STUDY OF THE TIME-SERIES VALIDATION SCHEMES ON THE PERFORMANCE OF TREE-BASED MODELS

Yoong Pei Xian

A dissertation submitted to
The School of Computing Sciences of the University of East Anglia
in partial fulfilment of the requirements for the degree of
MASTER OF SCIENCE.
AUGUST, 2022

SUPERVISOR(S), MARKERS/CHECKER AND ORGANISER

The undersigned hereby certify that the markers have independently marked the dissertation entitled "**Study of the Time-Series Validation Schemes on the Performance of Tree-Based Models**" by **Yoong Pei Xian**, and the external examiner has checked the marking, in accordance with the marking criteria and the requirements for the degree of **Master of Science**.

Supervisor: _____
Dr. Antony Jackson

Markers: _____
Marker 1: Dr. Antony Jackson

_____
Marker 2: Dr. Shaun Parsley

External Examiner: _____
Checker/Moderator

Moderator: _____
Dr. Wenjia Wang

DISSERTATION INFORMATION AND STATEMENT

Dissertation Submission Date: **August, 2022**

Student:	**Yoong Pei Xian**
Title:	**Study of the Time-Series Validation Schemes on the Performance of Tree-Based Models**
School:	**Computing Sciences**
Course:	**Computing Science**
Degree:	**MSc.**
Duration:	**2021–2022**
Organiser:	**Dr. Wenjia Wang**

STATEMENT:

Unless otherwise noted or referenced in the text, the work described in this dissertation is, to the best of my knowledge and belief, my own work. It has not been submitted, either in whole or in part for any degree at this or any other academic or professional institution.

Subject to confidentiality restriction if stated, permission is herewith granted to the University of East Anglia to circulate and to have copied for non-commercial purposes, at its discretion, the above title upon the request of individuals or institutions.

Yoong Pei Xian

Signature of Student

Abstract

With respect to time-series stock price prediction, cross-validation (CV) is an essential statistical method to evaluate the performance of machine learning models. Traditional validation schemes such as k-folds CV usually pose a limitation of the temporal dependencies issue as data selected in each fold is completely random. Reviewing the existing studies on stock price prediction reveals that there is only a limited number of papers examining the time-series CV schemes such as the use of Out-Of-Samples CV. Therefore, it provides an incentive to investigate the performance comparison of the traditional and time-series CV schemes, and the deterministic factors of their performances.

This study aims to investigate the performance of four validation schemes: i) Single Train-Validation, ii) K-Fold CV, iii) Rolling Window CV, and iv) Expanding Window CV, on tree-based models for the time-series S&P 500 stock price prediction. Understanding that input feature may be a deterministic factor in affecting the performance, two groups of input features i) technical, and ii) the combination of technical and sentiment indicators - VIX are proposed. Empirical experiments are conducted on major US stock indices, S&P 500 from 1997-07-01 to 2002-06-30.

Evidence suggests that the use of Expanding Window CV for time-series prediction to be more superior as it achieves a satisfying Sharpe ratio in most models while being less computationally expensive. Its limitation in handling the class imbalance issue may be addressed using class balancing techniques. Furthermore, we believe that its potential can be further elaborated when trained with more robust models, such as RFC to generate a profitable portfolio as EW_CV can effectively shorten the computational time of the more complex models. Our results suggest that ADA and BC are more capable of handling the class imbalance issue, which commonly exists in the stock price data, while RFC yields outstanding performance in generating portfolio profitability. Besides, we also discover the importance of input feature in affecting the performance of validation schemes and tree-based models, despite VIX does not show predictive power on the S&P 500 stock price prediction.

Acknowledgements

I would like to thank my supervisor, Dr. Antony Jackson, for his many suggestions and constant support during this research.
I would also like to express my gratitude to my family and friends for all the unconditional support in this very intense academic year.

Yoong Pei Xian

Norwich, UK.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

Stock price prediction is challenging due to its dynamic, non-linear, and non-parametric nature. Besides, the uncertainty of the future implies that no matter how good our analysis is, it is only as good as the information that is available right now. This argument has been also supported by the financial theories, the efficient market hypothesis (Fama 1965) which states that stock price reflects on all the publicly available information due to market participants being rational profit-maximizers, and it is impossible to outperform the overall market using any method since price should only react to new information (Downey 2022). If any anomaly existed, it would quickly be exploited and removed, leading to a more efficient state.

This hypothesis explains the random walk theory, meaning that most of the fluctuations in prices are explained by the changes in the instantaneous demand and supply of any given stock, causing the random walk-in prices (Qian & Rasheed 2006). There have been controversial results yielded on the random walk model testing. In Jensen (1978)'s study, the researchers even mentioned that no proposition can empirically support the price movement better than the efficient market hypothesis, contrary to the more recent studies providing evidence that the price movement does not only follow the random walk patterns but there are other factors affecting it (Gallagher & Taylor 2002, Doan & Lo 1988).

On the contrary, the Wisdom of Crowd hypothesis, first popularized by New Yorker writer James Surowiecki in his 2004 book, The Wisdom of Crowds states that a large crowd with a diversity of opinions, each with limited information, can provide very

accurate speculation on the market movement should the opinions are independent and free from the influence of others (Clay 2022). This might help explain how it becomes mob chaotic to affect the stock price of GameStop and Bitcoin significantly (Mims 2021). Nevertheless, there is no empirical study to prove its efficiency due to various uncertainties involved and the presence of other factors that could potentially affect the stock price movement. Over time, various studies have emerged to explain the different factors, such as political events, economic circumstances, and investors' sentiments to be the deterministic factors that can move the market. Indeed, in the late nineteenth century, Malkiel (2003) commented that the Efficient Market Hypothesis has become less influential as more researchers and financial institutions discovered that stock prices are at least partially predictable.

Driven by the motivation to reap profits, researchers have been actively investigating the three conventional approaches to predicting market movements, including traditional time-series forecasting, technical analysis, and machine learning method. Attributed to the tremendous ability of machine learning to predict future data based on the available information and patterns, plenty of research apply the application of machine learning in stock price prediction to develop models that can produce more accurate prediction to provide incentives to the professionals that involve in the algorithm trading. To minimize the associated risk of stock market investment, foreknowledge of the future price movement is desired. Investors are more likely to buy a stock whose price is expected to rise in the future and desist from those with expected falling prices. Existing studies on stock price prediction demonstrate that machine learning can produce better predictability compared to the traditional statistical and econometric models when optimized correctly (Hsu et al. 2016, Weng et al. 2017, Zhang & Wu 2009), rooting out potentially bad investments to avoid huge financial loss.

With the emergence of Artificial Intelligence, various algorithms have been employed in order to predict the stock market movement, including the basic statistical models, such as logistic regression (LR) (Brownstone 1996), simple decision tree (DT) (Wu et al. 2006), tree-based ensemble models (Ampomah et al. 2020) support vector machine (SVM) (Ren et al. 2019, Lee 2009), and more complex deep learning models (Donaldson & Kamstra 1996, Refenes et al. 1994). Most of the time cross-validation

(CV), a popular technique for tuning hyperparameters and producing robust measurements of model performance is often performed to evaluate these models. It accesses the generalization ability of the model, in the other words, the accuracy of a predictive model on the unseen data. One advantage of using CV is the full use of all available data in the dataset to achieve adequacy and diversity.

In the case of stock price prediction, where time-series data set is involved, CV becomes a critical issue as traditional CV presents limitations that may violate the fundamental assumptions of that all data should be identically distributed and independent (Arlot & Celisse 2009). Taking the most popular k-fold CV for instance, the data selected in each fold or stratified is completely random, causing the possibility of using the future data to predict the past, resulting in the issue of temporal dependencies that might cause unreasonable correlation between training and test set (Jiang & Wang 2017).

A closer look at the existing literature reveals that these limitations are not addressed properly in a large proportion of the research, and the limited number of studies have been demonstrating controversial findings when drawing comparison on the performance of various validation schemes. Several researchers argue that the traditional validation schemes are robust enough to produce outstanding model performance without the need to define forecasting origin. On the other hand, several researchers propose the use of time-series validation schemes, such as CV on the rolling basis where evaluation are formed on a rolling forecasting origin to ensure that no future observations can be used in constructing the prediction, therefore solving the theoretical limitations of CV. Thus, it is encouraging to gain further insight into the different validation schemes to compare how the model performs under traditional and time-series validation schemes.

Another crucial issue when using algorithms for stock price prediction is related with the feature selection. Due to the ease of implementation, the majority of the early studies focus on using technical indicators, predominantly historical price and volume as the input features to predict the stock price movement (Wu et al. 2020, Agrawal et al. 2019). Several of the technical indicators have been confirmed to be deterministic in affecting the predictability of the stock price such as the immediate

fore-running price fluctuations Chen et al. (2007).

Nonetheless, researchers have been raising concern on the limitation of using only technical indicators as the input features as the external changes in the macroeconomic conditions, that reflects the shift of investor sentiment is one of the huge drivers of the stock price movement. For instance, the uncertainty over pervasive inflation, the potential for a recession, hints from the Federal Reserve about more aggressive interest rate hikes, and gloomier sentiments in waves of layoffs among recently booming technology companies (Ponciano 2022) have led to the recent downfall in the US stock market, proving the vulnerability of the stock price to the expectations about future price movement and the shifts in public sentiments (Giaglis et al. 2015). This is further validated by André Kostolany, one of the most successful investors of the 20th century, quoting that "facts only account for 10% of the reactions on the stock market; everything else is psychology." (Palmer 2021). Sentiment indicators are usually generated by analysing social media content through Natural Language Processing (NLP) or using a simpler alternative of examining the Cboe Volatility Index (VIX), one of the most well-known measure of market sentiment by looking at the expected price fluctuations or volatility in the S&P 500.

With our manuscript, we attempt to close the gap in the existing studies by investigating the different validation schemes i) Single Train-Validation (STV), ii) K-Fold CV (KF_CV), iii) Rolling Window CV (RW_CV), and iv) Expanding Window CV (EW_RW) to understand their impacts on the time-series prediction. A selection of representative machine learning tree-based models are considered: i) Decision Tree Classifier (DTC), ii) Random Forest Classifier (RFC), iii) Extra Trees Classifier (ETC), iv) AdaBoost Classifier (ADA), v) Bagging Classifier (BC). Understanding the importance of feature selection and with respect to the model selection procedure, two groups of input features are proposed: i) only technical indicators (Group T), ii) the combination of technical and sentiment indicators - VIX (Group TS).

Empirical experiments are conducted on major US stock indices, S&P 500 from 1997-07-01 to 2002-06-30. The study focuses on the S&P 500 to adequately reflect the entire range of US stock of different industries. The accuracy and f1-score are computed to evaluate the model efficiency. Back-testing is performed to examine the

annual Sharpe ratio of the investments that can potentially be generated in real-life trading.

1.2 Aim and Objectives

The concise aims and objectives of this research are listed below:

- To investigate the performance of validation schemes on the time-series S&P 500 stock price prediction.
- To examine the performance of the tree-based models in S&P 500 stock price prediction.
- To study if input feature affects the performance of the proposed validation schemes and models.
- To explore the predictive power of VIX on S&P 500 stock price prediction

1.3 Structure of Dissertation

The rest of the paper is structured as follows. Section 2 briefly reviews the existing literature. Section 3 explains the research questions. Section 4, therefore, discusses the methodologies in details, regarding data partitioning in training and test set, and metrics proposed for model evaluation. Section 5 presents the experimental results and discusses key findings in our study. The conclusions and future directions for research are summarized in Section 6.

Chapter 2

Literature Review

2.1 Time-series Cross-Validation Schemes

Stock price prediction involves developing time-series models to predict future values based on past observed values. Time-series prediction is widely used for non-stationary data, whose statistical properties, for instance, the mean and standard deviation vary over time. Researchers have been looking for a more optimal methods to perform CV for time-series prediction as time-series data present several challenges. The use of CV appears to be problematic as the data selected in each fold or stratified is completely random, causing the possibilities of using the future data to predict the past.

In order to accurately simulate the real-world forecasting environment, Tashman (2000) recognized that the researchers must withhold all data about events that occur chronologically after the events used for fitting the model. Besides, Bergmeir & Benítez (2012) pointed out the issues of series correlation, where two columns of data are correlated to each other, except that the data are ordered in sequence through time, agreed by Jiang & Wang (2017) claiming that the existing CV lacks of ability to deal with series correlation, overlapping and periodicity as standard CV such as k-fold and ShuffleSplit assume the samples are independent and identically distributed, and would result in unreasonable correlation between training and test set.

There have been debates on whether the use of standard-fold CV is adequate for the time-series prediction. Bergmeir et al. (2015) provided evidence that standard-fold CV without any modification performed better than other time-series validation schemes, provided that the residuals of the model are uncorrelated, supported by Opsomer et al. (2000) claiming that cross-validation is bound to fail if correlation between errors of

time series exists. Nonetheless, they pointed out that this method should be avoided if the models underfit the data as it could lead to a systematic error underestimation. To solve the issue of series correlation, Ronchetti (2000) proposed to follow the non-dependent CV by removing all the data correlated with the test data from the training set but the study is criticized by Bergmeir & Benítez (2012) stating that the omission of dependent data may lead to significant data loss and may not be feasible depending on the embedding dimension of the dataset. To circumvent the theoretical problem of serial correlation, Bergmeir & Benítez (2012) proposed the use of blocked cross-validation that divides data into partitions sequentially.

Several CV methods for time series have been developed in the recent years to address the main critical issue of series correlation. One of the most popular methods is Out-Of-Sample evaluation, which is commonly applied in the time-series prediction. Out-Of-Sample adopts either expanding or rolling CV to determine the start date of the training set. The former utilizes the same start date while expanding the period length of the training set. The later successively update the start date while either adding new observations to the training set – rolling-origin evaluation or pruning the oldest observations to make the period length of the training set constant – rolling-window evaluation (Tashman 2000). Out-Of-Sample allows the serial dependence to be respected due to its iteration method of validating the future data, however, data leakage may be introduced to the model. Bergmeir & Benítez (2012) argued that rolling CV is only applicable if the model is rebuilt in every window, and the result would show distinct improvement only when the old data is disturbing the model generation. In Jiao & Jakubowicz (2017)'s studies to compare the performance of different CV, they discovered that rolling window method demonstrated the highest validation-test correlation, indicating it as a better choice in a time series context and it should be applied to any of the time-series prediction. Jiao & Jakubowicz (2017) mentioned in their research of the series splitting issues in implementing the Out-Of-Sample that the arbitrary choice of start date might affect the accuracy of the result, but it could be addressed by distinguishing the time interval that reflects the different critical phase of the research context, for instance, monthly or quarterly cycle in the research focusing on business activity. To ensure the consistency of the result, Bergmeir & Benítez (2012)

added on that the series horizons need to be kept constant as data of different horizons usually presents different statistical properties and taking the average from prediction of different horizons could be misleading. Mittal (2011) proposed a new validation scheme, k-fold sequential cross validation, where the researchers trained on all days up to a specific day and test for the next k days. The same validation scheme has been adopted by Jiao & Jakubowicz (2017), and it performed comparatively worse than k-fold CV and rolling window method in the validation-test correlation.

2.2 Tree-Based Models in Stock Price Prediction

Decision tree is commonly used in many of the studies of stock price prediction due to the ease of implementation. Bai et al. (2019) finds that the fast classification and readability of the decision tree makes it an excellent algorithm in predicting stock price movement Despite of the ease of implementation, there have been critics on the overfitting issue of the decision tree, thus resulting in inaccurate interpretation of the accuracy score (Kamble 2017). Many studies using decision tree to predict stock price movement reveals that it produces a relatively worse results compared to the simpler algorithm such as linear regression (Karim et al. 2021, Attigeri et al. 2015). Besides, decision tree also suffers from the high variance resulted from the great depth of trees , which does not usually occur in the ensemble models like random forest and extra trees. Therefore, ensemble models with decision tree as the base classifier are often preferred by many researchers to achieve satisfactory results.

Random Forest has been empirically proven to be the top performers among the tree-based models (Kumar & Thenmozhi 2006, Patel et al. 2015, Basak et al. 2019, Lohrmann & Luukka 2018, Weng et al. 2018). The study of Sadorsky (2021), which appears to be the first one to adopt Random Forest in the clean energy stock price prediction also finds the model to have strong predictive power. Basak et al. (2019) deploy two types of ensemble classifiers, Random Forest Classifier and Gradient Boosted Decision Trees (XGBoost) that consider only technical indicator based on historical price prevails several days earlier to predict the stock price movement of multiple US stocks. Researchers considered Random Forest Classifier to be more superior than

Decision Tree in the stock price prediction as it involves an intense voting-based conclusion which large amount of de-correlated trees are grown randomly to classify the instances. They also found out that Random Forest Classifier are less prone to overfitting, proven by the decreasing OOB error rate when more trees are added to the forest. The finding is in line with Kamble (2017), To validate the non-existence of heavy bias, researchers examine the F-score in comparison to the accuracy score.

Some recent examples also study into the Extra Trees, which resembles Random Forest by randomly sampling the features at each split point of a decision tree. Ampomah et al. (2020) discover that Extra Trees produced the highest accuracy on average when predicting the stock price movement of three US stock exchanges compared to other tree-based models. The finding is in line with the study conducted by (Sadorsky 2022), which the validity of the result is further strengthened by back testing the model using signals generated by Extra Trees. Nonetheless, existing studies reveals that Extra Trees is underutilized in the financial area compared to other more popular tree-based models such as Random Forest. Therefore, it is worth investigating into the effectiveness of Extra Trees in stock movement prediction.

Boosting and bagging are often introduced to improve the performance of single decision tree (Wang et al. 2009). There have been intense debates on the performance comparison between boosting and bagging. In terms of the predictive power, Freund & Schapire (2001) empirically reveals that boosting demonstrates superiority over bagging by reducing the prediction error to a greater extent, supported by many researchers (Drucker 1997, Akour et al. 2017). In the meantime, a considerable quantity of papers disagrees with the finding of Wang et al. (2009). Among the boosting algorithms, AdaBoost, short for Adaptive Boosting and is a very popular boosting technique which combines multiple weak classifiers, especially decision trees with a single split, called decision stumps, into a single strong classifier. Formulated by Yoav Freund and Robert Schapire in 1995, AdaBoost has been adopted by many researchers in the study of stock price prediction and claimed to enhance the model performance significantly. For instance, (Bauer & Kohavi 1996) study into the empirical comparison of voting classification and discover that AdaBoost reduces the misclassification error by 27% when compared with the individual decision tree, while improving the bias error.

