MASTER

Predictive analysis using machine learning techniques in financial markets

Cosan Mutlu, Guliz

Award date:
2018

Awarding institution:
Polytechnic University of Madrid

CAMPUS
DE EXCELENCIA
INTERNACIONAL

Máster Universitario en Ingeniería Informática

Universidad Politécnica de Madrid

Escuela Técnica Superior de
Ingenieros Informáticos

TRABAJO FIN DE MÁSTER

Predictive Analysis Using Machine Learning Techniques in Financial Markets

Autor: Guliz Cosan Mutlu

Director: Dr. Alejandro Rodríguez González

MADRID, JULY 2018

Declaration of Authorship

I, Guliz COSAN MUTLU, declare that this thesis titled, "Predictive Analysis Using Machine Learning Techniques in Financial Markets" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

"...And you know everything as much as you learn, learn this as well"

Can Yucel

UNIVERSIDAD POLITÉCNICA DE MADRID

Abstract

ETS de Ingenieros Informáticos

Master of Science

Predictive Analysis Using Machine Learning Techniques in Financial Markets

by Guliz COSAN MUTLU

Machine learning has proved its power on both scientific and business based studies. Yet, tremendous number of researches are still being conducted on variety of applications. This thesis is one of those applications of machine learning on a business case to reveal potential insights. The goal of this thesis is to apply machine learning techniques on high frequency daily market making performance for an analysis of whether certain financial market indicators affects the mentioned performance. Moreover, this research also includes a preview of a potential future work where the focus is to build models which can forecast future business revenues.

The research has concluded that [redacted].

UNIVERSIDAD POLITÉCNICA DE MADRID

Abstract

ETS de Ingenieros Informáticos

Master of Science

Predictive Analysis Using Machine Learning Techniques in Financial Markets

by Guliz COSAN MUTLU

El aprendizaje automático ha demostrado su efectividad en estudios científicos y en la industria. Sin embargo, todavía se están llevando a cabo una gran cantidad de investigaciones sobre una variedad de aplicaciones. Esta tesis se centra en la aplicación de aprendizaje automático en un caso de negocios, con el propósito de revelar ideas potenciales. El objetivo de esta tesis es aplicar técnicas de aprendizaje automático en el rendimiento diario de mercado de alta frecuencia y analizar si ciertos indicadores del mercado financiero afectan el rendimiento mencionado. Esta investigación también incluye una vista previa de un posible trabajo futuro en el que el objetivo es construir modelos que puedan pronosticar los ingresos comerciales futuros.

Acknowledgements

This thesis was written as a part of Master's in Data Science at EIT Digital Master Program at Eindhoven University of Technology in the first year and at Technical University of Madrid in the second year. This Master's was an amazing journey and this thesis puts an end this journey. The process of writing the thesis was challenging and full of hardworking. First of all, I would like to thank my supervisor Dr. Alejandro Rodriguez Gonzalez and Dr. Massimiliano Zanin and Dr. Ernestina Menasalvas Ruiz, who were also involved to this process, for all their advice and helpful comments.

During these 2 years and whole my life, I'm grateful for the endless support and love of my family and my beloved husband. Especially, to him, Aytek, thank you for being there in any condition.

The time I spent during this master, I had a chance to meet people from all over the world. To the ones with whom I shared many memories and learned so many things and had fun, I would like to thank for making these years unforgettable.

Lastly, I would like to thank to [redacted] for the opportunity and providing me this project. All the 'Good Job's encouraged me always and I had a pleasure working with you during my internship.

