.28	0.00	0.00	2944.53	1520.24
31	-945.53	0.62	0.62	-584.40	1999.00	935.85
32	449.19	0.65	0.65	293.58	2448.19	1229.43
33	160.99	0.31	0.00	0.00	2609.18	1229.43
34	37.76	0.73	0.73	27.43	2646.94	1256.86
35	-1338.60	0.86	0.86	-1154.30	1308.34	102.56
36	341.59	0.69	0.69	236.37	1649.93	338.92
37	169.96	0.46	0.00	0.00	1819.89	338.92
38	161.73	0.64	0.64	102.99	1981.62	441.92
39	377.53	0.72	0.72	270.50	2359.15	712.41
40	806.13	0.73	0.73	589.40	3165.28	1301.81

41	45.06	0.80	0.80	36.01	3210.34	1337.83
42	111.77	0.70	0.70	78.34	3322.11	1416.17
43	50.45	0.84	0.84	42.15	3372.56	1458.32
44	890.96	0.73	0.73	650.91	4263.52	2109.23
45	-32.20	0.70	0.70	-22.42	4231.32	2086.81
46	481.89	0.76	0.76	366.12	4713.21	2452.93
47	-25.05	0.57	0.57	-14.22	4688.16	2438.71
48	738.99	0.66	0.66	488.59	5427.15	2927.30
49	362.48	0.43	0.00	0.00	5789.63	2927.30
50	307.38	0.30	0.00	0.00	6097.01	2927.30
51	-2079.74	0.44	0.00	0.00	4017.27	2927.30
52	364.07	0.55	0.55	199.86	4381.34	3127.16
53	216.38	0.70	0.70	150.49	4597.72	3277.65
54	-181.07	0.85	0.85	-153.23	4416.65	3124.42
55	-350.42	0.73	0.73	-256.50	4066.23	2867.92
56	430.94	0.81	0.81	350.00	4497.17	3217.91
57	-126.34	0.56	0.56	-70.42	4370.83	3147.49
58	2006.79	0.53	0.53	1058.54	6377.62	4206.03
59	-676.34	0.52	0.52	-351.90	5701.28	3854.13
60	674.20	0.39	0.00	0.00	6375.48	3854.13
61	541.45	0.61	0.61	329.68	6916.93	4183.81
62	7.71	0.24	0.00	0.00	6924.64	4183.81
63	-2129.16	0.40	0.00	0.00	4795.48	4183.81
64	-483.10	0.80	0.80	-388.65	4312.38	3795.16
65	55.33	0.66	0.66	36.24	4367.71	3831.40
66	11.27	0.71	0.71	8.04	4378.98	3839.44
67	393.05	0.33	0.00	0.00	4772.03	3839.44
68	748.44	0.30	0.00	0.00	5520.47	3839.44
69	1078.92	0.74	0.74	795.26	6599.39	4634.69
70	-335.96	0.68	0.68	-229.45	6263.43	4405.25
71	-208.15	0.82	0.82	-171.67	6055.28	4233.58
72	-133.21	0.32	0.00	0.00	5922.07	4233.58
73	347.26	0.75	0.75	259.67	6269.33	4493.26
74	118.27	0.43	0.00	0.00	6387.60	4493.26
75	-15.37	0.58	0.58	-8.94	6372.23	4484.32
76	181.09	0.74	0.74	133.28	6553.32	4617.60
77	283.56	0.81	0.81	231.03	6836.88	4848.63
78	-482.35	0.48	0.00	0.00	6354.53	4848.63
79	440.75	0.57	0.57	252.79	6795.28	5101.42
80	243.73	0.74	0.74	179.41	7039.01	5280.83
81	1061.76	0.31	0.00	0.00	8100.77	5280.83
82	-493.02	0.84	0.84	-414.64	7607.75	4866.19
83	624.47	0.78	0.78	487.68	8232.22	5353.88
84	313.87	0.76	0.76	239.73	8546.09	5593.60