(Ampomah et al. 2020) also confirms that ADA achieves the highest accuracy among the tree-based models in their studies. Apart from combining ADA with decision tree. Researchers also look into other base classifier, such as the Random Forest Classifier and Extra Trees Classifier proposed by Ampomah et al. (2021), where the study finds out that ADA when used with Bagging Classifier significantly improves accuracy while achieving a satisfying f1-score. The limitation of this study is that it does not properly account for the class distribution, thus it is unknown whether class imbalance exists to affect the model evaluation. In the meantime, Galar et al. (2012) confirm that bagging is powerful when dealing with class imbalance if they are properly combined with the appropriate data preprocessing techniques in their reviews on ensembles for the class imbalance problem.

Mentioning of the class distribution, class imbalance is a common issue in the stock price prediction, where data is skewed towards a certain class. The imbalance in the class distribution may vary, but a severe imbalance is more challenging to model and may require specialized techniques. When fronting class imbalance, most algorithms tend to produce suboptimal results. Common approaches to tackle class imbalance includes modifying the models or resampling, however, boosting, bagging and their hybrids are also frequently examined to address the issue. While many researchers confirm that boosting outperforms bagging, Khoshgoftaar et al. (2011) shows that bagging performs better when it comes to handling class imbalance in the noisy data environment, supported by Rekha et al. (2019), Galar et al. (2012).

2.3 Input Features - Technical and Sentiment Indicators

Existing studies show that selecting the correct input features, along with their data preprocessing and model selection procedure may enhance the model performance (Di Persio & Honchar 2016, Bi et al. 2020, Song & Lee 2020). Understanding that tree-based models are skillful at finding the most important input features, for instance, DTC considers all features and creates a split on the one that is separating class labels the best in terms of entropy; RFC is capable of sampling the features and

uses only a subset of it, nonetheless, it is still of our interest to explore the impact of input features on the performance of validation schemes and models by comparing the models trained using different groups of input features. Thus, indicators involved in the model development of previous studies are reviewed.

In an attempt to predict stock price movement, researchers usually rely on using technical indicators, most frequently the historical price data to develop machine learning models. Controversial findings have been published by researchers on whether technical indicators alone possess sufficiently strong predictive power to accurately capture the future price movement. Neely et al. (2014) reports that technical and sentiment indicators detect the different types of information, and both are relevant in predicting the stock price movement. Technical indicators exhibit statistically and economically significant predictive power when it comes to figuring out the rise and decline of stock price in different phrases of business cycles. The study of Shynkevich et al. (2017), on the other hand, concludes that the predictive power of technical indicators is strongly dependent on the window length and horizon of the features where the most optimal performance is usually produced when a window length that is set to be equal to the prediction horizon. It is worth highlighting that the limitations of these studies is that none of them factors the potentially linked events, for example, political and legislation changes, current phase in the stock price cycle (Accumulation, markup, distribution and markdown) (Special 2021), that happen on the same period into account when conducting the research, prompting doubts on the extent of influence of these external events.

Due to the highly volatile and complex nature of the stock market, using historical price data solely often produce incorrect predictions. Several studies have revealed that by coupling historical price data with the sentiment indicators that can well represent and reflect the external factors, such as the emotions shift as a result of the changes in government and influences policies, improve the model performance significantly (Zhai et al. 2007). In the study of sentiment analysis, most existing papers adopt text classification, where investors' opinions are gathered from the online sources and preprocessed using natural language processing (NLP) algorithms to examine the context. Despite being poised to be one of the most disruptive areas of machine learning, NLP

can be time-consuming to train. In the meantime, VIX is an alternative to gauge the investors' sentiment but there has been relatively little literature published on studying the predictive power of VIX on the stock price movement. VIX explains the expected future stock market volatility. Market fear is higher when the VIX goes up. Smales (2014) confirmed the presence of a negative contemporaneous relationship between changes in VIX and investor sentiment.

It has been conclusively shown that there is a relationship between changes in VIX and news sentiment, particularly in times of market turmoil such as the financial crisis of 2007–2009 and the aftermath of the dot-com bubble in 2001. The changes in VIX are also negatively correlated to the contemporaneous returns on the §&P 500 index. The regression models developed by citeSmales:2017 illustrates the inclusion of VIX to improve the Adj.R2 by 150% on average, and the model fit (AIC), especially on the small-cap stocks, which is in consistency with the findings of Smales (2013), Wurgler & Baker (2006). Other studies have supported VIX to have superior explanatory power for future stock price prediction and enhance the accuracy in the model fitting process (Smales 2017, Professor & Wiggins 2001, Giot 2003). The main limitation of most of these studies is that they focus on the entire index, but not on specific industry. Smales (2017) highlighted that companies across different industries may response differently to the changes of VIX. For instance, the stock price of companies that are more subjective to value such as those heavily involved in the technology sector are more likely to fluctuate more in response to the changes VIX. Study conducted by Chen et al. (2021) narrowed down the investigation of the impact of VIX to only the stocks in the energy sectors, including natural gas spot, natural gas futures, WTI oil futures and spot, and Brent oil spot. Their findings demonstrated that despite being in the same sector, the predictive accuracy of VIX can still vary, where it only drastically enhances the predictive accuracy of crude oil-related stocks, but not natural gas futures and spot. This prompts questions to the research that focuses on the entire index such as S&P 500 and NASDAQ 100 on how feasible the models can be used by the individual stocks in different sectors.

Multiple papers relevant to adopting machine learning in stock price prediction have been reviewed in this section. Findings across these existing papers vary due to the

different experiment setting. The sample size, features, model building, hyperparameters tuning, selection, or even prediction interval could affect the overall performance, implying that it is impossible to draw a concluding comparison across these papers. However, some of the interesting topics that worth further exploration is discussed. These include the performance comparison of tree-based models, the predictive power of technical and sentiment indicators, and the existing cross-validation schemes.

A closer look to these papers on reveals several gaps and shortcomings. First, there are limited number of papers focusing on comparing the performance of different tree-based models in stock price prediction, so far lacking in the scientific literature. Most often tree-based models are compared with more complex algorithms such as Support Vector Machine (Nti et al. 2020, Kumar & Thenmozhi 2006, Kumar et al. 2016) and Deep Neural Network Krauss et al. (2016), Liu et al. (2020), Roy et al. (2020), Ampomah et al. (2021), Basak et al. (2019) are the very few researchers that evaluate the multiple tree-based models including the least used Extra Trees. Second, previous studies on sentiment analysis almost exclusively focus on using Natural Language Processing to analyse the social media context that comes with the cost of tremendous complexity. Khan et al. (2016) criticizes that many researchers neglect the importance of resolving the complex NLP challenges, thus hindering its performance. This provides an incentive to examine whether VIX, as a simpler alternative to understanding the social media context can effectively reflect the investors' sentiment on the stock price movement. Lastly, a still unsolved question is the performance comparison of various validation schemes in the time-series prediction, as most researchers only select one specific validation scheme in their studies. It remains to be a novel area to study, therefore, a more systematic and theoretical analysis is required to examine the robustness of different validation schemes

Chapter 3

Research Questions

Given that the existing research in stock price prediction poses limitations on the time-series validation schemes, it is of our interest to draw a comparison between the traditional and novel time-series cross-validation schemes on stock price prediction. This leads to the first research question.

— **How effective is the S&P 500 stock price prediction using the proposed validation schemes?**

Since algorithms play a major role in affecting the model performance due to their differences in characteristics by nature, we want to explore the predictive power of the proposed tree-based models and how they perform under each validation scheme, leading to the second research question.

— **How effective is the S&P 500 stock price prediction using the proposed tree-based models?**

We expect input features to play a deterministic role in the model performance, thus we are interested in examining if the input features affect the performance of the proposed validation scheme and algorithms by comparing the model performance with and without the use of VIX as an input feature. This leads to the third and fourth research questions.

—**Does feature affects the performance of the proposed validation schemes and models?**

—**Does VIX consists of the predictive power in S&P 500 stock price prediction?**

Chapter 4

Methodology Design

This section describes the data sources and their characteristics, as well as the model selection process to examine the performance of different validation schemes.

4.1 Data Pipeline

Experiments are set up to examine the predictability of the S&P 500 Price Movement using tree-based models. Python is selected as the main programming language and machine learning packages including pandas, matplotlib, and scikit-learn to develop models are involved. The data pipeline of the experiment is shown in Figure 4.1.

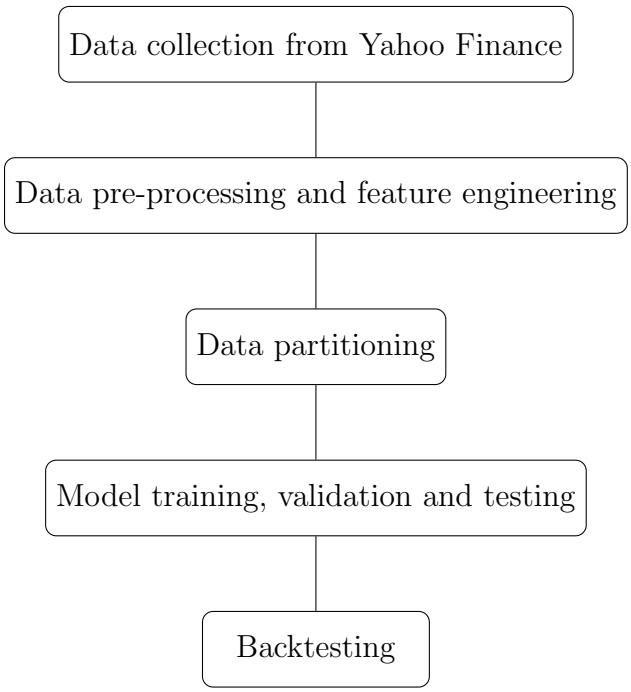


Figure 4.1: Data pipeline

4.2 Data Collection

Dataset used in this study is available on yahoo finance and it covers the period from 1997-07-01 to 2022-06-30. Daily stock prices of S&P 500 and VIX are extracted from Yahoo Finance website. Table 4.1 and 4.2 illustrates the sample data set extracted.

Date	Open	High	Low	Close	Adj Close	Volume
1997-07-01	885.140015	893.880005	884.539978	891.030029	891.030029	544190000
1997-07-02	891.030029	904.049988	891.030029	904.030029	904.030029	526970000
1997-07-03	904.030029	917.820007	904.030029	916.919983	916.919983	374680000
1997-07-07	916.919983	923.260010	909.690002	912.200012	912.200012	518780000
1997-07-08	912.200012	918.760010	911.559998	918.750000	918.750000	526010000

Table 4.1: Sample data of S&P 500

Date	Open	High	Low	Close	Adj Close	Volume
1997-07-07	18.709999	19.379999	18.440001	19.129999	19.129999	0
1997-07-08	19.080000	19.219999	18.760000	18.760000	18.760000	0
1997-07-09	18.580000	20.510000	18.570000	20.400000	20.400000	0
1997-07-10	19.889999	20.290001	19.620001	19.920000	19.920000	0
1997-07-11	19.290001	19.340000	18.719999	19.059999	19.059999	0

Table 4.2: Sample data of VIX Index

4.3 Feature Engineering

Input features of the models include:

- Daily Returns of S&P 500
- Technical indicators: ATR, BB, RSI, %K, %R, ADX, SMA, EMA, and MACD
- Sentiment indicator: Daily percentage change of VIX

Two groups of input features are formed:

- Group T: only technical indicators
- Group TS: technical and sentiment indicators

4.3.1 Daily Returns of S&P 500

S&P 500 Index or Standard & Poor's 500 Index is used to represent the aggregated US stock market. It is a market-capitalization-weighted index of 500 leading publicly traded companies in the U.S. (Kenton 2022). While it is impossible to directly invest in the S&P 500 as it is an index, investors can invest in the S&P 500 ETF, a fund that aims to duplicate the performance of the S&P 500 Index. S&P 500 has had an annualized return of about 11.5% (as of June 27) and it has generated about 8% of annualized return over the past 30 years, demonstrating it as a great first-time investment that provides significant monetary value while giving access to the 500 largest companies in the world (Kovaleski 2022).

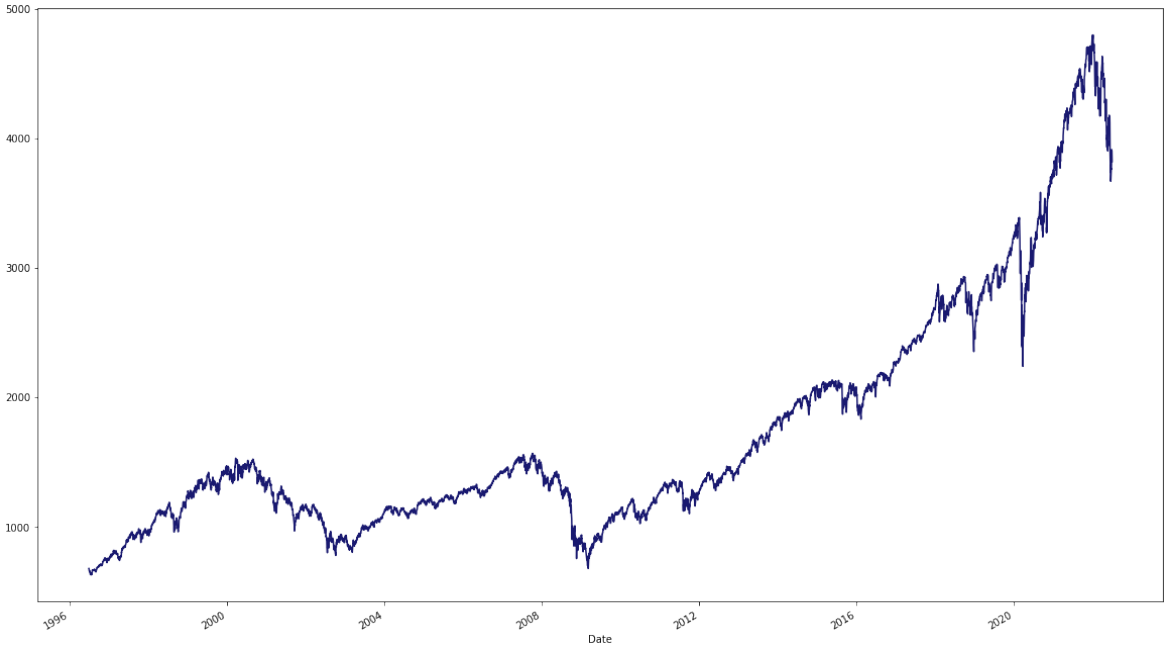


Figure 4.2: Price Movement of S&P 500

Figure 4.2 demonstrates the Price Movement of the S&P 500 from 1997-07-01 to 2022-06-30 and it suggests that the series may be non-stationary as there is no apparent pattern being observed and the variance seems to vary across time. Besides, the volatility fluctuates more near the end. Non-stationary data, as a rule, are unpredictable and cannot be modeled or predicted as it may result in a spurious result where a non-existence relationship between two variables may be indicated. Augmented Dickey-Fuller test, a unit root test for stationarity is performed to determine whether the series has a unit root. The p-value obtained from the closing price is greater than

the significance level of 0.05 and the Test Statistic is higher than any of the critical values, confirming the absence of unit root. Thereby, inferring that the series is non-stationary.

Results of Dickey-Fuller Test:
Test Statistic 0.525416
p-value 0.985625
#Lags Used 27.000000
Number of Observations Used 6263.000000
Critical Value (1%) -3.431395
Critical Value (5%) -2.862002
Critical Value (10%) -2.567016
dtype: float64

Data pre-processing is required to convert the series to stationary. This can be performed by using Daily Returns instead of the closing price that is scale-independent and statistically stationary. The return fluctuates only around a small positive value t in a properly functioning economy. Consequently, the total stock price for a company tends to grow roughly exponentially over time.

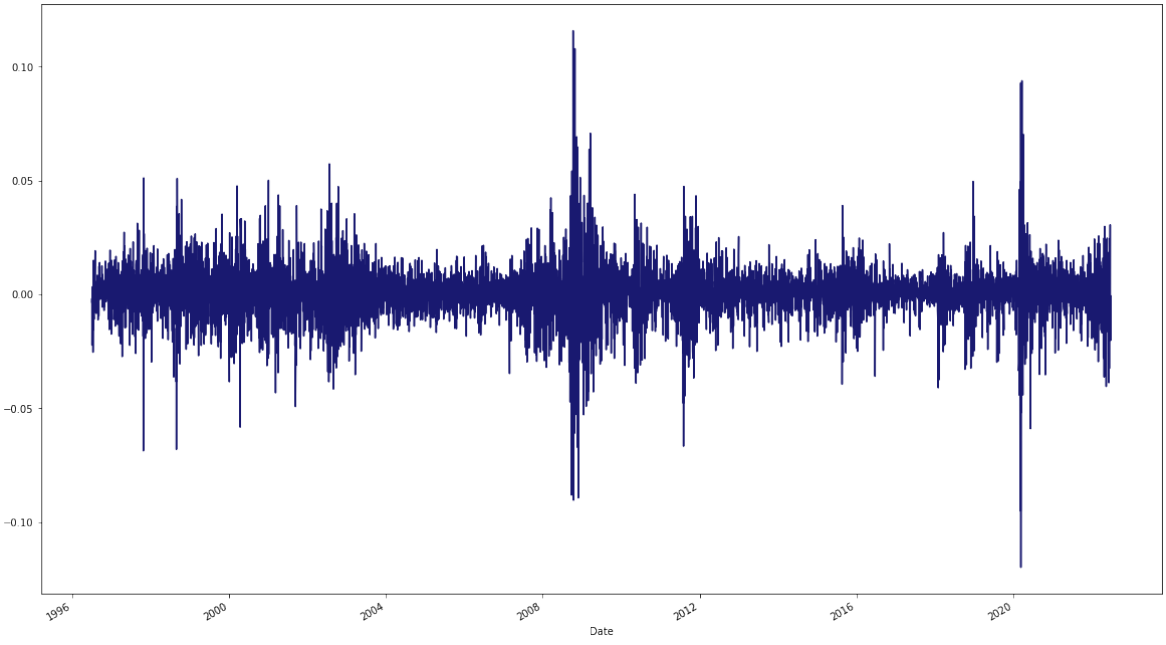


Figure 4.3: Daily Returns of S&P 500

Comparing Figures 4.2 and 4.3, it is clear to observe that the variance is more stable for Daily Returns, despite the period of financial turbulence such as the financial crisis in 2008 and the Covid-19 crisis in early 2020. The variance obtained from Close and

Daily Returns is 812964.5 and 0.000155 respectively, confirming the observation.