Contents

1 Introduction **1**

1.1 Background and Motivation 1

1.2 Goals 2

1.3 Thesis Report Structure 3

2 Literature Review **5**

2.1 State of Art 5

2.2 Discussion 7

3 Data and Methodology **9**

3.1 Data 9

3.2 Methodology 10

3.2.1 Regression 10

Linear Models 10

Linear Regression 11

LASSO Regression 12

Non-Linear Models 13

Multi Layer Perceptron Regressor (MLPR) 13

Support Vector Regressor (SVR) 15

3.3 Model Fitting 16

3.3.1 Overfitting 16

3.3.2 Parameter Tuning 17

3.3.3 Regularization 18

L1 Regularization 18

L2 Regularization 18

3.3.4 Feature Selection . . . 19
3.4 Tools . . . 20

4 Design of Experiment **21**
4.1 Data Preprocessing . . . 21
4.1.1 Data Transformation . . . 21
4.1.2 Data Integration . . . 22
4.1.3 Data Cleaning . . . 23
Null Values . . . 23
Outliers . . . 24
4.1.4 Data Standardization (z-score) . . . 27
4.2 Model Building . . . 28
4.3 Evaluation Measures . . . 31
4.3.1 Mean Squared Error . . . 31
4.3.2 Computation cost . . . 31

5 Experiment and Results **33**
5.1 Feature Selection Results . . . 33
5.2 Hyperparameters . . . 34
5.3 Accuracy Results . . . 35
5.4 Computational Cost . . . 36
5.5 Visualization . . . 37

6 Discussion **39**
6.1 Limitations . . . 42
6.2 Future Work . . . 42

7 Preview: Time-Series Forecasting **45**
7.1 Introduction & Goal . . . 45
7.2 Data . . . 46
7.3 Methodology . . . 46
7.3.1 Stationarity . . . 47
7.3.2 Granger Causality . . . 47

7.3.3 Forecasting 49
ARMA Model 49
ARMA with Exogenous Regressors 50
7.4 Preliminary Results 50
7.5 Discussion 52

8 Conclusion **53**

A Example Raw Financial Market Indicator Data **55**

B Null Value Deletion **57**
B.1 Mean and Standard Deviation of Features Before and After Data Deletion 57
B.2 t-test Results for the Effect of Data Deletion 57

C Learning Curves **59**

D Residual Q-Q Plots **61**

E Test Results for Preview: Time-Series Forecasting **63**
E.1 Stationarity Test Results 63
E.2 Causality Test Results 63
E.3 ACF and PACF Plots 64
E.4 Actual vs. Forecasted Plots 65
E.5 Residual Q-Q Plots 66
E.6 Autocorrelations of Residuals 67

Bibliography **69**

List of Figures

3.1 Example MLPR Architecture (Cakraci, 2017) 14
3.2 Optimal Hyperplane Selection of SVR (Kaiping, 2017) 15

4.1 The Box Plot of the Dependent Variable Before Winsorization or Truncation 25
4.2 The Box Plot of the Dependent Variable After Winsorization . . 26
4.3 The Box Plot of the Dependent Variable After Truncation 27
4.4 Example Cross Validated Procedure (Raschka, 2013) 30

5.1 Visualization of Regression Models on Dashboard 37

6.1 The Learning Curve of Linear Regression with Truncated Outliers 41

List of Tables

4.1 Number of Features and Observations 22
4.2 Summary of Number of Observations 24
4.3 Number of Observations Before and After Outlier Truncation . 27
4.4 Parameter grids for LASSO Regression 29
4.5 Parameter grids for MLPR 29
4.6 Parameter Grids for SVR 29

5.1 Number of Features Before and After Feature Selection for Linear Regression, MLPR and SVR Algorithms 33
5.2 Number of Features Before and After Natural Feature Selection of LASSO 34
5.3 Hyperparameters of LASSO Model with Truncated Outliers and with Winsorized Outliers 34
5.4 Hyperparameters of MLPR Model with Truncated Outliers and with Winsorized Outliers 34
5.5 Hyperparameters of SVR Model with Truncated Outliers and with Winsorized Outliers 35
5.6 Accuracy Results of Regression Algorithms with Truncated Outliers 35
5.7 Accuracy Results of Regression Algorithms with Winsorized Outliers 35
5.8 Accuracy Results of Dummy Model 35
5.9 Computation Times of Regression Algorithms 36

7.1 Number of Observations for each Region Before and After Data Deletion 46
7.2 Prepared Datasets for Time-Series Analyses 47
7.3 List of Granger Causality Tests Applied 48
7.4 Parameters of ARMA Models Built 50
7.5 Built ARMA Models 51
7.6 Preliminary Test Results of Built ARMA Models 51