85	653.73	0.87	0.87	571.69	9199.82	6165.30
86	-1641.90	0.47	0.00	0.00	7557.92	6165.30
87	693.55	0.87	0.87	604.76	8251.47	6770.05
88	-1688.66	0.57	0.57	-967.86	6562.81	5802.20
89	5.37	0.63	0.63	3.39	6568.18	5805.58
90	-225.60	0.68	0.68	-153.08	6342.58	5652.50
91	285.52	0.79	0.79	226.25	6628.10	5878.76
92	591.58	0.73	0.73	429.93	7219.68	6308.69
93	3.29	0.25	0.00	0.00	7222.97	6308.69
94	371.46	0.70	0.70	261.35	7594.43	6570.03
95	636.85	0.26	0.00	0.00	8231.28	6570.03
96	592.14	0.72	0.72	426.84	8823.42	6996.87
97	261.86	0.40	0.00	0.00	9085.28	6996.87
98	511.72	0.81	0.81	416.22	9597.00	7413.09
99	137.71	0.80	0.80	109.76	9734.71	7522.85
100	-1020.97	0.69	0.69	-702.88	8713.74	6819.97
101	-626.05	0.74	0.74	-465.79	8087.69	6354.18
102	-288.91	0.78	0.78	-226.56	7798.78	6127.62
103	358.08	0.90	0.90	323.24	8156.86	6450.86
104	391.41	0.68	0.68	266.38	8548.27	6717.24
105	307.23	0.76	0.76	233.44	8855.50	6950.68
106	-156.78	0.68	0.68	-106.22	8698.72	6844.46
107	-1440.55	0.55	0.55	-787.76	7258.17	6056.70
108	-438.87	0.69	0.69	-302.38	6819.30	5754.32
109	371.36	0.35	0.00	0.00	7190.66	5754.32
110	72.95	0.76	0.76	55.37	7263.61	5809.69
111	-569.87	0.58	0.58	-329.37	6693.74	5480.32
112	546.57	0.64	0.64	350.46	7240.31	5830.78

Table C- 20 Profit record of LR_EURGBPEURCAD_Total_Leg

	Profit	y_pred	y_filter	Profit_w_ML	Cumulative_profit	Cumulative_profit _w_ML
0	155.91	0.44	0.00	0.00	155.91	0.00
1	365.86	0.43	0.00	0.00	521.77	634.76
2	921.56	0.69	0.69	634.76	1443.33	634.76
3	512.30	0.39	0.00	0.00	1955.63	634.76
4	39.73	0.40	0.00	0.00	1995.36	634.76
5	839.97	0.43	0.00	0.00	2835.33	634.76
6	396.57	0.40	0.00	0.00	3231.90	634.76
7	349.69	0.40	0.00	0.00	3581.59	634.76
8	584.47	0.50	0.00	0.00	4166.06	634.76
9	-378.48	0.48	0.00	0.00	3787.58	634.76
10	511.26	0.42	0.00	0.00	4298.84	902.32
11	524.53	0.51	0.51	267.56	4823.37	1431.52
12	1032.26	0.51	0.51	529.20	5855.63	1800.68
13	688.31	0.54	0.54	369.16	6543.94	1284.65
14	-844.31	0.61	0.61	-516.02	5699.63	260.70
15	-1630.61	0.63	0.63	-1023.95	4069.02	260.70
16	11.34	0.42	0.00	0.00	4080.36	671.41
17	769.85	0.53	0.53	410.71	4850.21	671.41
18	-326.26	0.49	0.00	0.00	4523.95	623.95
19	-92.81	0.51	0.51	-47.46	4431.14	749.28
20	243.14	0.52	0.52	125.33	4674.28	749.28
21	167.26	0.40	0.00	0.00	4841.54	749.28
22	-2438.64	0.30	0.00	0.00	2402.90	547.85
23	-382.70	0.53	0.53	-201.43	2020.20	547.85
24	-443.31	0.36	0.00	0.00	1576.89	759.47
25	405.86	0.52	0.52	211.62	1982.75	-296.53
26	-1144.74	0.92	0.92	-1056.00	838.01	-372.60
27	-143.74	0.53	0.53	-76.07	694.27	-372.60
28	1713.89	0.17	0.00	0.00	2408.16	16.25
29	727.33	0.53	0.53	388.85	3135.49	268.50
30	493.71	0.51	0.51	252.25	3629.20	268.50
31	271.55	0.44	0.00	0.00	3900.75	261.00
32	-14.42	0.52	0.52	-7.50	3886.33	-375.89
33	-1254.96	0.51	0.51	-636.89	2631.37	-375.89
34	-68.51	0.36	0.00	0.00	2562.86	-375.89
35	131.40	0.45	0.00	0.00	2694.26	-375.89
36	-661.47	0.45	0.00	0.00	2032.79	-304.97
37	118.93	0.60	0.60	70.92	2151.72	180.00
38	911.04	0.53	0.53	484.97	3062.76	-242.29
39	-595.90	0.71	0.71	-422.29	2466.86	-223.11
40	25.50	0.75	0.75	19.19	2492.36	-458.74