Results of Dickey-Fuller Test:
Test Statistic -1.473653e+01
p-value 2.610299e-27
#Lags Used 3.300000e+01
Number of Observations Used 6.256000e+03
Critical Value (1%) -3.431396e+00
Critical Value (5%) -2.862002e+00
Critical Value (10%) -2.567016e+00
dtype: float64

The p-value obtained from Daily Returns is lower than the significance level of 0.05 and the Test Statistic is lower than any of the critical values, indicating that the series is stationary. Therefore, Daily Returns is used as the dependent variable. Besides, Daily Returns can help avoid the scaling problems which can increase the model errors. The Daily Returns used in this study is not contemporaneous but with a one-day lag, T-1 by shifting the values one day forward to best reflect the available information.

4.3.2 Technical indicators

Average True Range (ATR)

ATR is a volatility indicator derived from the 14-day simple moving average to show how much an asset move. A higher ATR of a company implied higher volatility of the stock.

$$ATR = \frac{1}{n}\sum_{i=1}^{n} TR_i \tag{4.3.1}$$

TR_i = a particular true range
n = the time period employed

Bollinger Bands (BB)

BB is a volatility indicator that depicts two standard deviations above and below a simple moving average and used to measure a market's volatility and identify overbought or oversold conditions.

$$UpperBB = MA + D\sqrt{\frac{\sum_{i=1}^{n} y_j - MA^2}{n}} \tag{4.3.2}$$

$$LowerBB = MA - D\sqrt{\frac{\sum_{i=1}^{n} y_j - MA^2}{n}} \tag{4.3.3}$$

MA = moving average

D = standard deviation y = mean

Relative Strength Index (RSI)

RSI is momentum indicator that describes the current price relative to average high and low prices over a previous trading period. This indicator estimates overbought or oversold status and helps spot trend reversals, price pullbacks, and the emergence of bullish or bearish markets.

$$RSI = 100 - \frac{100}{1 + \frac{AvgGain}{AvgLoss}} \tag{4.3.4}$$

Stochastic Oscillator (%K)

%K is a momentum indicator comparing a particular closing price of a security to a range of its prices over a certain period of time to generate overbought and oversold signals, utilizing a 0–100 bounded range of values. It helps traders determine where a trend might be ending.

$$\%K = \frac{C - L14}{H14 - L14} * 100 \tag{4.3.5}$$

C = The most recent closing price

L14 = The lowest price traded of the 14 previous trading sessions

H14 = The highest price traded during the same 14-day period

%K = The current value of the stochastic indicator

Williams (%R)

%R is a momentum indicator that is very similar to the Stochastic oscillator and is used in the same way. It moves between 0 and -100 and measures overbought and oversold levels.

$$\%R = \frac{HighestHighClose}{HighestHigh - LowestLow} \tag{4.3.6}$$

Highest High = Highest price in the lookback period, typically 14 days

Close = Most recent closing price

Lowest Low = Lowest price in the lookback period, typically 14 days

Average Directional Index (ADX)

ADX is a trend indicator consisting of two accompanying indicators, the negative directional indicator (-DI) and the positive directional indicator (+DI). It helps assess whether a trade should be taken long or short, or if a trade should be taken at all.

$$+DI = \frac{Smoothed + DM}{ATR} * 100 \tag{4.3.7}$$

$$-DI = \frac{Smoothed - DM}{ATR} * 100 \tag{4.3.8}$$

$$DX = \frac{|+DI - -DI|}{|+DI + -DI|} * 100 \tag{4.3.9}$$

$$ADX = \frac{(PriorADX * 13) + CurrentADX}{14} \tag{4.3.10}$$

+DM (Directional movement) = Current High - PH

PH = Previous High

-DM = Previous Low - Current Low CDM = Current DM

ATR = Average True Range

$$Smoothed + DM = \sum_{t-1}^{14} DM - \frac{\sum_{t-1}^{14} DM}{14} + CDM \tag{4.3.11}$$

Simple Moving Average (SMA)

SMA is a trend indicator to calculate the average of a selected range of prices, usually closing prices, by the number of periods in that range. SMA with a short time frame will react much quicker to price changes than an SMA with a long look-back period. When the short SMA crosses above the long SMA, it's known as a golden signal, signalling a buy signal. Meanwhile, when the short SMA crosses below the long SMA, it's known as a death cross, signalling a sell signal.

$$SMA = \frac{A_1 + A_2 + ... + A_n}{n} \tag{4.3.12}$$

A_n = the price of an asset at period n
n = the number of total periods

Exponential Moving Average (EMA)

EMA is a trend indicator that places a greater weight and significance on the most recent data points. EMA differs from SMA by reacting more significantly to recent price changes, where SMA pplies an equal weight to all observations in the period.

$$EMA_{today} = (Value_{today} * \frac{Smoothing}{1 + Days}) + EMA_{yesterday} * (1 - (\frac{Smoothing}{1 + Days})) \tag{4.3.13}$$

Moving Average Convergence Divergence (MACD)

MACD is a trend indicator that can help signal shifts in market momentum and help signal potential breakouts. It is calculated using two exponential moving averages (EMA) - short term and long term.

MACD=12-Period EMA - 26 - Period EMA

4.3.3 Daily Returns of VIX

VIX is used as the proxy to capture the emotion that is driving the market now. VIX is a key measure of market expectations of near-term volatility conveyed by S&P 500 stock index option prices over the next 30 days (CNN 2022) as it presents a strong negative correlation with S&P 500. It is considered the barometer of sentiment and market volatility which is known as the fear index.

VIX is calculated by averaging the weighted prices of out-of-the-money puts and calls, producing a measure of constant, 30-day expected volatility of the U.S. stock market. In this study, the daily change in VIX is computed. The initial analysis of the correlation between S&P 500 and VIX is performed by observing their closing price changes over time. The data of S&P 500 and VIX is used to make a line chart as shown in Figure 4.4. In the figure, the X axis represents the time index, and the Y axis represents the closing price.

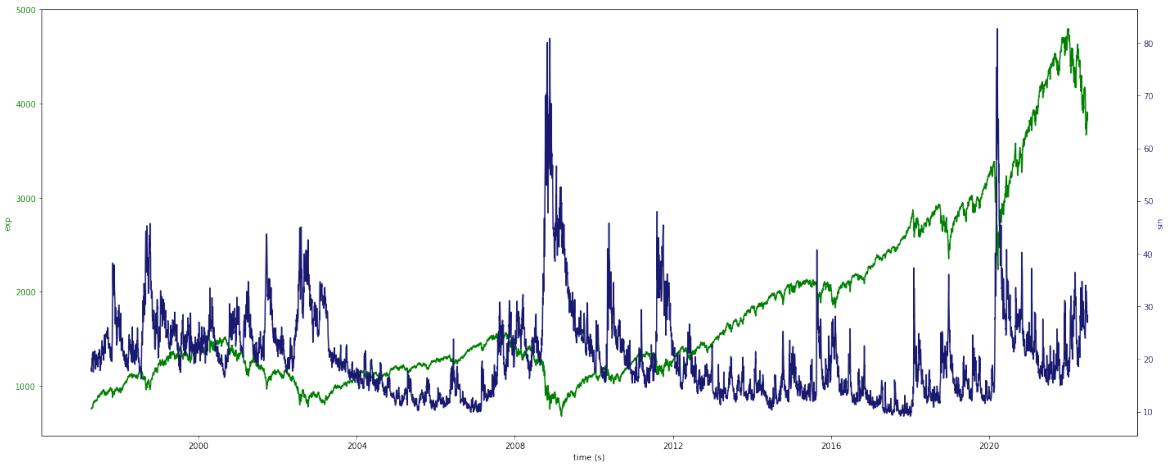


Figure 4.4: Price Movement of S&P 500 and VIX

As observed in Figure 4.4, VIX tends to increase when the market collapse, such as during the 2008 financial crisis and the Great Recession. The VIX reached a phased peak when the S&P 500 fell from October 2007 to March 2009 and dropped slightly when S&P 500 started recovering its losses from March 2009. On the other hand, the VIX usually fluctuates within a small range and shows a certain degree of decline when the market condition is stable.

Spearman correlation test is used to confirm the correlation between the log return of S&P 500 and VIX as it does not assume a distribution of the data. As a statistical

hypothesis test, the method assumes that the samples are uncorrelated (fail to reject H0). Null and alternative hypotheses are proposed as follows:

H0: Samples are uncorrelated (fail to reject H0).

H1: Samples are correlated (reject H0).

Study period	Coefficient	p-value
1997-07-01 – 2022-06-30	-0.780	0.000

Table 4.3: Results of Spearman correlation test

The statistical test reports a strong negative correlation coefficient with a value of -0.780. The p-value is close to zero, which means that the likelihood of observing the data given that the samples are uncorrelated is very unlikely with a confidence level of 95% (alpha = 0.05) that we can reject the null hypothesis that the samples are uncorrelated.

Despite this, statistical significance does not confirm the predictive ability of the models as p-values only represent how likely it is to observe a given data set under the assumption that the null hypothesis is true. The findings of correlation only merely document correlation and are not intended to express causality. Furthermore, it is unclear whether S&P 500 leads the changes in VIX or vice versa.

4.4 Data Normalization

Features used in the study represent continuous values in different ranges. Therefore, Standardization scaling (z-score is used to normalize the values by removing the mean and scaling to unit variance. Z-score transforms features to produce Gaussian-distributed values so that the mean of observed values is 0 and the standard deviation is 1.

$$z(x) = (x[:, i]u_i)/d_i \tag{4.4.1}$$

u = mean of the ith feature

d = standard deviation of the ith feature

4.5 Data Partitioning

Dataset is split into 3 – training, validation, and test sets in accordance with the validation schemes, while the test set dated from 2017-07-01 to 2022-06-30 is completely withheld from all the training processes and used to generate the predicted signal after the model is fitted and validated.

4.5.1 Single Train-Validation (STV)

Single train-validation consists of only a single validation set and it simulates the typical real-world application scenario. Dataset dated from 1997-07-01 to 2017-06-30 is split into 80% of the training set dated from 1997-07-01 to 2017-06-30 and 20% of the validation set dated from 2012-07-01 to 2017-06-30. The model is only refit once.

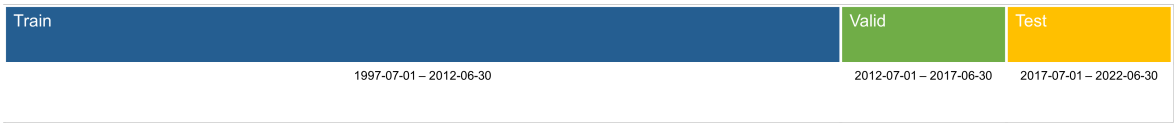


Figure 4.5: Single Train-Validation

4.5.2 K-fold Cross-Validation (KF_CV)

KF_CV has a single parameter called k that refers to the number of groups that a given data sample is to be split into. When a specific value for k is chosen, it may be used in place of k in the reference to the model, such as k=5 becoming a 5-fold cross-validation. k=5 is selected to refit the model 5 times each leaving out one-5th of the original data. For standard KF_CV, the data is partitioned randomly into k sets, and within k turns every set is used as a test set and the rest for training and this may give rise to the issue of serial correlation. To resolve this, the dataset is partitioned into 4 blocks sequentially, each consisting of 5-years of data. Each block is used as a validation set once.

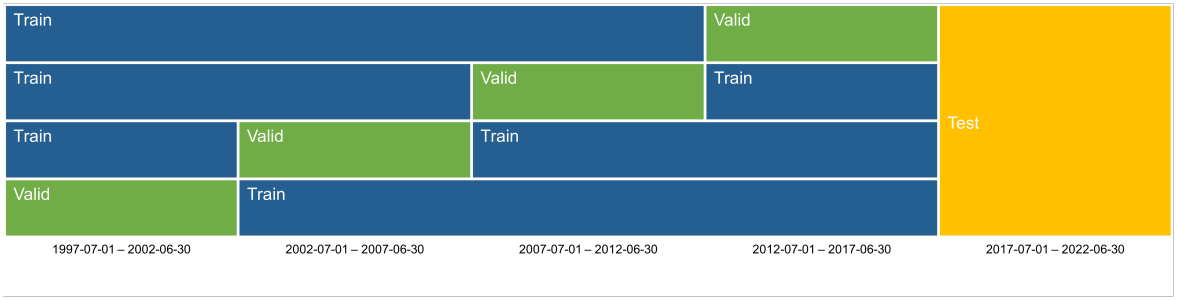


Figure 4.6: K-Fold Cross-Validation

4.5.3 Rolling Window Cross-Validation (RW_CV)

RW_CV divides the training set into two folds at each iteration on the condition that the validation set is always ahead of the training set. The dataset is partitioned into blocks by annual, each consisting of 1-year data. Each block except the first block is used as the validation set once. The oldest observation is pruned as the process of updating the model co-efficient by cleaning up the old data, making the window of each training window constant, T1, T2, T3, etc. RW_CV produces a new prediction for periods with a successively updated start date during the process of validating.

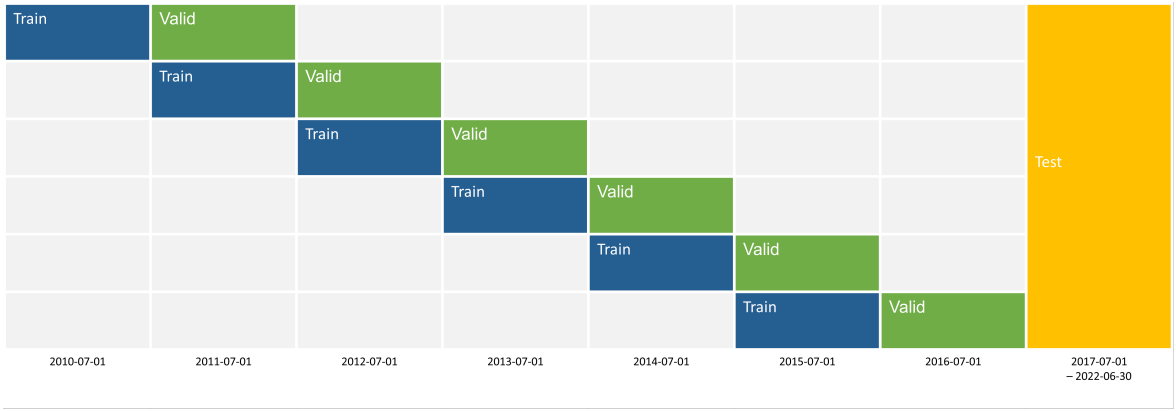


Figure 4.7: Rolling Window Cross-Validation

4.5.4 Expanding Window Cross-Validation (EW_CV)

Similar to RW_CW, the dataset is partitioned into blocks by annual, each consisting of 1-year data. Each block except the first block is used as the validation set once. Unlike RW_CV where the window of each training set remains the same, EW_CV expands the window of the training set by annual, T+1, T+2, T+N, meaning that the window of the successive training set is expanded, while the window of the validation set remains constant throughout the time.

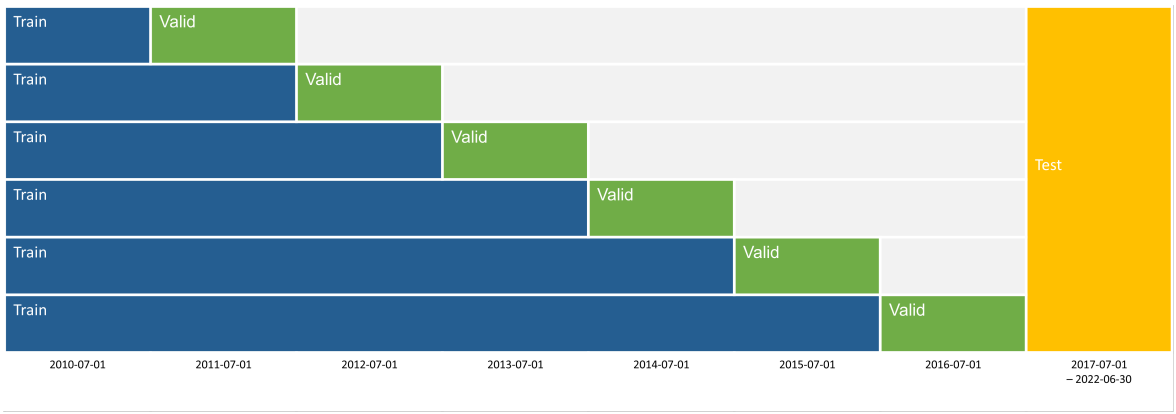


Figure 4.8: Expanding Window Cross-Validation

4.6 Machine Learning Algorithms

Classification-tree-based models including Decision Tree Classifier (DTC), Random Forest Classifier (RFC), Extra Trees Classifier (ETC), AdaBoost Classifier (ADA) and Bagging Classifier (BC) are developed to identify the upwards and downwards Price Movement signaling buy or sell indicators. The parameters of each model are tuned and evaluated over several different scenarios to determine the optimum model calibration.

4.6.1 Base Classifier: Decision Tree Classifier (DTC))

DTC is used as the base classifier for all the ensemble classifiers in this study. It is one of the most used algorithms to predict the value of the discrete-valued target in classification problems as it is fast and efficient compared to other classification algorithms. Like the tree-based algorithms, it does not require any feature transformation for non-linear data as multiple weighted combinations are not considered simultaneously. Multiple studies have proven the effectiveness of DTC in predicting stock price (Huang et al. 2019, Inamdar et al. 2019). Among the tree-based models, Decision Tree is more prone to overfitting, resulting in lower bias and higher variance which can lead to errors in the final estimation and high inaccuracy in the results. Nevertheless, DTC is extremely sensitive to the input data, and research suggests using multiple trees to ensure the reliability of the model. This provides an incentive to adopt the Random Forest comprised of a varying number of trees. This has been supported by Tsai et al. (2011)'s study demonstrating that ensemble classifiers outperformed the single classifiers in terms of prediction.