List of Abbreviations

ACF	**A**uto**C**orrelation **F**unction
ADF	**A**ugmented **D**ickey-**F**uller
AIC	**A**kaike **I**nformation **C**riteria
API	**A**pplication **P**rogramming **I**nterface
ARMA	**A**uto**R**egressive **M**oving **A**verage
BIC	**B**ayesian **I**nformation **C**riteria
LASSO	**L**east **A**bsolute **S**hrinkage and **S**election **O**perator
MAPE	**M**ean **A**bsolute **P**ercentage **E**rror
MLPR	**M**ulti **L**ayer **P**erceptron **R**egressor
MSE	**M**ean **S**quared **E**rror
PACF	**P**artial **A**uto**C**orrelation **F**unction
SGD	**S**tochastic **G**radient **D**escent
SVM	**S**upport **V**ector **M**achine
SVR	**S**upport **V**ector **R**egressor

Chapter 1

Introduction

1.1 Background and Motivation

Machine learning can be perceived as using computers not only for calculation and data processing but also performing calculations on the processed data which could share an analogy with the notion of learning. Moreover, the aim is computers making reasonable decisions based on observed instances and previous observations without being adhered to predetermined rules and plans.

As the rational decision-making is being transferred to computers from human beings, machine learning offers a great opportunity for businesses so that the overall work flow can be freed from subjectivity. Therefore, it is critical for businesses to develop an understanding intuition about how these machine learning techniques can be beneficial for their decision-making process. According to an analysis report published by McKinsey Global Institute, many major industry groups are relevant within the scope of machine learning and they carry a huge potential. Needless to mention, finance is one of those areas (Henke et al., 2016).

Today, machine learning has come to play an integral role in many phases

of the financial ecosystem, from approving loans, to managing assets and to assessing risks. Moreover, current revenue reporting and forecasting approaches will be obsolete soon because technology innovation drives accelerating waves of business disruption, it's more challenging than ever for companies to analyze and forecast revenue and profitability (Alexander et al., 2016). Therefore, there are more uses cases of machine learning in finance than ever before. Given high volume, accurate historical records, and quantitative nature of the finance world, few industries are better suited for machine learning applications. This suitability offers many opportunities to finance industry. As the industry itself is quite dynamic due to its nature, it is at utmost importance to reveal hidden insights from financial data in the most accurate and fastest way. However, even finding and learning the basics which are suitable for the particular business case can be a challenging task.

This research is motivated by the challenge itself. The concern is to understand suitability of application of machine learning a business case with a focus on analysis of company performance. Therefore, one of the motivations is to find suitable machine learning methods to make the most accurate predictive analysis. The other motivation of this study is to reveal material insights with which business can judge their performance under the light of analysis results.

1.2 Goals

As much deeper insights become possible with the developments in machine learning technologies and big data, their application in the areas where there is a certain need for an improve becomes an important research topic.

It is believed that machine learning techniques have the power to renew not only the answers to business questions but also the questions that should be asked. Therefore, the effect of machine learning to industries is expected to be revolutionary.

The central focus of this research is to apply different machine learning techniques on a real business case and to discuss their applications within the scope of the particular data. Specifically, the focus is to contribute the decision-making process of the business itself.

Besides, one goal of this study is to identify certain financial market indicators that affects high frequency trading market making performance with regression based machine learning techniques so that consequences of certain market movements can be identified. The performance here is defined as [redacted]

Another goal of this study is to show a preview of times series forecasting methods within scope of machine learning to forecast future high frequency trading market making performance in order to offer a guidance for future research possibilities.

1.3 Thesis Report Structure

This report includes the following chapters:

Chapter 2. Literature Review This chapter depicts the state of art related to the study.

Chapter 3. Data and Methodology The data that is utilized for the study is described in detail in this chapter. Moreover; the scientific approach, tools and the specific methods for the experiments are explained.

Chapter 4. Design of Experiment Design of the models generated for experimenting the study is provided in Chapter 4.

Chapter 5. Experiment Results This chapter includes the performance results obtained from experiments of the models.

Chapter 6. Discussion Experiment results are discussed and analyzed in this chapter. Moreover, all of the issues during experiments and drawbacks in the model are mentioned.