41	-412.33	0.57	0.57	-235.63	2080.03	-458.74
42	266.23	0.46	0.00	0.00	2346.26	-225.98
43	430.73	0.54	0.54	232.76	2776.99	-225.98
44	254.87	0.43	0.00	0.00	3031.86	-225.98
45	865.03	0.42	0.00	0.00	3896.89	-225.98
46	833.81	0.30	0.00	0.00	4730.70	-95.99
47	214.16	0.61	0.61	129.99	4944.86	-795.21
48	-1240.31	0.56	0.56	-699.21	3704.55	-665.46
49	246.54	0.53	0.53	129.75	3951.09	-665.46
50	113.86	0.40	0.00	0.00	4064.95	-665.46
51	-1343.90	0.44	0.00	0.00	2721.05	-665.46
52	402.61	0.48	0.00	0.00	3123.66	-505.08
53	312.69	0.51	0.51	160.38	3436.35	-505.08
54	315.18	0.43	0.00	0.00	3751.53	-505.08
55	339.71	0.43	0.00	0.00	4091.24	-505.08
56	122.41	0.45	0.00	0.00	4213.65	-505.08
57	-400.35	0.47	0.00	0.00	3813.30	-533.51
58	-56.55	0.50	0.50	-28.43	3756.75	-533.51
59	281.71	0.45	0.00	0.00	4038.46	-533.51
60	-2896.79	0.43	0.00	0.00	1141.67	-533.51
61	-292.37	0.40	0.00	0.00	849.30	-533.51
62	466.49	0.49	0.00	0.00	1315.79	-533.51
63	548.37	0.50	0.00	0.00	1864.16	-367.14
64	318.43	0.52	0.52	166.37	2182.59	-390.49
65	-37.43	0.62	0.62	-23.35	2145.16	-1974.88
66	-2813.63	0.56	0.56	-1584.39	-668.47	-1974.88
67	408.42	0.45	0.00	0.00	-260.05	-1961.85
68	23.18	0.56	0.56	13.03	-236.87	-1961.85
69	499.41	0.43	0.00	0.00	262.54	-1705.52
70	391.36	0.65	0.65	256.33	653.90	-1705.52
71	-128.95	0.40	0.00	0.00	524.95	-1705.52
72	446.45	0.49	0.00	0.00	971.40	-1479.52
73	442.96	0.51	0.51	226.01	1414.36	-1479.52
74	202.51	0.36	0.00	0.00	1616.87	-1479.52
75	187.22	0.31	0.00	0.00	1804.09	-1479.52
76	-1949.97	0.44	0.00	0.00	-145.88	-1479.52
77	414.41	0.47	0.00	0.00	268.53	-1479.52
78	52.01	0.44	0.00	0.00	320.54	-1479.52
79	-315.04	0.45	0.00	0.00	5.50	-1692.53
80	-375.70	0.57	0.57	-213.01	-370.20	-1692.53
81	-533.08	0.48	0.00	0.00	-903.28	-1984.74
82	-581.79	0.50	0.50	-292.20	-1485.07	-1984.74
83	-85.70	0.43	0.00	0.00	-1570.77	-2456.72
84	-758.45	0.62	0.62	-471.99	-2329.22	-2320.47