4.6.2 Random Forest Classifier (RFC)

RFC consists of many deep but decorrelated trees built on the various samples of the data. Krauss et al. (2016) discovered that the high number of decor-related trees helps achieve greater accuracy and returns. Compared to DTC, it has an even lower chance of overfitting (Cutler et al. 2011). Unlike DTC which gives high importance to a particular set of features, a RFC does not depend highly on any specific set of features, allowing it to generalize over the data in a better way and produce a more

accurate result even with noisy data. It also generates a lower variance compared to DTC. Hastie et al. (2004) considered RFC as an all-purpose model requiring even less parameter tuning than boosting. Besides, Caruana & Niculescu-Mizil (2006) claims that RFC can yield higher accuracy than algorithms such as artificial neural networks and support vector machines, supported by Inamdar et al. (2019) using Random Forest Regressor (RFR) to predict stock Price Movement. Despite the robustness of RFC, it presents a limitation of higher computational time and complicated interpretability.

4.6.3 Extremely Randomized Trees Classifier (ETC)

Similar to the RFC, ETC aggregates the results of multiple de-correlated DTC collected in a forest to output its classification result. It differs itself from other tree-based models by selecting features and split-point completely at random and growing the trees using the entire training data without bootstrapping aggregation, resulting in lower bias and variance compared to DTC and RFC (Ampomah et al. 2021) Unlike RFC which chooses the optimum split, the ETC selects split randomly. Once the split points are selected, ET choose the best one between features. Therefore, ET add randomization but still consists of optimization (Aznar 2020). Since splits are chosen at random for each feature in the ET, it demonstrated a faster execution time and lower computation cost than RFC.

4.6.4 Adaptive Boosting(ADA)

ADA is a boosting algorithm applied to any classifier giving it the ability to learn its errors and hence suggest an accurate and better model for future use as an ADA example. It is most commonly used with DTC with one level to enhance the performance of multiple weak learners by re-weighting the original training data through iteration to achieve arbitrarily high accuracy (Ampomah et al. 2020). Misclassified item is assigned a higher weight so that it appears in the training data set of the next classifiers with higher probability. Boosting relies on the weak learners generally results in the highest accuracy. Coming to the advantages, ADA is less prone to overfitting as the input parameters are not jointly optimized. It also requires lesser parameter tuning unlike algorithms like Support Vector Machines (SVC). ADA uses a progressively learning

boosting technique and therefore, high-quality data is needed. Besides, the trade-off of yielding a high degree of precision is that ADA is extremely sensitive to noise and outliers, which requires the elimination of these factors before using the data. Compared to XGBoost, it presents a higher computational time due to the slow learning process resulting from the model spending too much time on learning extreme cases and skewing results (Kaggle 2021).

4.6.5 Bagging Classifier(BC)

BC, also known as bootstrapping aggregation is an ensemble meta-estimator that fits base Classifiers each on random subsets of the original dataset and then aggregates their individual predictions (either by voting or by averaging) to form a final prediction (scikit learn.org 2022). Homogeneous weak learners are often considered, where BC would learn them independently from each other in parallel and combines them using deterministic averaging process. Breiman (2004) considers BC can effectively reduce the variances without making the predictions biased of the model by performing attributes selection. This is particularly useful for models which tend to overfit the dataset, such as DTC. Due to the high instability of DTC, slight changes in the dataset could result in substantial changes in its structure, and BC can effectively address the (Bühlmann 2012).

4.7 Evaluation Methods and Measures

Several of the more frequently used evaluation metrics are considered:

4.7.1 Accuracy Score

Accuracy score is the sum of true positives and true negatives divided by the total number of samples.

$$Accuracyscore = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.7.1}$$

TP = True positive
FP = False positive
FN = False negative
TN = True negative

4.7.2 Precision

Precision is the ratio tp / (tp + fp) where tp is the number of true positives and fp the number of false positives. The precision is intuitively the ability of the classifier not to label as positive a sample that is negative.

$$Precision = \frac{TP}{TP + FP} \tag{4.7.2}$$

4.7.3 Recall

Recall is the ratio tp / (tp + fn) where tp is the number of true positives and fn the number of false negatives. The recall is intuitively the ability of the classifier to find all the positive samples.

$$Precision = \frac{TP}{TP + FN} \tag{4.7.3}$$

4.7.4 F1-score

F1-score is a harmonic mean of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0. The relative contribution of precision and recall to the F1 score are equal.

$$Precision = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4.7.4}$$

4.7.5 Sharpe ratio

Sharpe ratio adjusts a portfolio's past performance—or expected future performance—for the excess risk that was taken by the investor.

$$Sharperatio = \frac{R_p - R_f}{D_p} \tag{4.7.5}$$

R_p= Average rate of returns

R_f= Risk-free rate of returns

D_p= Standard deviation of the returns

Chapter 5

Evaluation and Discussion

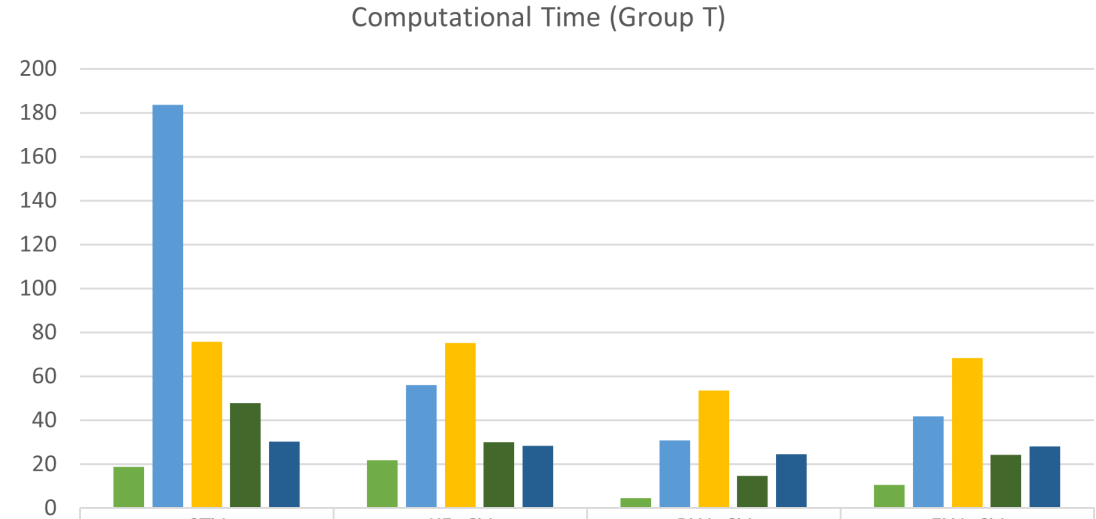
A number of models developed using different model selection procedures are tested using the three dimensions of analysis; i) computational efficiency, ii) accuracy score, precision, recall and f1-score for the examination of class imbalance, and iii) strategy performance using Sharpe ratio. Besides, the impact of feature, and the predictive power of VIX are analysed. The research questions are answered at the end of every section.

5.1 Evaluation of Validation Schemes

5.1.1 Computational Efficiency

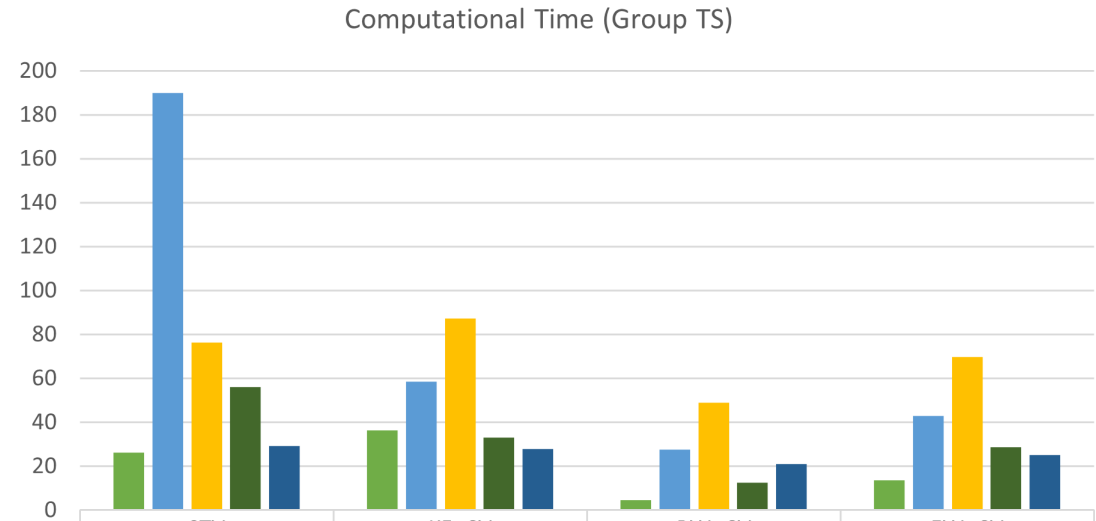
Figure 5.1 demonstrates the computational time of four validation schemes; clearly, the STV and KF_CV methods are more computationally expensive. In general, they record the longest computational time as both methods use a larger amount of training data during model fitting as compared to RW_CV and EW_CV, especially the KF_CV which repeats 5 times in the 5-fold cross-validation process as each sample is given the opportunity to be used in the test set 1 time and used to train the model 4 times.

RW_CV results in the shortest computational time due to the shortest window length of its training set, which consists of only 1-year data each. EW_CV yields a slightly longer computational time compared to RW_CV. Despite the window length of its initial training sets being of the same size as RW_CV, the window length of the subsequent training sets increases consecutively and the last training set is expanded to the extent of consisting of all available training data. Existing studies raise concerns about the higher computational time of RW_CV and EW_CV as they cross-validate



	STV	KF_CV	RW_CV	EW_CV
DTC	18.72	21.95	4.5	10.52
RFC	183.6	55.93	30.79	41.92
ETC	75.86	75.21	53.67	68.29
ADA	47.83	29.91	14.71	24.22
BC	30.42	28.32	24.59	28.07

Figure 5.1: Computation Efficiency (Group T)

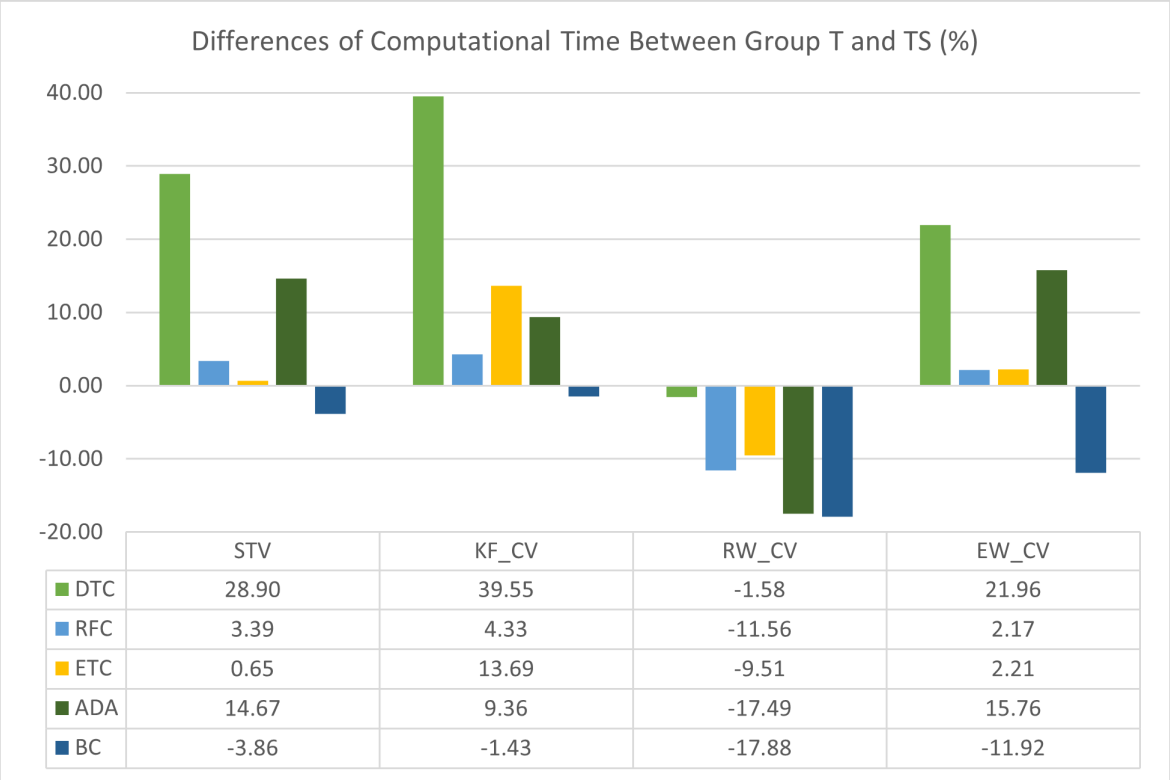


	STV	KF_CV	RW_CV	EW_CV
DTC	26.33	36.31	4.43	13.48
RFC	190.05	58.46	27.6	42.85
ETC	76.36	87.14	49.01	69.83
ADA	56.05	33	12.52	28.75
BC	29.29	27.92	20.86	25.08

Figure 5.2: Computation Efficiency (Group TS)

many times using different training sets (Qiang & Shen 2021, Chou & Nguyen 2018). Nonetheless, it must be pointed out that the window length of each training set plays a crucial role in determining the computational time, and that researchers have been

examining the effectiveness of using simple statistical methods and extensive search-based methods to find out the most optimal windows (Iqbal et al. 2019).



	STV	KF_CV	RW_CV	EW_CV
DTC	28.90	39.55	-1.58	21.96
RFC	3.39	4.33	-11.56	2.17
ETC	0.65	13.69	-9.51	2.21
ADA	14.67	9.36	-17.49	15.76
BC	-3.86	-1.43	-17.88	-11.92

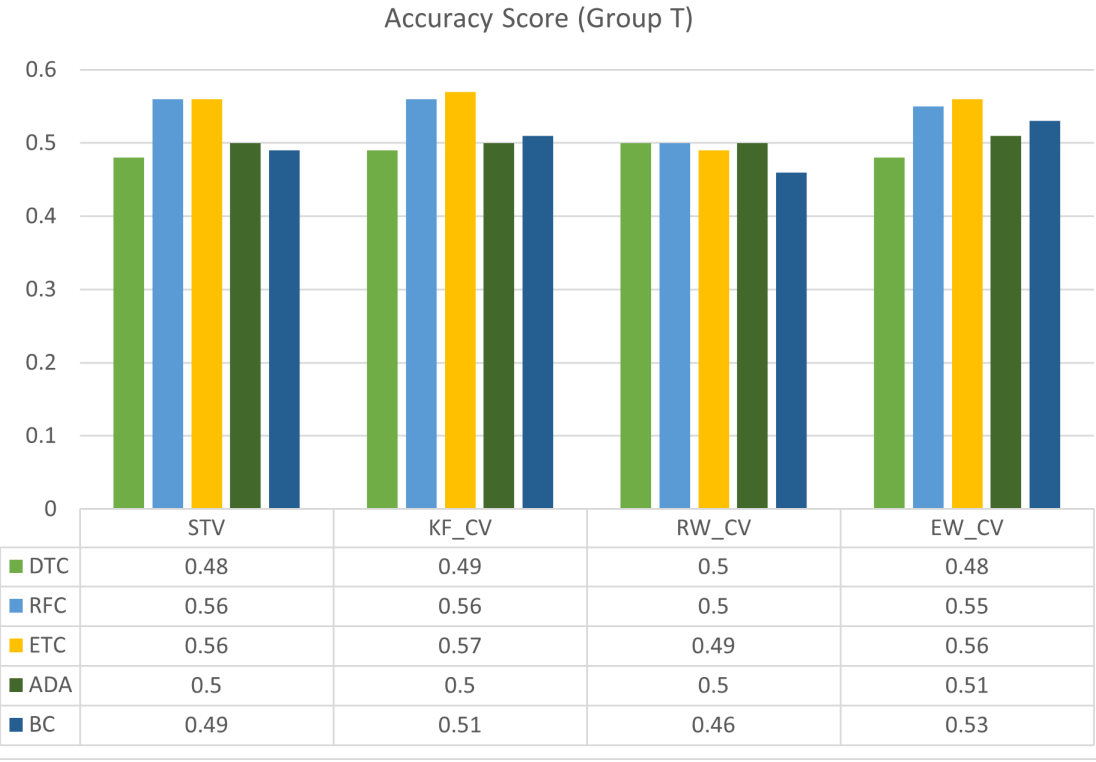
Figure 5.3: Percentages Differences of Computational Time Between Group T and TS (%)

The computational time increases along with the increase in the number of features, except for RW_CV. It is interesting to observe that all models in RW_CV demonstrate a shorter computational time in Group TS when the number of features increase, as shown in Figure 5.3.

5.1.2 Accuracy Score, Precision, Recall and F1-Score

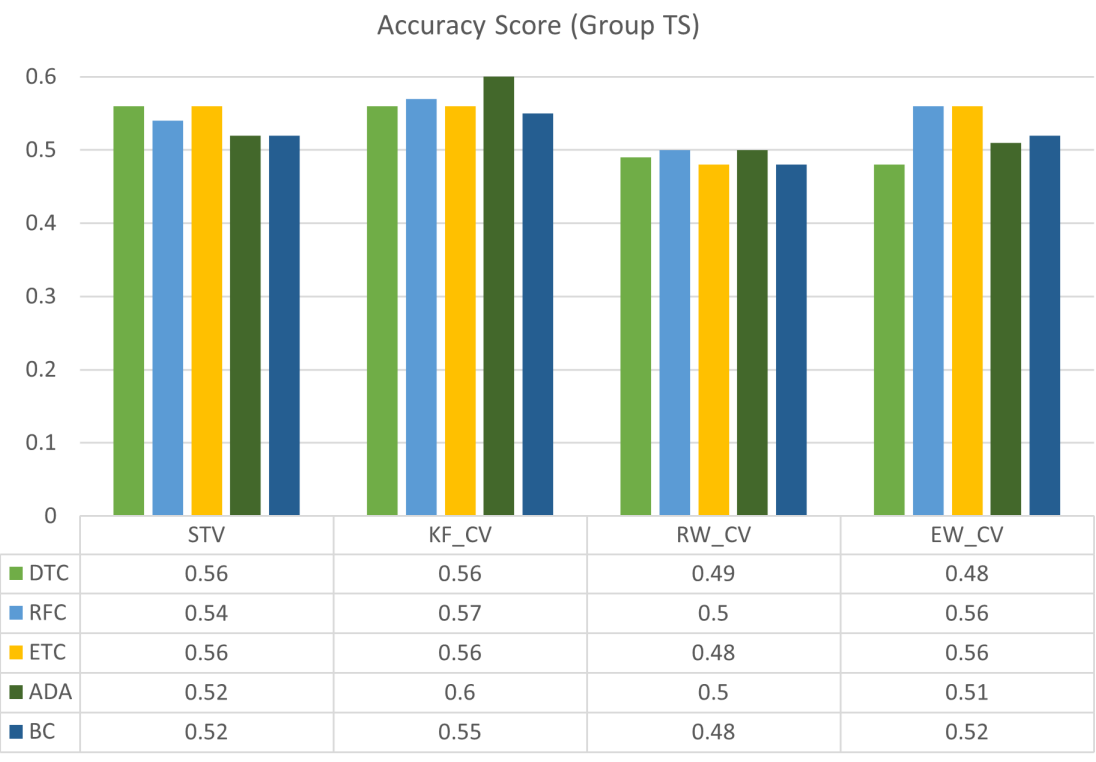
The dataset used in this study is slightly skewed where it is dominated by a positive signal, +1, signaling a high chance for the models to be biased towards the majority class. Therefore, apart from the accuracy score, it is crucial to also analyze the precision, recall, and F1 score. While the accuracy score looks at correctly classified observations in both majority and minority classes, it might be misleading especially when the dataset is not well balanced. Hence, precision and recall, and the f1-score, which investigates the balance of precision and recall in both classes, is more applicable in this study as by default the models would be extremely good at predicting only one

class, resulting in high, but biased accuracy.