Chapter 7. Preview: Time-Series Forecasting This chapter exhibits a preview of another study conducted on the same business application with another approach.

Chapter 8. Conclusion This chapter summarizes and finalizes the work done. Possible improvements and potential future studies are discussed.

Chapter 2

Literature Review

2.1 State of Art

This research is based on machine learning methodologies and their application on a real business case. The scope within the research covers especially [redacted]. Therefore, it is important to understand the current related studies and improvements on machine learning applications on business, with the focus of finance. This understanding would route the focus of the study through research topics where there is potential for improvement. For this purpose, a thorough literature research is conducted.

Machine learning has already infiltrated to the core of many business applications. It is being used to evaluate and investigate each and every process within companies to achieve the most efficient working cycle possible. A study of Accenture reveals that 38% of early business adopters have benefited from machine learning with at least a factor of 2 (Wilson, Mulani, and Alter, 2017).

On the finance focus, the most popular application of machine learning on businesses is observed on trading activities. Conventionally, financial market traders were executing deals first via phone conversations and then with

electronic chats and trading platforms manually. All of those executions were based on human decisions and actions. However, machine learning brought the question of whether the deals can be traded in a more efficient and a structured manner. This question itself has completely changed the financial trading industry and risen the term "high frequency trading" (Conerly, 2014). High frequency trading is a method of executing orders with pre-defined trading instructions based on the computerized decisions taken in accordance with market movements without any human interaction (Gordon, 2017). The motivation of high frequency trading is to reduce the trading costs and to consume more data during the decision process itself (Biais, 2011). Because, machine learning enables the systems to take trading decisions based on all of the available data in the market.

These developments on the trading industry has also computerized the working principle of market makers. A market maker is a market participant who constantly posts buy and sell orders to the market (Aldridge, 2011). Where this process, similar to all trading activities, was completely manual, today most of the market makers utilize machine learning techniques to predict the market movements and post orders accordingly (Kanagal, Wu, and Chen, 2017). The prediction is basically conducted by utilizing data from financial markets for a learning analysis so that market making performance could be analyzed and improved.

Similar to high frequency market making, there has been extensive studies to analyze business performance by utilizing unique metrics and factors of each different industry or business area for different purposes. Following, exemplary applications of machine learning on business performance in variety of sectors will be discussed.

Sunthornjittanon, 2015 studies business performance, i.e. net income, of an agrochemical company to find out whether it is possible to explain it with financial indicators of the company through a linear regression. All of the indicators are populated from internal financial ratios of the company.

Furthermore, another study (Mohamad, Ibrahim, and Massoud, 2013) approaches the same goal of analyzing business performance with the focus of construction companies. The research utilizes both regression and neural networks on internal and external financial indicators to assess net profit of construction companies.

Gajewar and Bansa, 2016 aims to forecast revenue of enterprise products with time-series forecasting and regression models. The study is structured to gain insights about quarterly revenue of company with the help of historical quarterly values and external macroeconomic market conditions.

Another popular approach is to predict stock market movements in order to boost revenue. An exemplary study is conducted by Shen, Jiang, and Zhang, 2012 where the outcome of the study suggests potential trend of the next trading day in the market. One other study (Avdalović and Milenković, 2017) investigates the relationship between stock prices and company performance.

2.2 Discussion

It is an undeniable fact that machine learning is becoming an extremely valuable asset for businesses day-by-day. It is believed that machine learning will even outsmart humans by 2050 (Yunus, 2018). State of art also proves that the transformation process is ongoing rapidly and therefore it is essential for businesses to stay up-to-date with the latest implementations of machine learning.

It should be noted that even if each of the current applications are built with similar machine learning techniques, designing the optimal structure for the specific goal is a challenging process which requires a significant amount of effort. Therefore, what has been already done could be used merely as a guide and the design and implementation of a new business application is a brand new process as far as machine learning models are concerned.

What Chapter 2.1 claims that machine learning has been there in many industries for business performance analysis and also in finance industry. Yet, this study is not only focusing on offering an application to increase performance within a business or to apply machine learning on a finance focus. This study is conducted to combine both and apply generic predictive analytics methods on business performance with a specific focus on financial data to reveal underlying business factors. To clarify, the study will offer an innovative application of machine learning on high frequency market making [redacted] Even tough literature scan resulted in inspirations, it is believed that this study would suggest an innovative approach.