85	268.47	0.51	0.51	136.25	-2060.75	-2320.47
86	66.83	0.47	0.00	0.00	-1993.92	-2320.47
87	-1152.24	0.28	0.00	0.00	-3146.16	-2477.70
88	-266.08	0.59	0.59	-157.23	-3412.24	-1818.63
89	1280.57	0.51	0.51	659.07	-2131.67	-1818.63
90	441.73	0.42	0.00	0.00	-1689.94	-1818.63
91	347.51	0.44	0.00	0.00	-1342.43	-1691.13
92	239.43	0.53	0.53	127.50	-1103.00	-1691.13
93	-53.77	0.49	0.00	0.00	-1156.77	-1691.13
94	-82.76	0.35	0.00	0.00	-1239.53	-1691.13
95	-37.98	0.37	0.00	0.00	-1277.51	-1615.12
96	136.22	0.56	0.56	76.01	-1141.29	-1615.12
97	922.42	0.38	0.00	0.00	-218.87	-1615.12
98	390.73	0.43	0.00	0.00	171.86	-1615.12
99	-10.48	0.36	0.00	0.00	161.38	-1615.12
100	306.04	0.40	0.00	0.00	467.42	-1615.12
101	-958.10	0.48	0.00	0.00	-490.68	-1615.12
102	-695.18	0.46	0.00	0.00	-1185.86	-1327.87
103	502.83	0.57	0.57	287.24	-683.03	-1327.87
104	186.17	0.50	0.00	0.00	-496.86	-1327.87
105	158.03	0.43	0.00	0.00	-338.83	-1327.87
106	243.73	0.45	0.00	0.00	-95.10	-1327.87
107	-1837.83	0.43	0.00	0.00	-1932.93	-1327.87
108	-455.53	0.48	0.00	0.00	-2388.46	-1319.59
109	14.98	0.55	0.55	8.28	-2373.48	-794.44
110	786.56	0.67	0.67	525.15	-1586.92	-794.44
111	236.97	0.47	0.00	0.00	-1349.95	-794.44
112	-692.43	0.47	0.00	0.00	-2042.38	-794.44
113	466.53	0.49	0.00	0.00	-1575.85	-738.55
114	103.39	0.54	0.54	55.89	-1472.46	-738.55
115	-347.71	0.47	0.00	0.00	-1820.17	-738.55
116	148.26	0.45	0.00	0.00	-1671.91	-639.50
117	163.17	0.61	0.61	99.05	-1508.74	-639.50
118	196.45	0.42	0.00	0.00	-1312.29	-421.14
119	410.56	0.53	0.53	218.36	-901.73	-421.14
120	-433.30	0.40	0.00	0.00	-1335.03	-301.78
121	221.75	0.54	0.54	119.36	-1113.28	-301.78
122	205.69	0.41	0.00	0.00	-907.59	36.12
123	618.45	0.55	0.55	337.90	-289.14	36.12
124	140.44	0.47	0.00	0.00	-148.70	-2.80
125	-77.78	0.50	0.50	-38.92	-226.48	-2.80
126	-93.68	0.42	0.00	0.00	-320.16	-2.80
127	468.28	0.44	0.00	0.00	148.12	232.71
128	368.68	0.64	0.64	235.51	516.80	232.71

129	-1132.15	0.43	0.00	0.00	-615.35	232.71
130	-1195.39	0.42	0.00	0.00	-1810.74	232.71
131	-460.37	0.42	0.00	0.00	-2271.11	502.87
132	493.67	0.55	0.55	270.16	-1777.44	546.45
133	85.25	0.51	0.51	43.58	-1692.19	546.45
134	270.61	0.47	0.00	0.00	-1421.58	546.45
135	-255.96	0.48	0.00	0.00	-1677.54	546.45
136	-1415.16	0.29	0.00	0.00	-3092.70	434.09
137	-207.09	0.54	0.54	-112.35	-3299.79	434.09
138	296.65	0.38	0.00	0.00	-3003.14	330.73
139	-166.31	0.62	0.62	-103.36	-3169.45	330.73
140	286.95	0.49	0.00	0.00	-2882.50	471.79
141	275.99	0.51	0.51	141.06	-2606.51	471.79
142	305.21	0.40	0.00	0.00	-2301.30	471.79
143	488.16	0.48	0.00	0.00	-1813.14	471.79
144	-840.35	0.48	0.00	0.00	-2653.49	319.76
145	-290.08	0.52	0.52	-152.03	-2943.57	319.76
146	602.95	0.48	0.00	0.00	-2340.62	319.76
147	-609.60	0.46	0.00	0.00	-2950.22	319.76
148	-741.57	0.38	0.00	0.00	-3691.79	319.76

REFERENCES

[1] "US Airways Group, Inc. Private Company with Public Debt Profile." https://www.capitaliq.com/CIQDotNet/company.aspx?companyId=169142 (accessed 3 November 2019.