	STV	KF_CV	RW_CV	EW_CV
DTC	0.48	0.49	0.5	0.48
RFC	0.56	0.56	0.5	0.55
ETC	0.56	0.57	0.49	0.56
ADA	0.5	0.5	0.5	0.51
BC	0.49	0.51	0.46	0.53

Figure 5.4: Accuracy Score (Group T)

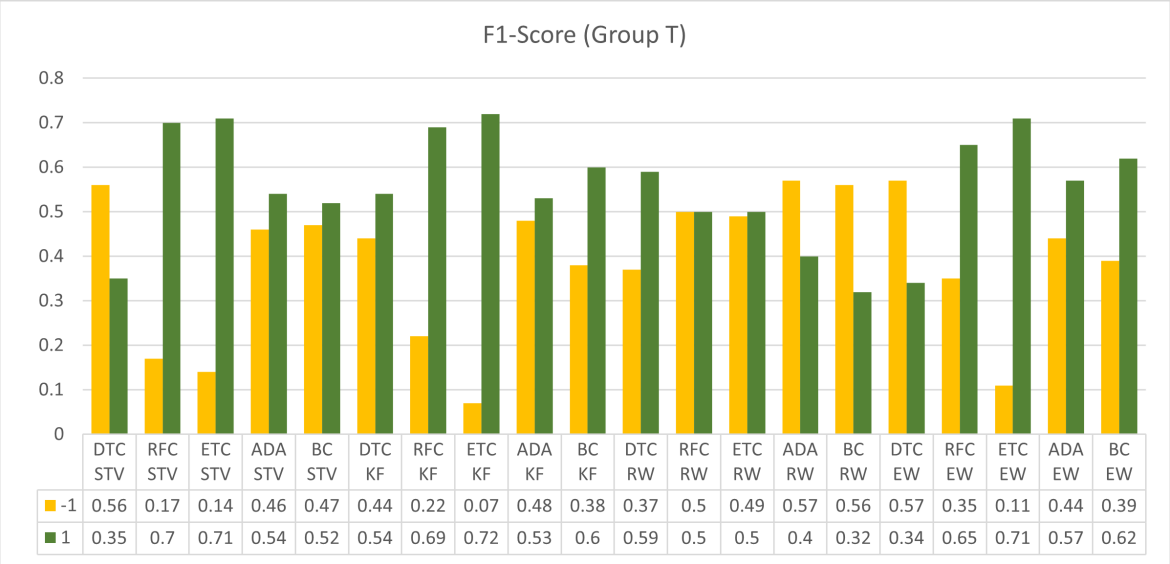


	STV	KF_CV	RW_CV	EW_CV
DTC	0.56	0.56	0.49	0.48
RFC	0.54	0.57	0.5	0.56
ETC	0.56	0.56	0.48	0.56
ADA	0.52	0.6	0.5	0.51
BC	0.52	0.55	0.48	0.52

Figure 5.5: Accuracy Score (Group TS)

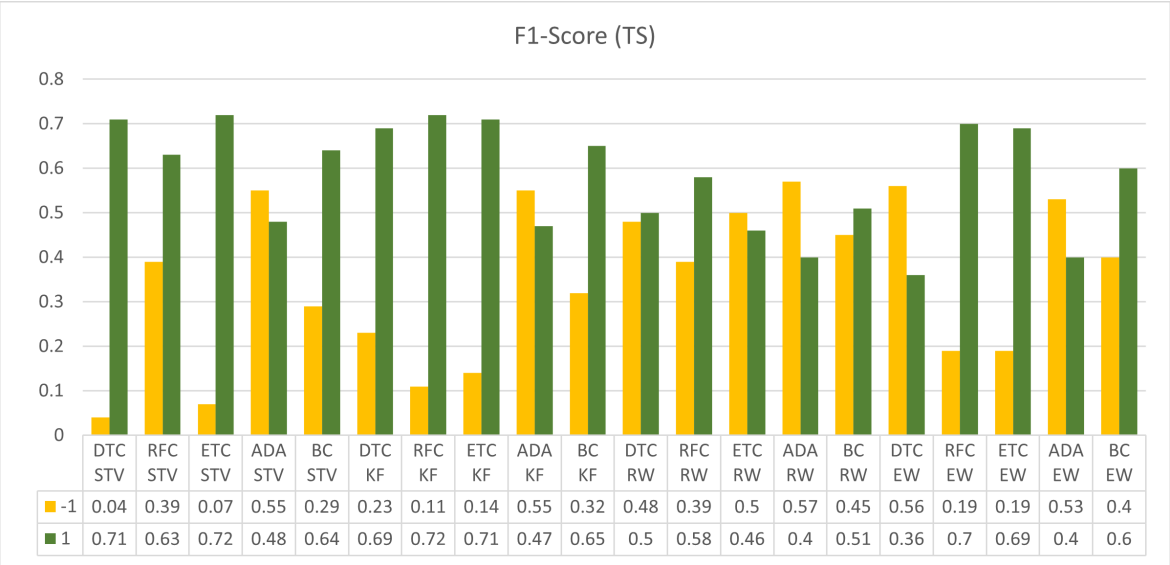
Figure 5.4 and 5.5 show that most models in STV and KF_CV achieve a slightly higher accuracy score compared to RW_CV and EW_CV. It is worth pointing out that all models in KF_CV achieve the highest accuracy scores of over 0.55 to 0.60 in Group TS. The outperformance of STV and KF_CV can be attributed to the larger samples in their training sets as larger samples tend to produce a prediction that better represents the population parameters (Asiamah et al. 2017). On the other hand, the initial training set of RW_CV and EW_CV are smaller, which might result in a lower accuracy score on the first few iterations.

Contrastingly, RW_CV achieves the lowest accuracy score, followed by EW_CV. This might be due to the smaller samples of their initial training sets. Wang & Li (2008) point out that the smaller training size would deteriorate the model performance when there is a large number of features as trees are more easily influenced by redundant features when the sample size is small. The feature importance scores show that only three to four features are weighted significantly and used to perform splits, indicating that there are quite a few redundant features in each model, that may hinder the model performance, thus reducing the accuracy score.



	DTC STV	RFC STV	ETC STV	ADA STV	BC STV	DTC KF	RFC KF	ETC KF	ADA KF	BC KF	DTC RW	RFC RW	ETC RW	ADA RW	BC RW	DTC EW	RFC EW	ETC EW	ADA EW	BC EW
-1	0.56	0.17	0.14	0.46	0.47	0.44	0.22	0.07	0.48	0.38	0.37	0.5	0.49	0.57	0.56	0.57	0.35	0.11	0.44	0.39
1	0.35	0.7	0.71	0.54	0.52	0.54	0.69	0.72	0.53	0.6	0.59	0.5	0.5	0.4	0.32	0.34	0.65	0.71	0.57	0.62

Figure 5.6: F1-Score (Group T)



	DTC STV	RFC STV	ETC STV	ADA STV	BC STV	DTC KF	RFC KF	ETC KF	ADA KF	BC KF	DTC RW	RFC RW	ETC RW	ADA RW	BC RW	DTC EW	RFC EW	ETC EW	ADA EW	BC EW
-1	0.04	0.39	0.07	0.55	0.29	0.23	0.11	0.14	0.55	0.32	0.48	0.39	0.5	0.57	0.45	0.56	0.19	0.19	0.53	0.4
1	0.71	0.63	0.72	0.48	0.64	0.69	0.72	0.71	0.47	0.65	0.5	0.58	0.46	0.4	0.51	0.36	0.7	0.69	0.4	0.6

Figure 5.7: F1-Score (Group TS)

To determine the robustness of each validation scheme in handling class imbalance, f1-score is examined. Despite STV and KF_CV achieving an overall higher accuracy score; the huge differences in the f1-score of both classes show that they are not capable of dealing with class imbalance. The situation is worse for DTC, RFC, and ETC as their performance is strongly related to the actual distribution of the dataset. Several reasons may contribute to this result. First, majority votes tend to go to the majority class in the skewed training sets, achieving an accuracy equal to the proportion of test cases belonging to the majority class.

STV demonstrates 11.39% of differences in class distribution where positive signal, +1 dominates the training set. KF_CV shows a range from 9.82% to 15.17% of differences in class distribution across 4 training sets, averaged to 11.65%. It is not unexpected that the majority class achieves a higher f1 score in this scenario. Second, there are a lesser number of fittings in STV and KF_CV, where STV is only fitted once and KF_CV is fitted 4 times.

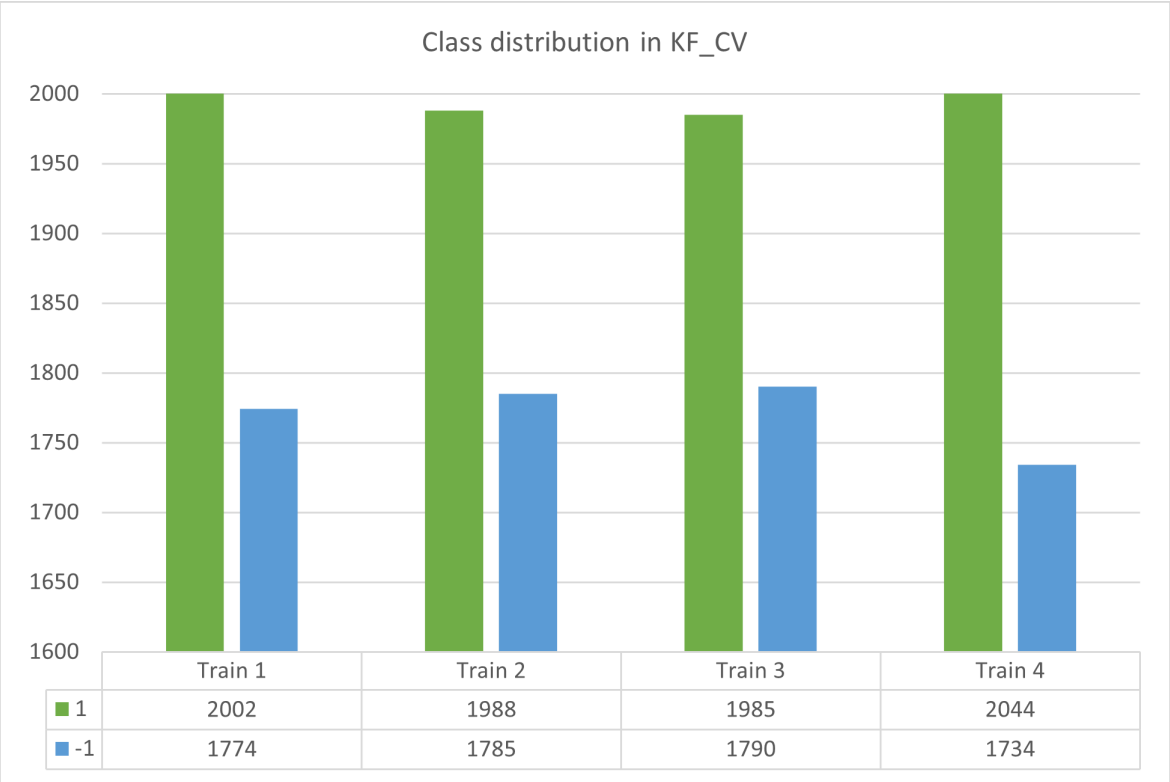


Figure 5.8: Class Distribution of KF_CV

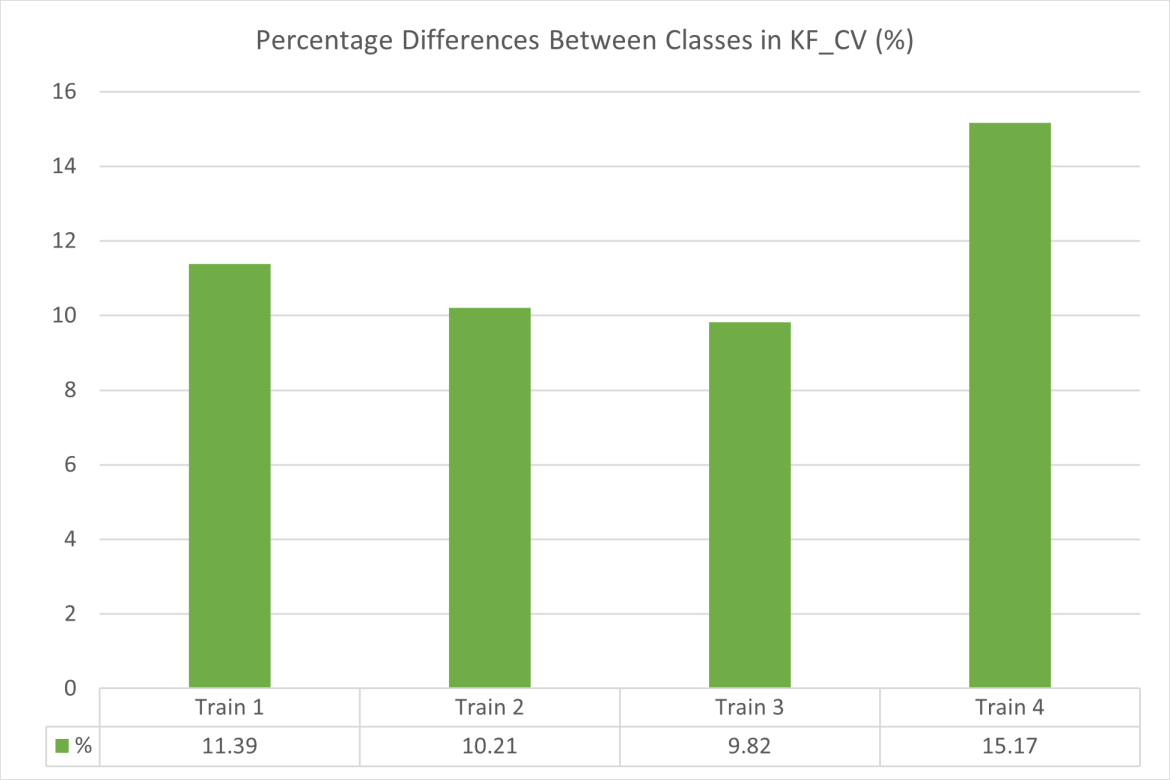
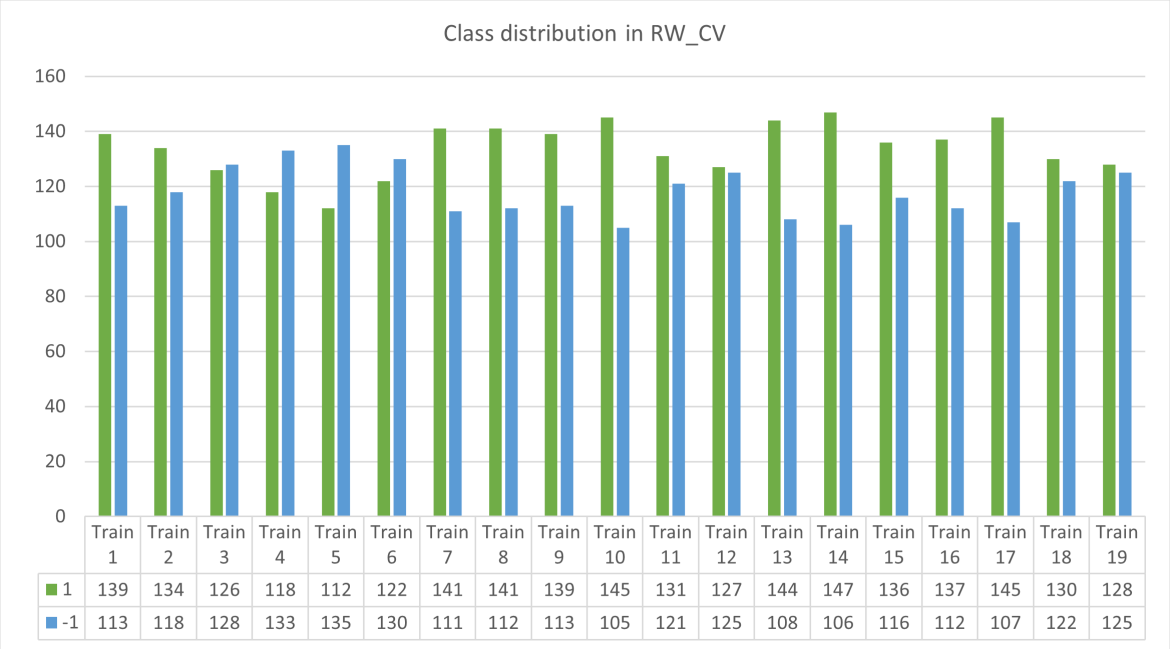


Figure 5.9: Percentage Differences Between Classes in KF_CV (%)

Despite the lowest accuracy score achieved by RW_CV, most models attain a moderate, but equally balanced f1-score in both classes. A larger number of training sets are generated, and the class distribution of each training set varies. Figure 5.10 demonstrates that out of 19 training sets, 11 sets are dominated by positive signals, 2 sets by negative signals, and 6 sets are nearly balanced. We speculate that the differences in the class distribution of each training set might be the reason behind its balanced f1 score. The impact of several skewed training sets may be offset by the more equally distributed training sets, or it is a coincidence that the class distribution of the test set is in resemblance to the training sets.