Chapter 3

Data and Methodology

3.1 Data

Financial market indicators mentioned in Chapter 1.2 are global and regional indicators such as stock exchange indexes, volatility indexes etc. In this study, historical end of day values for determined indicators are fetched from public financial data source [redacted] with following columns: [redacted]. An exemplary data fetched is depicted in Appendix A.

Examples of financial indicators for which data is fetched from [redacted] are [redacted]

[redacted]

[redacted]

etc.

As high frequency trading is a rapidly changing industry, the trading strategies and the underlying financial products traded change quite often. For this reason, above-mentioned historical financial market indicators data are obtained daily starting from the first trading day of 2016.

Market making daily trading performance data mentioned in Chapter 1.2 is obtained confidentially and utilized for exemplary experiments in this study. However, any different performance indicator data could be utilized with the methodology described in Chapter 3.2.

Financial market indicators dataset and market making daily trading performance dataset are retrieved both in raw format and then pass through various data preprocessing steps (Chapter 4.1) before being utilized in the experiments.

3.2 Methodology

3.2.1 Regression

The main goal of any data analysis is to extract valuable and accurate estimation from raw information. Concerning if there is any statistical relation between a response variable and explanatory variables, regression analysis is the answer in order to model its relationship. Therefore, regression analysis is used to look for financial market indicators that affects high frequency trading daily market making performance. Every regression analysis consists of dependent and independent variables where the dependent variable is the response variable and the independent variables are the explanatory variables. The method is to vary the values of the independent variable(s) in a formal way to observe the changes in the dependent variable.

The dependent variable within the regression analysis is the historical daily market making performance and the independent variables are certain financial market indicators explained in Chapter 3.1.

Linear Models

Linear Regression Single linear regression is the simplest regression model where the aim is to look for a relationship between a target value and a single feature.

$$\text{Y} = \text{intercept} + \beta_1 * \text{Feature}_1 + \epsilon \tag{3.1}$$

In the expression 3.1, a target variable Y is explained with a feature called $Feature_1$. 'intercept' is the offset value which determines the common offset observed among the observations (also called bias) and ϵ is the error term. Lastly, β_1 is the slope term which expresses the rate of affect of changes in the value of $Feature_1$ to the value of Y.

On the other hand, multiple linear regression attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data. Every value of the independent variable x is associated with a value of the dependent variable y (Lacey, 1997-1998).

A basic representation of a multiple linear regression is as follows:

$$\begin{aligned} \text{Y} = \text{intercept} &+ \beta_1 * \text{Feature}_1 \\ &+ \beta_2 * \text{Feature}_2 \\ &+ \beta_3 * \text{Feature}_3 \\ &+ \beta_4 * \text{Feature}_4 \\ &. \\ &. \\ &+ \beta_n * \text{Feature}_n + \epsilon \end{aligned}$$

where intercept is the bias in the regression model, β values are the regression coefficients and ϵ is the residual (i.e. error term).

The purpose of the model is to find the set of regression coefficients $\hat{\beta}$ that

minimizes sum of squares of all ϵ values among the observations (Troeger, 2012-2013).

$$(\hat{\epsilon})^2 = \Sigma_{i=1}^{n}(\mathrm{y}_i - (\mathrm{intercept} + \hat{\beta} * \mathrm{Feature}_i))^2 \tag{3.2}$$

As linear regression is the base and the simplest regression, it does not require any parameters during its training process.