[2] "United Airlines, Inc. Private Company with Public Debt Profile." https://www.capitaliq.com/CIQDotNet/company.aspx?companyId=26977 (accessed 3 November, 2019).

[3] W. Goetzmann, K. Rouwenhorst, and E. Gatev, "Pairs Trading: Performance of a Relative Value Arbitrage Rule," *Review of Financial Studies,* vol. 19, pp. 797-827, 02/17 2006, doi: 10.1093/rfs/hhj020.

[4] C. C. F. Chu and P. K. Chan, "Mining Profitable High Frequency Pairs Trading Forex Signal Using Copula and Deep Neural Network," in *2018 19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, 27-29 June 2018 2018, pp. 312-316, doi: 10.1109/SNPD.2018.8441125.

[5] J. Rudy, C. Dunis, and J. Laws, "Mean Reversion Based on Autocorrelation: A Comparison Using the S&P 100 Constituent Stocks and the 100 Most Liquid ETFs," *SSRN Electronic Journal*, 01/31 2011, doi: 10.2139/ssrn.2272794.

[6] T. Datta Chaudhuri, I. Ghosh, and P. Singh, "Applications of Machine Learning Tools in Predictive Modeling of Pairs Trade in Indian Stock Market," *The IUP Journal of Applied Finance,* vol. 23, pp. 5-25, 01/01 2017.

[7] C.-F. Huang, C.-J. Hsu, C.-C. Chen, B. Chang, and C.-A. Li, "An Intelligent Model for Pairs Trading Using Genetic Algorithms," *Computational intelligence and neuroscience,* vol. 2015, p. 939606, 09/04 2015, doi: 10.1155/2015/939606.

[8] C. Dunis, G. Giorgioni, J. Laws, and J. Rudy, "Statistical Arbitrage and High-Frequency Data with an Application to Eurostoxx 50 Equities," *SSRN Electronic Journal*, 04/01 2010, doi: 10.2139/ssrn.2272605.

[9] Y. Chen, W. Chen, and S. Huang, "Developing Arbitrage Strategy in High-frequency Pairs Trading with Filterbank CNN Algorithm," in *2018 IEEE International Conference on Agents (ICA)*, 28-31 July 2018 2018, pp. 113-116.

[10] P. J. Kaufman, *Alpha Trading: Profitable Strategies That Remove Directional Risk*. John Wiley & Sons, 2011.

[11] *EURUSD Historical exchange data*, MetaQuotes Software Corp.

[12] *EURUSD and AUDUSD with RSI indicator Historical exchange data*, MetaQuotes Software Corp.

[13] S. Riedinger, "Demystifying Pairs Trading: The Role of Volatility and Correlation," *SSRN Electronic Journal*, 01/01 2017, doi: 10.2139/ssrn.2774063.

[14] J. F. Caldeira and G. V. Moura, "Selection of a Portfolio of Pairs Based on Cointegration: A Statistical Arbitrage Strategy," 2013.

[15] E. Sinclair, *Volatility Trading*. Hoboken, New Jersey: John Wiley and Sons, 2013.

[16] J. Heaton, *Introduction to Neural Networks for Java*, 2nd ed. St. Louis: Heaton Research, Inc., 2008.

[17] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *International Conference on Learning Representations*, 12/22 2014.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov,

"Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[19] D. W. T. H. R. T. Gareth James, *An introduction to statistical learning : with applications in R*. New York : Springer, [2013] ©2013, 2013.

VITA

NAME รณชัย จิระพงษ์พันธ์

DATE OF BIRTH 6 ตุลาคม 2535

PLACE OF BIRTH กรุงเทพ

INSTITUTIONS ATTENDED ปริญญาโท

HOME ADDRESS 11/32 หมู่บ้านธีรินทร์พาร์ค ถนนบางแวก แขวงบางไผ่ เขตบางแค กรุงเทพ 10160