	Train 1	Train 2	Train 3	Train 4	Train 5	Train 6	Train 7	Train 8	Train 9	Train 10	Train 11	Train 12	Train 13	Train 14	Train 15	Train 16	Train 17	Train 18	Train 19
1	139	134	126	118	112	122	141	141	139	145	131	127	144	147	136	137	145	130	128
-1	113	118	128	133	135	130	111	112	113	105	121	125	108	106	116	112	107	122	125

Figure 5.10: Class Distribution of RW_CV

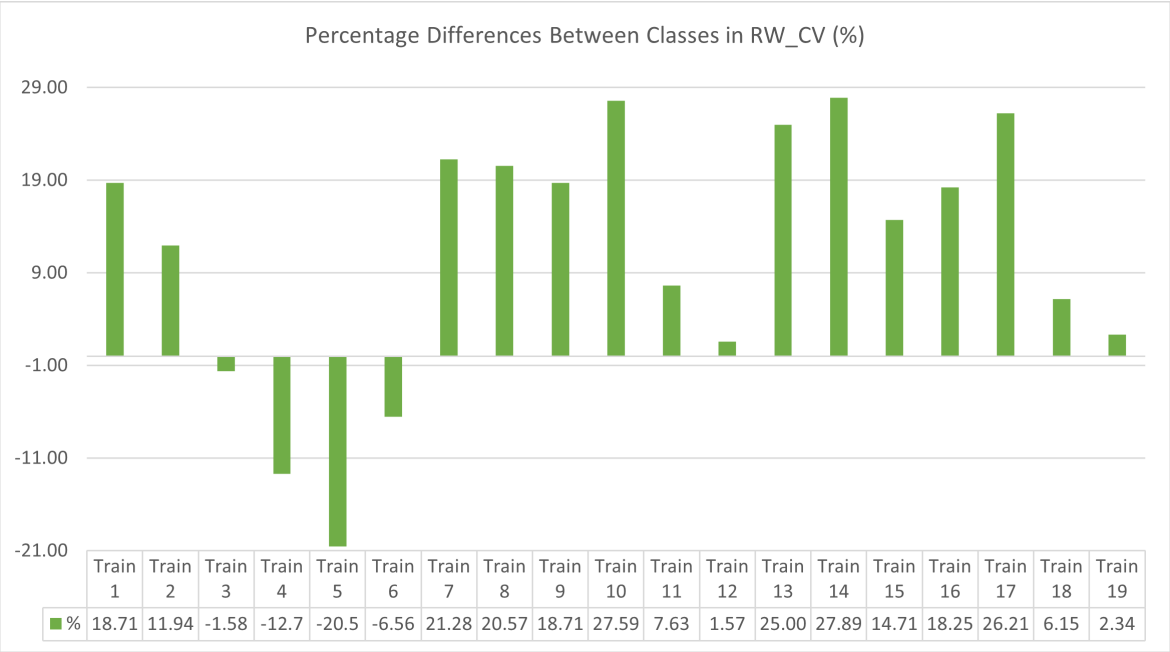


Figure 5.11: Percentage Differences Between Classes in RW_CV (%)

By comparing the f1-score of RW_CV and EW_CV, it is observed that RW_CV outperforms EW_CV despite its higher accuracy score. Each training set in EW_CV consists of data from the previous training set as the window length of the successive training set is expanded without pruning the oldest data. Figure 5.12 shows that 18 out of 19 training sets are dominated by positive signals, which can explain the reason most models are skewed. Besides, this finding also led to the understanding that while larger samples in the training set may improve the accuracy score, it might hinder the model's ability to tackle the class imbalance issue.

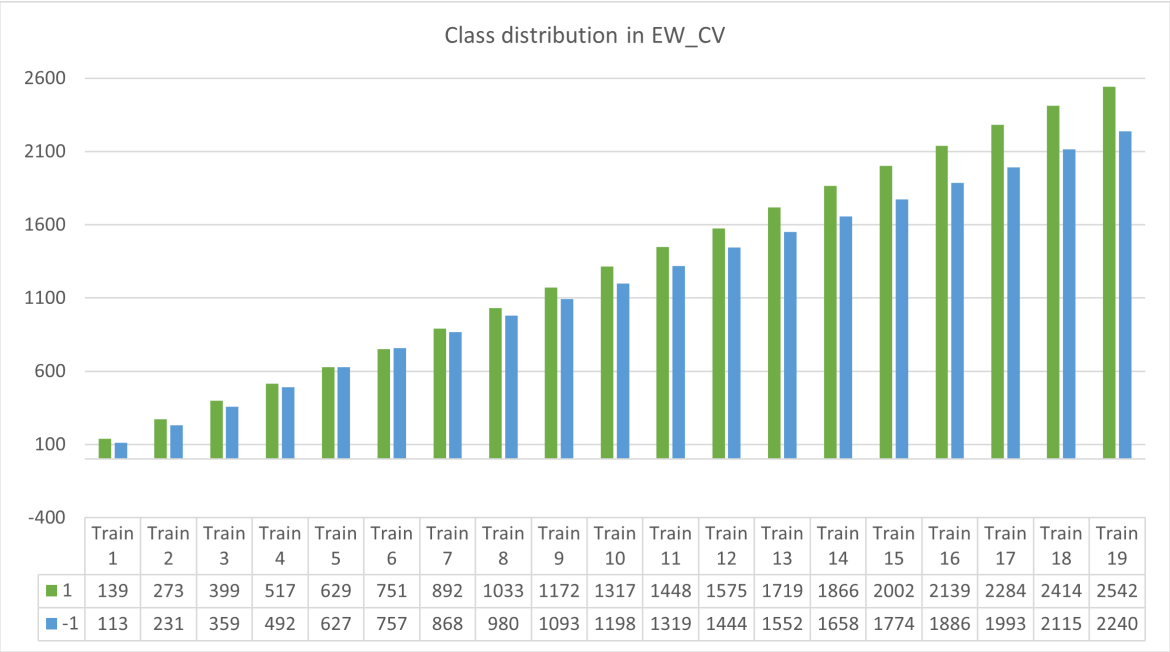


Figure 5.12: Class Distribution of EW_CV

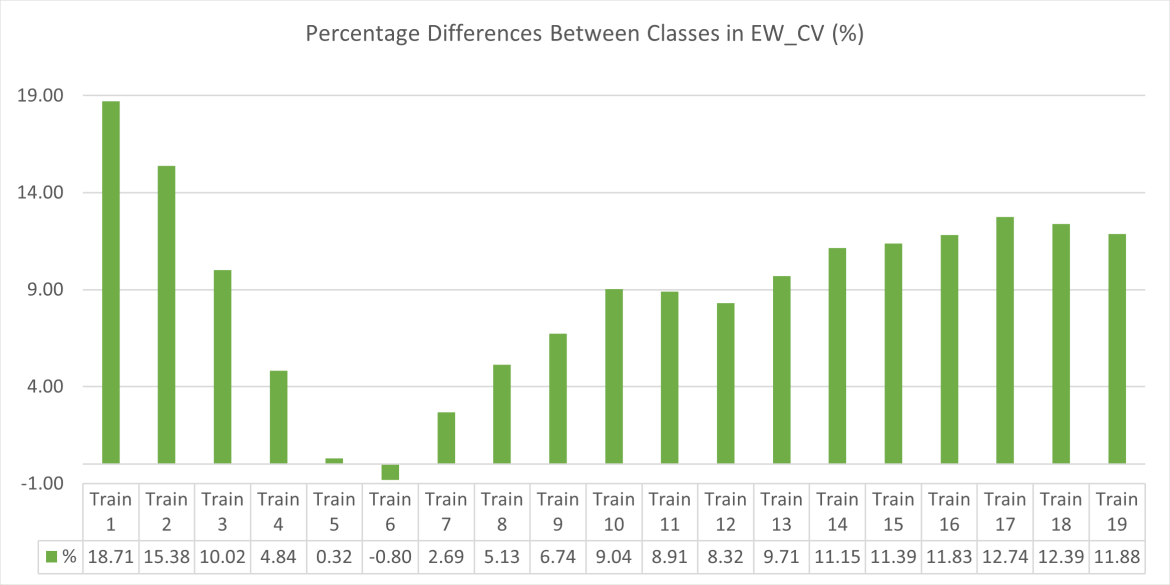
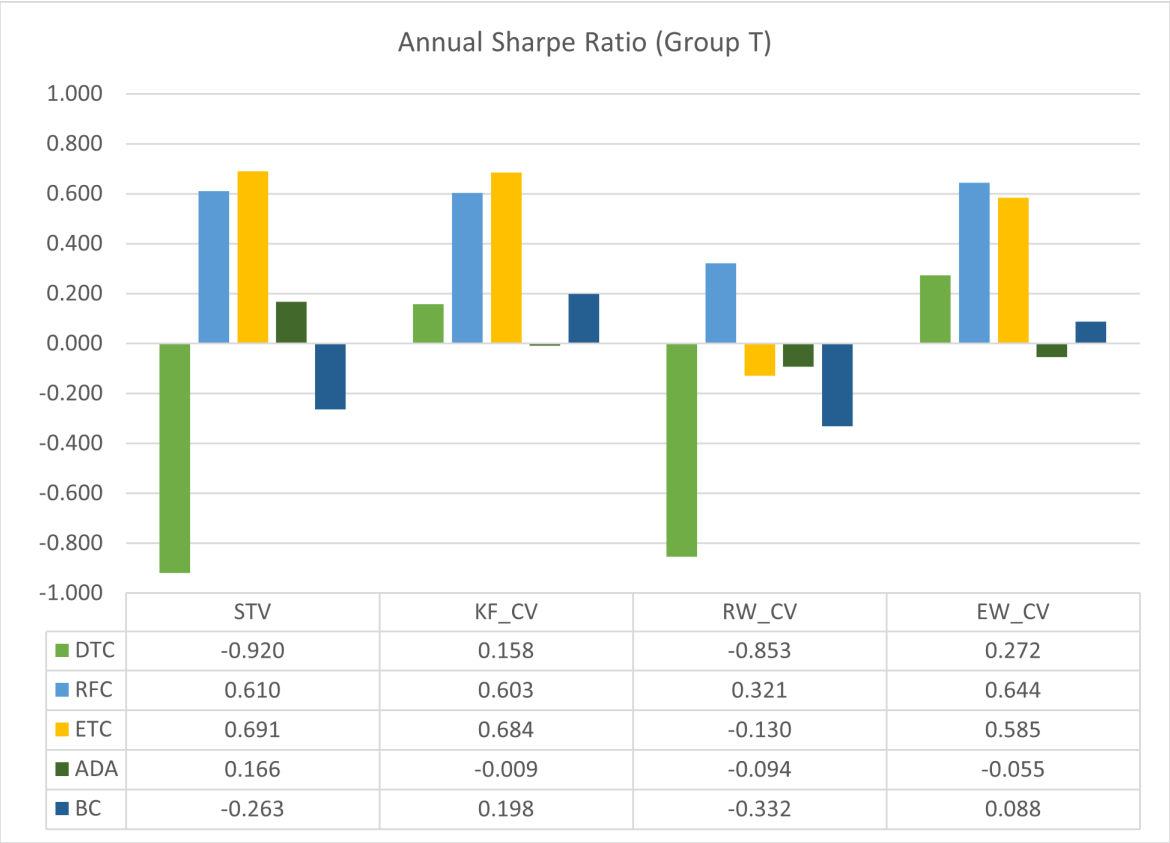


Figure 5.13: Percentage Differences Between Classes in EW_CV (%)

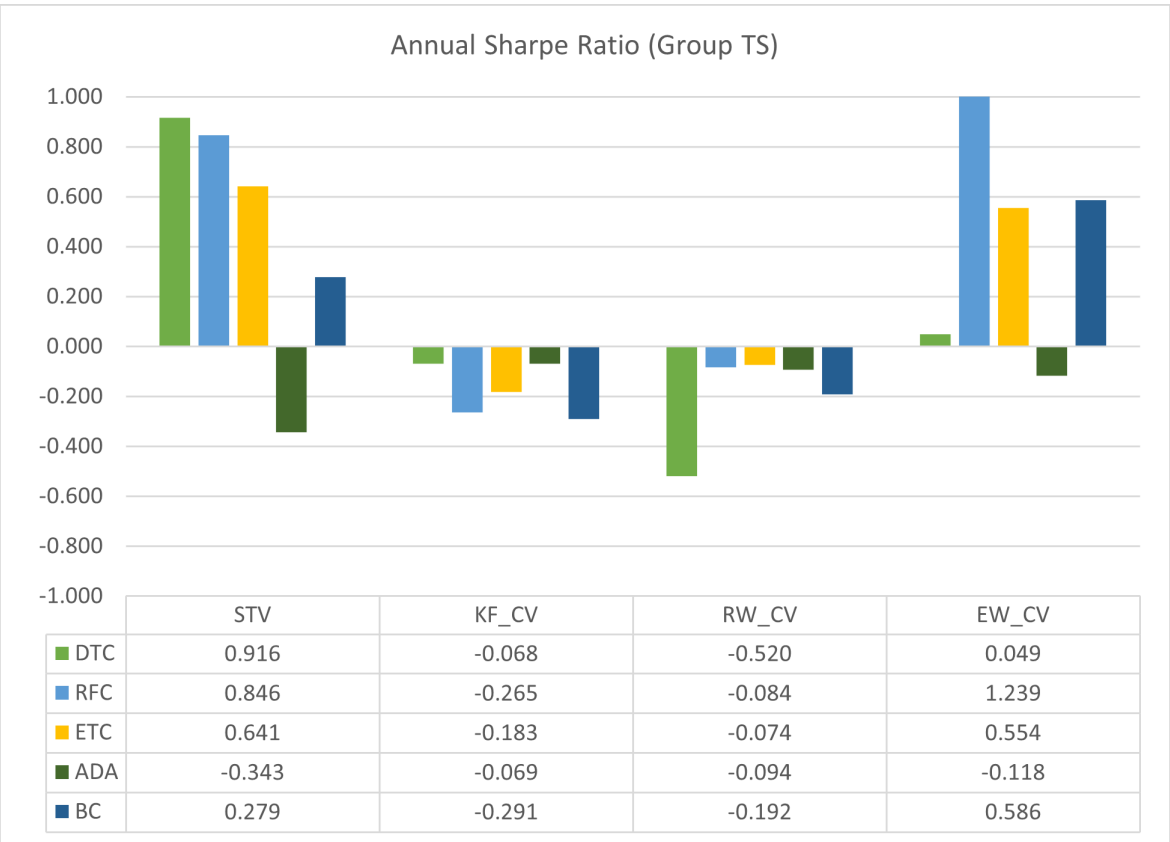
5.1.3 Strategy Performance

STV and EW_CV outperform KF_CV and RW_CV where more models achieve a positive Sharpe ratio in Group T and TS. RW_CV consistently demonstrates the worst performance in Group T and TS.



	STV	KF_CV	RW_CV	EW_CV
DTC	-0.920	0.158	-0.853	0.272
RFC	0.610	0.603	0.321	0.644
ETC	0.691	0.684	-0.130	0.585
ADA	0.166	-0.009	-0.094	-0.055
BC	-0.263	0.198	-0.332	0.088

Figure 5.14: Annual Sharpe Ratio (Group T)



	STV	KF_CV	RW_CV	EW_CV
DTC	0.916	-0.068	-0.520	0.049
RFC	0.846	-0.265	-0.084	1.239
ETC	0.641	-0.183	-0.074	0.554
ADA	-0.343	-0.069	-0.094	-0.118
BC	0.279	-0.291	-0.192	0.586

Figure 5.15: Annual Sharpe Ratio (Group TS)

RQ 1: How effective is the S&P 500 stock price prediction using the proposed validation schemes?

In a theoretical proof, we have shown that time-series cross-validation schemes including RW_CV and EW_CV are more computationally efficient compared to the traditional approaches - STV and KF_CV which requires a larger training sets and they may only be suitable for the application using a smaller training set. Stock price prediction often involves a large amount of data, for instance, S&P 500 started trading in 1957 and the amount of its historical price data is so enormous that would be extremely costly for a company, or an investor to develop the models. Evaluating their accuracy and f1-scores also reveals that they are more suitable for the application where accuracy is the primary concern without considering how the data is distributed, and there is no major downside to predicting false negatives as they can produce a higher accuracy score than RW_CV and EW_CW. However, in the case of stock trading, the cost of predicting false negatives is high as it may cost investors to lose millions of dollars.

RW_CV is the least computationally expensive, which makes it a suitable validation scheme for the application that requires a large amount of data, such as in our case the stock price prediction. Besides, it is more robust when it comes to class imbalance, where it effectively improves the ability of models to capture the minority class, as reflected in the highly balanced f1-score achieved.

It is important to highlight that strategy performance is more crucial to determining whether a model can enhance portfolio performance in a real-life setting. A model that excels in all the evaluation metrics does not always guarantee its profitability when it is back-tested in a real-life setting.

In terms of the strategy performance, STV is one of the top performers as most of its models yield a positive Sharpe ratio, and it can be quite robust when coupled with stable tree-based models like RFC and ETC so the performance would not be easily affected by changes in input feature. In the meantime, RW_CV achieves a negative Sharpe ratio in most models, indicating that it lacks the capabilities to enhance the profitability of the trades in real-life trading. EW_CV as a similar alternative to the RW_CV without pruning the oldest observation achieves a significantly higher Sharpe ratio compared to RW_CV despite lacking the ability to handle class imbalance issues.