LASSO Regression LASSO regression is another linear regression method which shrinks the regression coefficients by imposing a penalty on their size. The name LASSO stands for 'Least Absolute Shrinkage and Selection Operator'. Different than linear regression, the lasso coefficients minimize a penalized residual sum of squares (Hastie, Tibshirani, and Friedman, 2017). This penalization is explained in Chapter 3.3.3.

$$\hat{\beta}_{lasso} = argmin_{\beta}\Sigma_{i=1}^{n}(y_i - (\mathrm{intercept} + \hat{\beta} * \mathrm{Feature}_i))^2 \tag{3.3}$$

subject to

$$\Sigma_{j=1}^{p}|\beta_j| < t \tag{3.4}$$

and t has to be higher than 0. If it is high enough, the additional constraint loses its effect, but for a smaller t, some coefficients will be set to zero. This elimination of variables leads to generation of optimal subset of features. This subsets can be perceived as the selected features for the model. The necessity and the benefit of feature selection is also mentioned in Chapter 3.3.4

A Lasso regressor can be built similar to a Linear regressor with one additional parameter of alpha. Alpha parameter simply determines the L1 penalization (Chapter 3.3.3). Setting alpha to zero leads to same results with a linear

regressor since the penalization is suppressed and setting alpha to one means penalization term is fully applied. This parameter can be varied between zero and one for the optimal regularization.

Non-Linear Models

Multi Layer Perceptron Regressor (MLPR) MLPR is a type of regressor that utilizes the principles of multi layer perceptrons (MLP). MLP is a neural network which includes an input and an output layer. Moreover, there exists one or more hidden layer(s) between the input and the output layer (Bishop, 1995). Each of the input, output and hidden layers include neurons inside. each neuron is a node that represents the current evolution of the model. For example, a neuron at the input layer represents one of the features and each feature has its own input neuron in the architecture.

Each neuron in the hidden layer is a weighted summation of values from the previous layer followed by an activation function such as identity function[1], logistic sigmoid function[2], hyperbolic tangent function[3] and rectified linear unit function[4]. Mentioned weighted summation process enables MLPR technique to carefully adjust the weights of neurons so that the optimal weights can be obtained for the regression. The activation functions are responsible for transforming the weighted summation value to an output value for each neuron.

The output layer is the layer which calculates the final output with again a weighted summation. This layered structure enables MLPR to be used for non-linear datasets as well.

An example MLPR architecture is shown in Figure 3.1.

[1]http://mathworld.wolfram.com/IdentityFunction.html
[2]http://mathworld.wolfram.com/SigmoidFunction.html
[3]http://mathworld.wolfram.com/HyperbolicTangent.html
[4]https://www.kaggle.com/dansbecker/rectified-linear-units-relu-in-deep-learning/code

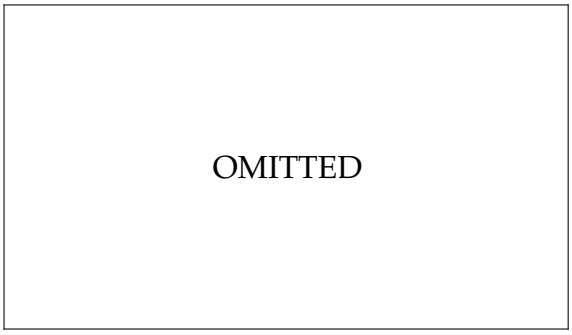

FIGURE 3.1: Example MLPR Architecture (Cakraci, 2017)

Obviously, an infinite number of different MLPR's can be built by varying its structure. The most important parameters for an MLPR are listed below:

⇒ Hidden Layers: The number and size of the hidden layers determines the main structure of the regressor and therefore is quite important. The more layers and the bigger sizes, the more complex model is. So, it is crucial to choose convenient number of layers with appropriate sizes that matches the characteristics of the dataset.

⇒ Activation Function: One of the activation functions mentioned above should be chosen and used during model built.

⇒ Solver: This parameter determines how to optimize each layer's weights during training process. The most popular solver types are Limited-Memory Broyden–Fletcher–Goldfarb–Shanno (l-bfgs) (Saputro and Widyaningsih, 2016), Stochastic Gradient Descent (SGD) (Bottou, 2010), Adam (Diederik P. Kingma, 2015).

⇒ Alpha: Similar to the regularization penalization in Lasso (Chapter 3.2.1), MLPR has an L2 regularization parameter (Chapter 3.3.3).

⇒ Optimization Tolerance: This parameter is helpful to stop training process if the model has stopped improving. Optimization tolerance simply determines the threshold of accuracy improvement of the model to decide when to stop training.