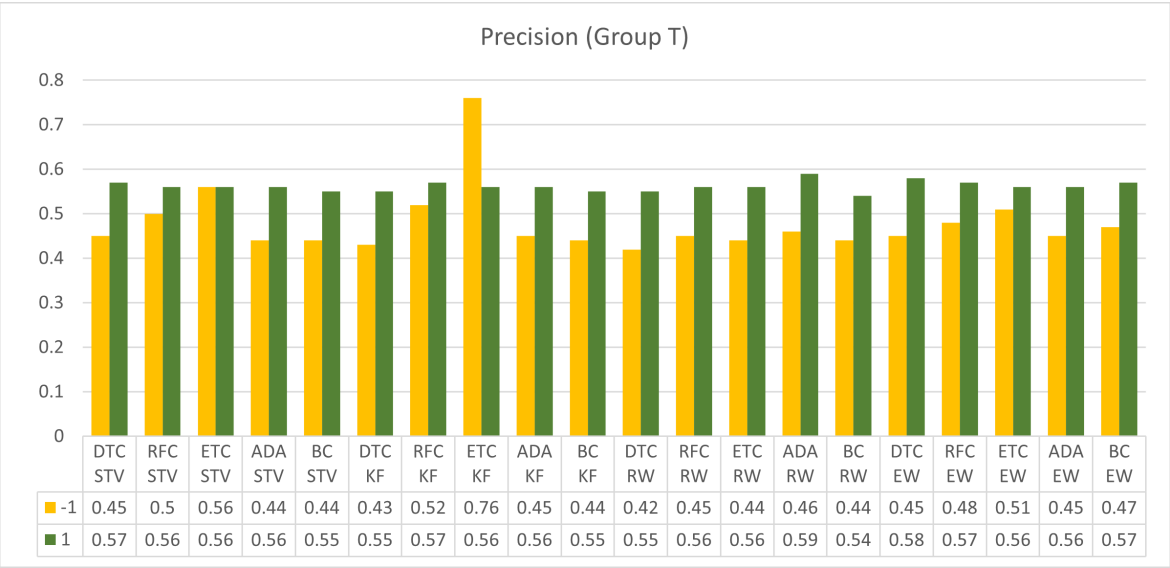
5.2 Evaluation of Tree-Based Models

5.2.1 Computational Efficiency

DTC takes the shortest computational time, regardless of the validation schemes. An explanation for this is because DTC typically makes greedy decisions in the fitting process at each stage - they optimize the sub-problem for finding an optimal split with the data in the given node and keep moving forward in the fitting process, resulting in a smaller subset of data after each split. However, the increased number of features increases the computational time, which is empirically proven and illustrated in Figure 5.2. On the contrary, RFC is the most computational expensive especially due to the presence of a large number of trees compared to DTC, and the derive of the optimal split point. While having a lot in common with RFC, ETC is slightly faster as it randomly chooses the split point and does not calculate the optimal one. Another promising finding is that ADA and BC are less computationally expensive than RFC and ETC.

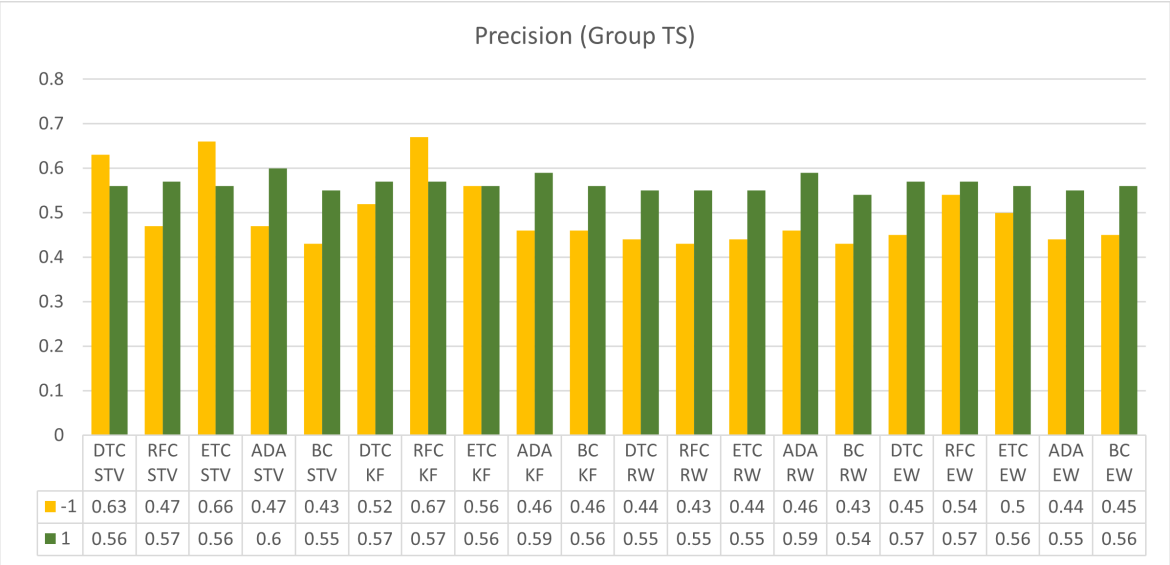
5.2.2 Accuracy Score, Precision, Recall and F1-Score

Figure 5.4 and 5.5 demonstrate that all models present very similar accuracy score lying within the range of 0.46 and 0.60. ADA_KF_CV achieves a slightly higher accuracy score of 0.60.



	DTC STV	RFC STV	ETC STV	ADA STV	BC STV	DTC KF	RFC KF	ETC KF	ADA KF	BC KF	DTC RW	RFC RW	ETC RW	ADA RW	BC RW	DTC EW	RFC EW	ETC EW	ADA EW	BC EW
-1	0.45	0.5	0.56	0.44	0.44	0.43	0.52	0.76	0.45	0.44	0.42	0.45	0.44	0.46	0.44	0.45	0.48	0.51	0.45	0.47
1	0.57	0.56	0.56	0.56	0.55	0.55	0.57	0.56	0.56	0.55	0.55	0.56	0.56	0.59	0.54	0.58	0.57	0.56	0.56	0.57

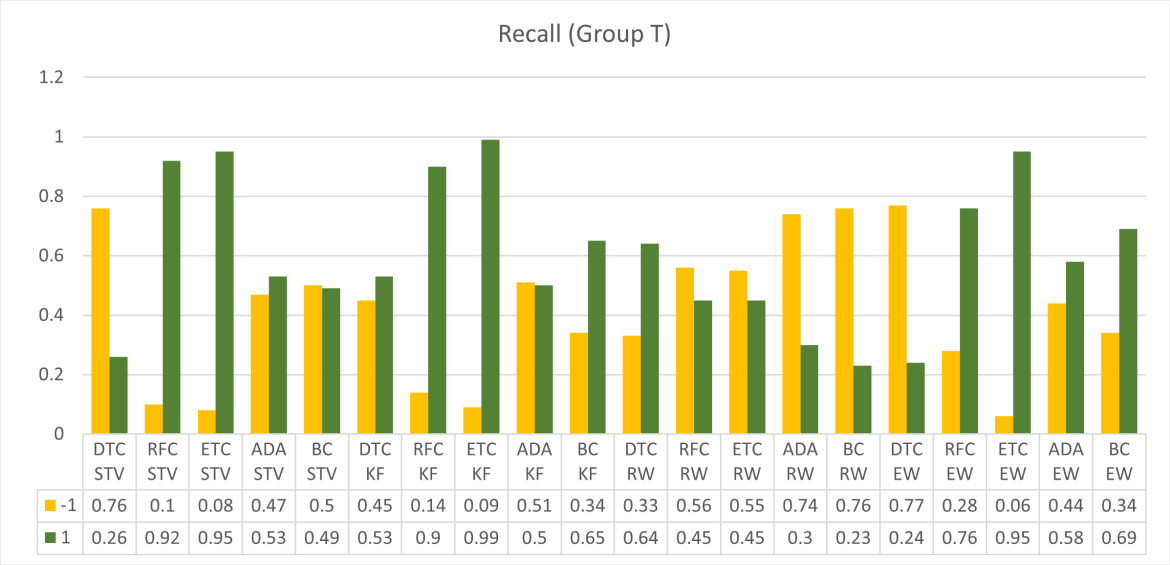
Figure 5.16: Precision Score (Group T)



	DTC STV	RFC STV	ETC STV	ADA STV	BC STV	DTC KF	RFC KF	ETC KF	ADA KF	BC KF	DTC RW	RFC RW	ETC RW	ADA RW	BC RW	DTC EW	RFC EW	ETC EW	ADA EW	BC EW
-1	0.63	0.47	0.66	0.47	0.43	0.52	0.67	0.56	0.46	0.46	0.44	0.43	0.44	0.46	0.43	0.45	0.54	0.5	0.44	0.45
1	0.56	0.57	0.56	0.6	0.55	0.57	0.57	0.56	0.59	0.56	0.55	0.55	0.55	0.59	0.54	0.57	0.57	0.56	0.55	0.56

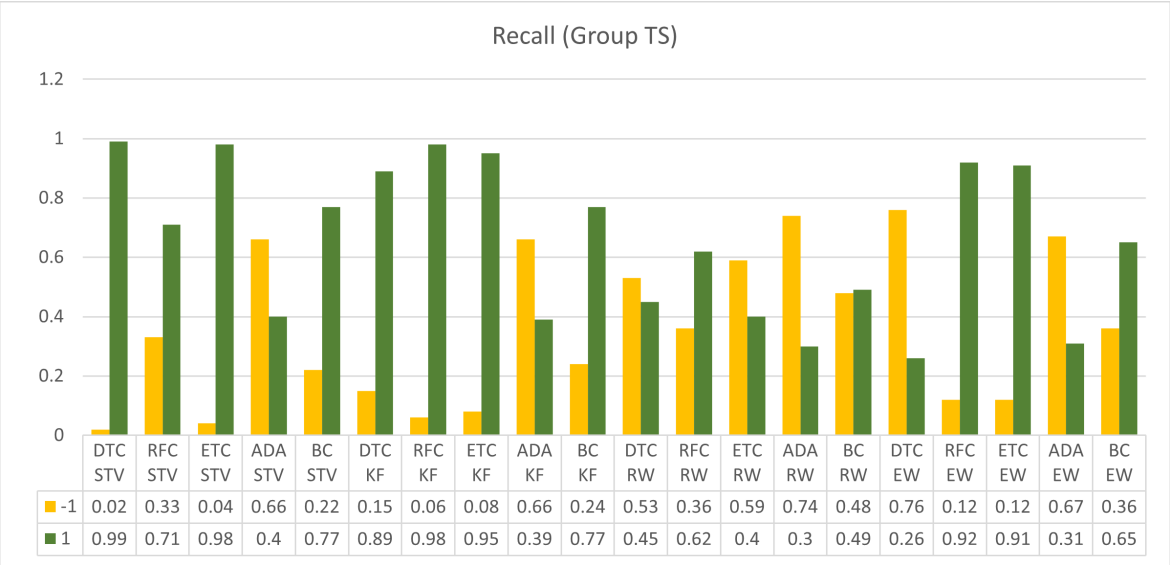
Figure 5.17: Precision Score (Group TS)

A precision score of greater than 0.5 is more desirable. Most models achieve a precision score revolving around 0.42 to 0.59, except ETC_KF_CW with an exceptionally high precision score of 0.76 in the minority class. However, this is not the case for the minority class. There is only 25% of the models in Group T, and 35% of models in Group TS achieve a precision of over 0.5 in detecting minority classes. It is interesting to observe the extremely high precision score achieved by ETC_KF_CV in Group T. This might indicate that it is more capable of handling class imbalance as the high precision is not observed across other validation schemes, but this can only be confirmed after examining its recall score.



	DTC STV	RFC STV	ETC STV	ADA STV	BC STV	DTC KF	RFC KF	ETC KF	ADA KF	BC KF	DTC RW	RFC RW	ETC RW	ADA RW	BC RW	DTC EW	RFC EW	ETC EW	ADA EW	BC EW
-1	0.76	0.1	0.08	0.47	0.5	0.45	0.14	0.09	0.51	0.34	0.33	0.56	0.55	0.74	0.76	0.77	0.28	0.06	0.44	0.34
1	0.26	0.92	0.95	0.53	0.49	0.53	0.9	0.99	0.5	0.65	0.64	0.45	0.45	0.3	0.23	0.24	0.76	0.95	0.58	0.69

Figure 5.18: Recall Score (Group T)



	DTC STV	RFC STV	ETC STV	ADA STV	BC STV	DTC KF	RFC KF	ETC KF	ADA KF	BC KF	DTC RW	RFC RW	ETC RW	ADA RW	BC RW	DTC EW	RFC EW	ETC EW	ADA EW	BC EW
-1	0.02	0.33	0.04	0.66	0.22	0.15	0.06	0.08	0.66	0.24	0.53	0.36	0.59	0.74	0.48	0.76	0.12	0.12	0.67	0.36
1	0.99	0.71	0.98	0.4	0.77	0.89	0.98	0.95	0.39	0.77	0.45	0.62	0.4	0.3	0.49	0.26	0.92	0.91	0.31	0.65

Figure 5.19: Recall Score (Group TS)

Subsequently, the recall score is examined. In the practical sense, recall is relatively important in the stock price prediction as every false signal would result in an investment loss. The recall score of the majority class is higher than the minority class in general, revealing a serious imbalance class issue where models are heavily biased towards the majority class with a few exceptions. DTC, RFC, and ETC demonstrates a lower recall score compared to ADA and BC, in line with the findings of Provost (2008) claiming that tree-based classifier is highly incapable of learning with class imbalance due to the reason that they usually predict the class most prevalent at the matching leaf and this tends to violate the assumption where maximizing accuracy is the end-goal, and the class distribution of training and test sets are the same. In the meantime, ETC_KF_CV, which achieves an exceptionally high precision score in the minority class obtain a relatively low recall score. Its recall on minority class is recorded as 0.09, indicating that it correctly identifies only 9% of the negative signals. The extremely low recall score might be the result of its nature of selecting split points completely at random, resulting in a high possibility of it keeps splitting repeatedly on a subset of the data without separating out the minority class.

At this stage of understanding, we believe that the accuracy score is not a preferred metric to measure the model performance as it overlooks the class imbalance issue and can be misleading when the dataset is not normally distributed. In contrast, f1-score reflects the ability of the model to correctly classify the majority and minority classes,

which is more appropriate for this study in the stock price prediction as an incorrect signal might result in a huge investment loss.

5.2.3 Strategy Performance

RFC consistently achieves a positive Sharpe Ratio in all validation schemes in Group T. It also shows an outstanding performance in RW_CV when all other models achieve a negative Sharpe ratio. Despite the worse performance in Group TS, such that RFC_KF_CV and RFC_RW_CV observe a negative Sharpe ratio, this can be negated by proper selection of input features and validation schemes.

In the meantime, DTC yields the worst performance due to its instability, where minor changes in the input features can result in a drastically different tree, therefore yielding a completely different result. This can be proven by the drastically different value achieved by DTC in Group TS, especially in the simple validation scheme like STV, in line with the findings of Li & Belford (2002) and Dwyer & Holte (2007). It is also worth highlighting that boosting and bagging do not consistently yield better performance than the basic tree-based model and their performances are highly dependent on the input features and validation schemes. For instance, ADA_STV performs better than DTC_STV in Group T, but completely different results are produced in Group TS.

RQ 2: How effective is the S&P 500 stock price prediction using the proposed tree-based models?

An examination into the tree-based models shows that DTC, RFC, and ETC are highly sensitive to class imbalance in comparison to ADA and BC, however, the issue can be resolved by training them with RW_CV to enhance their abilities to capture the minority class accurately. When it comes to strategy performance, RFC appears to be the most robust model in real-life trading where it consistently achieves a positive Sharpe ratio across all validation schemes, despite of its high computational cost, which can be addressed by validating it with computational efficient validation schemes such as RW_CV and EW_CV.

5.3 Evaluation of The Input Feature and Predictive Power of VIX

RQ 3: Does feature affects the performance of the proposed validation schemes and models?

Comparing the model performance of Group T and TS demonstrate that feature does affect the performance of validation schemes and models, and the issue is particularly obvious in the cases of class imbalance. Examining the recall scores of the tree-based models reveals that input features can critically affect the model performance in the cases of class imbalance by worsening the recall scores of the minority class if the input feature is deemed redundant by the models. Besides, the situation can worsen when the model is trained with a highly unstable algorithm such as DTC, as proven by the sharp decline of the recall score of DTC_STV from 0.73 to 0.02 after VIX is included as an input feature.

Tiwari (2014) and Forman (2003) point out that class imbalance is always accompanied by the issue of high dimensionality so it is no doubt that the input features, particularly those redundant, can negatively affect the model performance. Besides, it can also be observed that the models in KF_CV show a drastic decline in Sharpe ratio when VIX is included as the input feature, further validating the assumption that the input feature is one of the deterministic factors in affecting the performance of the validation schemes. However, the extent to which the input features can affect the class imbalance is unknown.

RQ 4: Does VIX consists of the predictive power in S&P 500 stock price prediction?

Despite the statistically significant causal relationship between S&P 500 and VIX, the evaluation of model performance using accuracy and f1-score demonstrates inconsistent results in Group TS, where VIX is included as input feature, doubting the predictive power of VIX on the S&P 500 stock price prediction. Furthermore, it is impossible to capture a specific patterns to explain the predictive power of VIX across models and validation schemes.

Further examination into the strategy performance shows that VIX only improves the Sharpe ratio of 25% of the models, including DTC_STV, RFC_STV, BC_RW_CV, RFC_EW_CV, and BC_EW_CV while 75% of the models demonstrate deterioration in the Sharpe ratio. This indicates that VIX has no capability of enhancing the profitability of the portfolio consistently.

We consider several reasons which may explain this situation. VIX is designed to reflect investors' view of future US stock market volatility – in other words, how much investors think the S&P 500 Index will fluctuate in the next 30 days. However, it is just a measure of where the market stands on any trading day. Wells Fargo analyst Scott Wren wrote ' When the S&P 500 is trading meaningfully to the downside on any given day or series of days, the VIX typically jumps higher in response', assuming that VIX is merely a reflection of the S&P 500 price movement. This explains the strong correlation between S&P 500 and VIX but the little or even non-existence of the predictive power of VIX.

Chapter 6

Conclusion

In this study, we have explored the performances of four validation schemes on the time-series prediction that are evaluated on the tree-based models. Besides, we examine the impact of the feature on the model performance by incorporating VIX as the input feature. This allows us to also examine the predictive power of VIX. We have achieves these objectives by empirically studying the stock price movement of S&P 500 and placed our focus on answering four research questions. The first question - not extensively studied in the literature, was the effectiveness of proposed validation schemes in the time-series stock price prediction. The second question consisted in examining the effectiveness of the proposed tree-based models in predicting the S&P 500 stock price movement. The third and four questions are in regards to the input features - whether input features affect the performance of validation schemes and tree-based models, and the predictive power of VIX on the S&P 500 stock price movement. To address these questions, we developed and validated a number of tree-based models using different model selection procedures, where each model was trained using different validation schemes, and input features.

Regarding the first question, considering the many aspects of how the model performs, we determine EW_CV to be more suitable in the application of stock price prediction as it achieves a satisfying Sharpe ratio in most models while being less computationally expensive. Its trade-off of being unable to handle class imbalance can be negated by performing certain kinds of class balancing such as class weights, sampling, or a specialized loss function. We believe that its potential can be further elaborated when trained with more robust, and stable models such as RFC which yields a desirable Sharpe ratio in most cases as EW_CV can effectively shorten the computational

time of RFC as well. In the meantime, while STV yields a positive Sharpe ratio in most models, it is less recommended due to the extremely high computational cost.

More importantly, for our second question, we find that ADA and BC are less sensitive to class imbalance. This result goes in hand with existing studies which find evidence suggesting that boosting and bagging can effectively address the issue of class imbalance. When it comes to strategy performance, RFC outperforms other tree-based models by achieving a positive Sharpe ratio in all validation schemes.

Furthermore, we find that input features to be one of the deterministic factors in affecting the performance of validation schemes and tree-based models. Despite VIX does not contain predictive power on the time-series S&P 500 stock price prediction, empirical findings show that adding the right input features is extremely crucial to improve the performance of state-of-the-art classification models.

Overall, having trained stock price prediction models using traditional and time-series validation schemes on the tree-based models, and analyzing the impact of input features on the performance of the validation schemes and models, our results further validate the importance of evaluating the time-series data appropriately to achieve robust prediction result. Besides, we prove that several models are capable in generating profitability in real-life trading when trained using specific validation schemes, models and input features, indicating that stock price movement does not only follow the random walk patterns but there are other deterministic factors affecting it.

Future investigations are necessary to validate the kinds of conclusions that can be drawn from this study. We believe that it is necessary to continue exploring the robustness of time-series validation schemes including EW_CV and RW_CV by investigating their association with the different models and feature selection to enhance their profitability in real-life trading. Besides, it is worth examining into the combining these validation schemes with more complex models, such as deep neural network to tackle the issue of computational cost, while boosting their robustness.

References

Agrawal, M., Khan, A. & Shukla, P. (2019), 'Stock indices price prediction based on technical indicators using deep learning model', *International Journal on Emerging Technologies* **10**.

Akour, M., Alsmadi, I. & Alazzam, I. (2017), 'Software fault proneness prediction: A comparative study between bagging, boosting, and stacking ensemble and base learner methods', *International Journal of Data Analysis Techniques and Strategies* **9**, 1–16.

Ampomah, E., Qin, Z. & Nyame, G. (2020), 'Evaluation of tree-based ensemble machine learning models in predicting stock price direction of movement', *Information* **11**, 332.

Ampomah, E., Qin, Z., Nyame, G. & Botchey, F. (2021), 'Stock market decision support modeling with tree-based adaboost ensemble machine learning models', *Informatica* **44**.

Arlot, S. & Celisse, A. (2009), 'A survey of cross validation procedures for model selection', *Statistics Surveys* **4**.

Asiamah, N., Mensah, H. K. & Oteng-Abayie, E. F. (2017), 'Do larger samples really lead to more precise estimates? a simulation study', *American Journal of Educational Research* **5**, 9–17.

Attigeri, G. V., M, M. P. M., Pai, R. M. & Nayak, A. (2015), Stock market prediction: A big data approach, *in* 'TENCON 2015 - 2015 IEEE Region 10 Conference', pp. 1–5.

Aznar, P. (2020), 'What is the difference between extra trees and random forest?', https://quantdare.com/what-is-the-difference-between-extra-trees-and-random-forest/. [Accessed on 8 February 2022].

Bai, M., Liu, X., Yang, K. & Li, Y. (2019), Stock investment strategy based on decision

tree, *in* '2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)', pp. 151–155.

Basak, S., Kar, S., Saha, S., Khaidem, L. & Dey, S. R. (2019), 'Predicting the direction of stock market prices using tree-based classifiers', *The North American Journal of Economics and Finance* .

Bauer, E. & Kohavi, R. (1996), 'An empirical comparison of voting classification algorithms : Bagging, boosting, and variants', *Machine Learning* **36**, 1–38.

Bergmeir, C. & Benítez, J. M. (2012), 'On the use of cross-validation for time series predictor evaluation', *Inf. Sci.* **191**, 192–213.

Bergmeir, C., Hyndman, R. J. & Koo, B. (2015), A note on the validity of cross-validation for evaluating time series prediction.

Bi, Q., Yan, H., Chen, C. & Su, Q. (2020), An integrated machine learning framework for stock price prediction, *in* 'Information Retrieval: 26th China Conference, CCIR 2020, Xi'an, China, August 14–16, 2020, Proceedings', Springer-Verlag, p. 99–110.

Breiman, L. (2004), 'Bagging predictors', *Machine Learning* **24**, 123–140.

Brownstone, D. (1996), 'Using percentage accuracy to measure neural network predictions in stock market movements', *Neurocomputing* **10**, 237–250.

Bühlmann, P. (2012), 'Bagging, boosting and ensemble methods', *Handbook of Computational Statistics* .

Caruana, R. & Niculescu-Mizil, A. (2006), 'An empirical comparison of supervised learning algorithms', *Proceedings of the 23rd international conference on Machine learning* .

Chen, Y., Yang, B. & Abraham, A. (2007), 'Flexible neural trees ensemble for stock index modeling', *Neurocomputing* **70**, 697–703.

Chen, Z., Liang, C. & Umar, M. (2021), 'Is investor sentiment stronger than vix and uncertainty indices in predicting energy volatility?', *Resources Policy* **74**, 102391.

Chou, J.-S. & Nguyen, T.-K. (2018), 'Forward forecast of stock price using sliding-window metaheuristic-optimized machine-learning regression', *IEEE Transactions on Industrial Informatics* **14**(7), 3132–3142.

Clay, H. (2022), 'Wisdom of crowds', https://www.investopedia.com/terms/w/wisdom-crowds.asp: :text=Wisdom%20of%20the%20crowd%20is%20a%20theory%20that,at%20decisio making%2C%20problem-solving%2C%20and%20innovating%20than%20an%20individual. [Accessed on 8 February 2022].

CNN (2022), 'Fear & greed index', https://edition.cnn.com/markets/fear-and-greed. [Accessed on 8 February 2022].

Cutler, A., Cutler, D. & Stevens, J. (2011), *Random Forests*, Vol. 45, pp. 157–176.

Di Persio, L. & Honchar, O. (2016), 'Artificial neural networks architectures for stock price prediction: Comparisons and applications', **10**, 403–413.

Doan, T. & Lo, A. (1988), 'Stock market prices do not follow random walks: Evidence from a simple specification test', *Review of Financial Studies* **1**, 41–66.

Donaldson, R. & Kamstra, M. J. (1996), 'Forecast combining with neural networks', *Journal of Forecasting* **15**, 49–61.

Downey, L. (2022), 'Efficient market hypothesis (emh)', https://www.investopedia.com/terms/e/efficientmarkethypothesis.asp. [Accessed on 8 February 2022].

Drucker, H. (1997), 'Improving regressors using boosting techniques', *Proceedings of the 14th International Conference on Machine Learning* .

Dwyer, K. & Holte, R. (2007), Decision tree instability and active learning, Vol. 4701, pp. 128–139.

Fama, E. F. (1965), 'The behavior of stock-market prices', *The Journal of Business* **38**(1), 34–105.
URL: *http://www.jstor.org/stable/2350752*

Forman, G. (2003), 'An extensive empirical study of feature selection metrics for text classification [j]', *Journal of Machine Learning Research - JMLR* **3**.

Freund, Y. & Schapire, R. (2001), 'Discussion of the paper 'arcing classifiers' by leo breiman', **26**.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H. & Herrera, F. (2012), 'A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches', *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* **42**(4), 463–484.

Gallagher, L. & Taylor, M. (2002), 'Permanent and temporary components of stock prices: Evidence from assessing macroeconomic shocks', *Southern Economic Journal* **69**, 345–362.

Giaglis, G., Georgoula, I., Pournarakis, D., Bilanakos, C. & Sotiropoulos, D. (2015), Using time-series and sentiment analysis to detect the determinants of bitcoin prices.

Giot, P. (2003), 'Implied volatility indexes and daily value at risk models', *Journal of Derivatives - J DERIV* **12**.

Hastie, T., Tibshirani, R., Friedman, J. & Franklin, J. (2004), 'The elements of statistical learning: Data mining, inference, and prediction', *Math. Intell.* **27**, 83–85.

Hsu, M.-W., Lessmann, S., Sung, M.-C., Ma, T. & Johnson, J. (2016), 'Bridging the divide in financial market forecasting: Machine learners vs. financial economists', *Expert Systems with Applications* **61**.

Huang, J.-Z., Huang, W. & Ni, J. (2019), 'Predicting bitcoin returns using high-dimensional technical indicators', *The Journal of Finance and Data Science* .

Inamdar, A., Bhagtani, A., Bhatt, S. & Shetty, P. (2019), Predicting cryptocurrency value using sentiment analysis, pp. 932–934.

Iqbal, W., Berral, J., Carrera, D. & Baig, S.-u.-R. (2019), 'Adaptive sliding windows for improved estimation of data center resource utilization', *Future Generation Computer Systems* **104**.

Jensen, M. C. (1978), 'Some anomalous evidence regarding market efficiency', *Capital Markets: Market Efficiency* .

Jiang, G. & Wang, W. (2017), 'Markov cross-validation for time series model evaluations', *Information Sciences* **375**.

Jiao, Y. & Jakubowicz, J. (2017), Predicting stock movement direction with machine learning: An extensive study on s&p 500 stocks, *in* '2017 IEEE International Conference on Big Data (Big Data)', pp. 4705–4713.

Kaggle (2021), 'Adaboost vs xgboost which is best and why ?', https://www.kaggle.com/general/218408. [Accessed on 8 February 2022].

Kamble, R. A. (2017), Short and long term stock trend prediction using decision tree, *in* '2017 International Conference on Intelligent Computing and Control Systems (ICICCS)', pp. 1371–1375.

Karim, R., Alam, M. & Hossain, M. (2021), Stock market analysis using linear regression and decision tree regression, pp. 1–6.

Kenton, W. (2022), 'The s&p 500 index: Standard & poor's 500 index', https://www.investopedia.com/terms/s/sp500.asp. [Accessed on 8 February 2022].

Khan, M. T., Durrani, M. Y., Ali, A., Inayat, I., Khalid, S. & Khan, K. H. (2016), 'Sentiment analysis and the complex natural language', *Complex Adaptive Systems Modeling* **4**, 1–19.

Khoshgoftaar, T. M., Van Hulse, J. & Napolitano, A. (2011), 'Comparing boosting and bagging techniques with noisy and imbalanced data', *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* **41**(3), 552–568.

Kovaleski, D. (2022), 'Amazon vs. the s&p 500: Which is the better first investment?', https://www.msn.com/en-us/money/topstocks/amazon-vs-the-s-26p-500-which-is-the-better-first-investment/ar-AAZ27ft. [Accessed on 8 February 2022].

Krauss, C., Do, X. & Huck, N. (2016), 'Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the s&p 500', *European Journal of Operational Research* **259**.

Kumar, D., Meghwani, S. & Thakur, M. (2016), 'Proximal support vector machine based hybrid prediction models for trend forecasting in financial markets', *Journal of Computational Science* **17**.

Kumar, M. & Thenmozhi, M. (2006), 'Forecasting stock index movement: A comparison of support vector machines and random forest', *SPGMI: Compustat Fundamentals (Topic)* .

Lee, M.-C. (2009), 'Using support vector machine with a hybrid feature selection method to the stock trend prediction', *Expert Systems with Applications* **36**, 10896–10904.

Li, R.-h. & Belford, G. (2002), 'Instability of decision tree classification algorithms'.

Liu, G., Mao, Y., Sun, Q., Huang, H., Gao, W., Li, X., Shen, J., Li, R. & Wang, X. (2020), Multi-scale two-way deep neural network for stock trend prediction, *in* 'IJCAI'.

Lohrmann, C. & Luukka, P. (2018), 'Classification of intraday s&p500 returns with a random forest', *International Journal of Forecasting* **35**.

Malkiel, B. (2003), 'The efficient market hypothesis and its critics', *Journal of Economic Perspectives* **17**, 59–82.

Mims, C. (2021), 'Gamestop, bitcoin and qanon: How the wisdom of crowds became the anarchy of the mob', https://www.wsj.com/articles/gamestop-bitcoin-and-qanon-how-the-wisdom-of-crowds-became-the-anarchy-of-the-mob-11611928823. [Accessed on 8 February 2022].

Mittal, A. (2011), Stock prediction using twitter sentiment analysis.

Neely, C. J., Rapach, D. E., Tu, J. & Zhou, G. (2014), 'Forecasting the equity risk premium: The role of technical indicators', *Management Science* **60**(7), 1772–1791.

Nti, I. k., Adekoya, A. & Weyori, B. (2020), 'Efficient stock-market prediction using ensemble support vector machine', **10**, 154–163.

Opsomer, J., Wang, Y. & Yang, Y. (2000), 'Nonparametric regression with correlated errors', *Statistical Science* **16**.

Palmer, C. (2021), 'How social listening and machine learning are used to predict bitcoin price volatility', https://bitcoinmagazine.com/markets/social-sentiment-and-the-bitcoin-price. [Accessed on 8 February 2022].

Patel, J., Shah, S., Thakkar, P. & Kotecha, K. (2015), 'Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques', *Expert Systems with Applications* **42**, 259–268.

Ponciano, J. (2022), 'Stock market could crash another 20% if u.s. plunges into recession—these industries are most at risk', https://www.forbes.com/sites/jonathanponciano/2022/06/21/stock-market-could-crash-another-20-if-us-plunges-into-recession-these-industries-are-most-at-risk/?sh=511b72686197. [Accessed on 8 February 2022].

Professor, D. & Wiggins, R. (2001), 'S&p futures returns and contrary sentiment indicators', *Journal of Futures Markets* **21**, 447 – 462.

Provost, F. J. (2008), Machine learning from imbalanced data sets 101.

Qian, B. & Rasheed, K. M. (2006), 'Stock market prediction with multiple classifiers', *Applied Intelligence* **26**, 25–33.

Qiang, Z. & Shen, J. (2021), Bitcoin high-frequency trend prediction with convolutional and recurrent neural networks.

Refenes, A.-P. N., Zapranis, A. & Francis, G. (1994), 'Stock performance modeling using neural networks: A comparative study with regression models', *Neural Networks* **7**, 375–388.

Rekha, G., Tyagi, A. & Reddy, V. (2019), 'Solving class imbalance problem using bagging, boosting techniques, with and without using noise filtering method', *International Journal of Hybrid Intelligent Systems* **15**, 1–10.

Ren, R., Wu, D. D. & Liu, T. (2019), 'Forecasting stock market movement direction using sentiment analysis and support vector machine', *IEEE Systems Journal* **13**(1), 760–770.

Ronchetti, E. (2000), 'Regression and time series model selection by allan d. r. mcquarrie; chih-ling tsai', *Journal of the American Statistical Association* **95**, 1008–1009.

Roy, S., Chopra, R., Lee, K., Spampinato, C. & ivatlood, B. (2020), 'Random forest, gradient boosted machines and deep neural network for stock price forecasting: a comparative analysis on south korean companies', *International Journal of Ad Hoc and Ubiquitous Computing* **33**, 62.

Sadorsky, P. (2021), A random forests approach to predicting clean energy stock prices.

Sadorsky, P. (2022), 'Forecasting solar stock prices using tree-based machine learning classification: How important are silver prices?', *The North American Journal of Economics and Finance* **61**, 101705.

scikit learn.org (2022), 'sklearn.ensemble.baggingclassifier', https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingClassifier.html. [Accessed on 8 February 2022].

Shynkevich, Y., Mcginnity, T., Coleman, S., Belatreche, A. & Li, Y. (2017), 'Forecasting price movements using technical indicators: Investigating the impact of varying input window length', *Neurocomputing* .

Smales, L. (2013), 'News sentiment and the investor fear gauge', *Finance Research Letters* **11**.

Smales, L. (2017), 'The importance of fear: investor sentiment and stock market returns', *Applied Economics* **49**, 1–27.

Smales, L. A. (2014), 'News sentiment and the investor fear gauge', *Finance Research Letters* **11**, 122–130.

Song, Y. & Lee, J. (2020), 'Importance of event binary features in stock price prediction', *Applied Sciences* .

Special, E. S. (2021), 'Extent of elon musk's influence on cryptocurrency; where is it headed?', https://economictimes.indiatimes.com/markets/cryptocurrency/extent-of-elon-musks-influence-on-cryptocurrency-where-is-it-headed/articleshow/83037268.cms. [Accessed on 8 February 2022].

Tashman, L. (2000), 'Out-of-sample tests of forecasting accuracy: An analysis and review', *International Journal of Forecasting* **16**, 437–450.

Tiwari, D. (2014), Handling class imbalance problem using feature selection.

Tsai, C.-F., Lin, Y.-C., Yen, D. & Chen, Y.-M. (2011), 'Predicting stock returns by classifier ensembles', *Appl. Soft Comput.* **11**, 2452–2459.

Wang, H., Jiang, Y. & Wang, H. (2009), Stock return prediction based on bagging-decision tree, *in* '2009 IEEE International Conference on Grey Systems and Intelligent Services (GSIS 2009)', pp. 1575–1580.

Wang, Y.-Y. & Li, J. (2008), 'Feature-selection ability of the decision-tree algorithm and the impact of feature-selection/extraction on decision-tree results based on hyperspectral data', *International Journal of Remote Sensing - INT J REMOTE SENS* **29**, 2993–3010.

Weng, B., Ahmed, M. & Megahed, F. (2017), 'Stock market one-day ahead movement prediction using disparate data sources', *Expert Systems with Applications* **79**.

Weng, B., Lu, L., Wang, X., Megahed, F. & Martinez, W. (2018), 'Predicting short-term stock prices using ensemble methods and online data sources', *Expert Systems with Applications* **112**.

Wu, J. M.-T., Li, Z., Srivastava, G., Frnda, J., García Díaz, V. & Lin, C.-W. (2020), A cnn-based stock price trend prediction with futures and historical price.

Wu, M.-C., Lin, S.-Y. & Lin, C.-H. (2006), 'An effective application of decision tree to stock trading', *Expert Systems with Applications* **31**, 270–274.

Wurgler, J. & Baker, M. (2006), 'Investor sentiment and the cross-section of stock returns', *Journal of Finance* **61**, 1645–1680.

Zhai, Y., Hsu, A. & Halgamuge, S. (2007), Combining news and technical indicators in daily stock price trends prediction, pp. 1087–1096.

Zhang, Y.-D. & Wu, L. (2009), 'Stock market prediction of s&p 500 via combination of improved bco approach and bp neural network', *Expert Systems with Applications* **36**, 8849–8854.