Support Vector Regressor (SVR) Support Vector Regressor is the extended version of Support Vector Machines (SVM) into regression problems. It has been studied extensively and used for several applications such as pattern recognition, hand written character, and text categorization (Joachims, 1997). In its most basic explanation, SVM is an optimization technique that attempts to find a hyperplane in the original input space to separate a given training set correctly and leave as much distance as possible from the closest instances to the hyperplane on both sides (Ojemakinde, 2006).

The application of SVM principle into regression problems (SVR) aims to separate the given training dataset with a hyperplane where the nearest observations on both sides have as much distance from the hyperplane as possible. What makes SVR a non-linear regressor is the kernel functions. These kernel functions map the original input space to a higher dimensionality which is named as the feature space. Then, an hyperplane is sought with the same logic mentioned above (Ojemakinde, 2006).

An illustration of optimal hyperplane selection is depicted in Figure 3.2

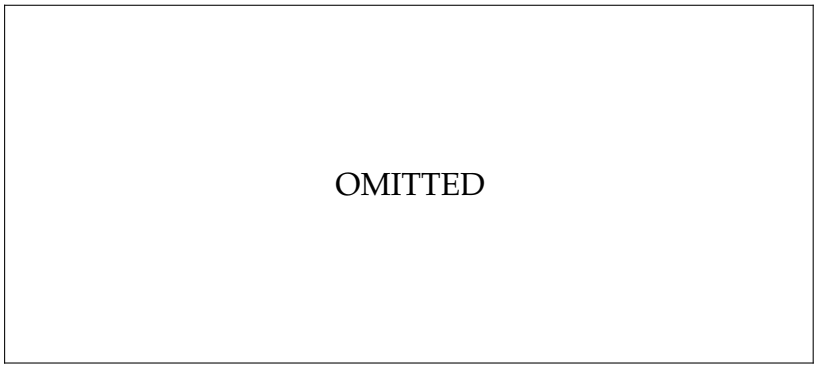

FIGURE 3.2: Optimal Hyperplane Selection of SVR (Kaiping, 2017)

The advantage of this regressor is the fact that it does not focus on reducing the training error but the generalization.

Due to its nature, SVR building process also needs some parameters.

⇒ Error Penalty: This parameter defines the penalization on the model error during training process.

⇒ Kernel: As mentioned above, kernel functions are quite important for SVR as it does the mapping to a higher dimensional plane. Potential kernel functions are linear function[5], polynomial function[6], radial basis function[7] and logistic sigmoid function[8].

⇒ Optimization Tolerance: This parameter has completely purpose with the tolerance parameter of MLPR. It simply determines the threshold of accuracy improvement of the model to decide when to stop training.

3.3 Model Fitting

As mentioned in Chapter 3.2.1, regression models aim to find the optimal intercept and coefficient values among all observations to minimize the error term. This iterative process of minimization is called model fitting. This process consist of variety of challenges that are mentioned in Chapter 3.3.1, Chapter 3.3.2, Chapter 3.3.3 and Chapter 3.3.4.

3.3.1 Overfitting

The fitted model should be assessed in terms of generalization. This means that the model should behave similar in general, both with seen and with unseen data.

If it is observed that the model's performance in terms of accuracy reduces significantly with unseen data, this is an obvious clue of the phenomena called

[5]http://mathworld.wolfram.com/LinearFunction.html
[6]http://www.mathcentre.ac.uk/resources/uploaded/mc-ty-polynomial-2009-1.pdf
[7]https://www.cs.cmu.edu/afs/cs/academic/class/15883-f13/slides/rbf.pdf
[8]http://mathworld.wolfram.com/SigmoidFunction.html

Overfitting. In another words, an overfitted model offers a solution with a high variance. Even small changes in the dataset result in huge differences in the result (ZeBlemoyer, 2012).

The most reliable way for assessing overfitting might be cross validation especially for small and middle size problems (Stone, 1974). K-fold cross validation can be used as it is one of the most used cross validation algorithms. It can be implemented by partitioning data into k equally sized folds and k iterations of training and validation processes are performed where a different fold of the data is used for validation while the rest of the folds (k-1) are used for training within each iterations. The performance of each model on each iteration can be measured using mean squared error (Chapter 4.4). The average MSE can be taken as the overall evaluation metric of the training model as each partition generates the different models which rely on the same data.

3.3.2 Parameter Tuning

Some machine learning algorithms need to be adjusted by using parameters before using the model. The aim of the parameter tuning is to set those parameters to optimal values that leads to the model which perform the best. It should be noted that a bad parameter choice can heavily influence regression performance. Therefore, a proper parameter tuning methodology should be performed in order to build the model.

The most straightforward method is grid search. Grid search simply walks through all potential combinations of parameter sets and tries to find the combination with which the best performance can be achieved (Bergstra and Bengio, 2012).

3.3.3 Regularization

Among the mentioned regression algorithms in Chapter 3.2.1, there exists algorithms that utilizes regularization. Regularization is a technique applied to handle overfitting (Chapter 3.3.1) in regression models. Regularization simply prevents model to capture the noise in data. It penalties the model for its reactions to noisy data and therefore prevents it becoming more complex (Ohlsson, 2010).

There exists two popular regularization techniques of L1 and L2 regularizations:

L1 Regularization

L1 Regularization was born together with the idea of LASSO regression (Tibshirani, 1996) (Chapter 3.2.1) and therefore called as LASSO regularization.

$$(\hat{\beta})^{lasso}(\lambda) = \text{argmin}_{\beta}||\mathbf{y} - \mathbf{X}\beta||_2^2 + \lambda||\beta||_1 \tag{3.5}$$

where the penalization aims to minimize the size of L1 norm[9] of coefficients.

L2 Regularization

L2 regularization is first suggested by (Hoerl and Kennard, 1970) where the aim is to penalize L2 norm[10] of coefficients. The L2 regularization is also called as ridge regularization.

[9] http://mathworld.wolfram.com/L1-Norm.html
[10] http://mathworld.wolfram.com/L2-Norm.html

$$(\hat{\beta})^{ridge}(\lambda) = \text{argmin}_{\beta}||\mathbf{y} - \mathbf{X}\beta||_2^2 + \lambda||\beta||_2^2 \tag{3.6}$$

3.3.4 Feature Selection

Feature selection is the method and process of selecting a subset of features for a predictive model. The process aims to find out the relevant features that should be used in the regression analysis.

There are many purposes for such practice. Firstly, a sub-setting would decrease the number of feature and therefore simplify the model without losing any relevant information. Related to the first aspect, processing (training) time would decrease proportionally with a feature selection process. Thirdly and most importantly, removing irrelevant features would help avoiding overfitting (Chapter 3.3) (Deng, 1998).

Feature selection is not applicable for some algorithms where the algorithm itself takes care of the selection. Within the scope of this study, LASSO regression (Chapter 3.2.1) selects relevant features and shrinks the others while training. Therefore, full feature set should be provided to Lasso without any feature selection. However, other regression methods (Linear Regression, SVR and MLP) (Chapter 3.2.1) are trained after selecting the relevant features.

The feature selection process is applied by utilizing cross validated recursive feature elimination methodology[11]. This method both decides the optimal number of features and the relevant features with a backward elimination procedure.

[11] http://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFECV.html

3.4 Tools

For the analyses conducted, following sources are utilized:

- **Programming Language:** Python 3.0[12]
- **IDE:** Jupyter Notebook[13]
- **Libraries:** numpy[14], pandas[15], scikit-learn[16], matplotlib[17], seaborn[18]
- **Visualization:** Qlik[19]
- **Data Source:** Yahoo Finance[20]

[12]https://www.python.org/download/releases/3.0/
[13]http://jupyter.org
[14]http://www.numpy.org
[15]https://pandas.pydata.org
[16]http://scikit-learn.org/stable/
[17]https://matplotlib.org
[18]https://seaborn.pydata.org
[19]https://www.qlik.com
[20]https://finance.yahoo.com

Chapter 4

Design of Experiment

4.1 Data Preprocessing

Both external and internal data sources are used for the regression algorithms mentioned in Chapter 3.2.1. The detailed data description can be found in Chapter 3. It is always the case that the dataset used may contain null values and outliers which need to be dealt with. Therefore, it is a crucial step to transform raw data to a meaningful format (Quilumba et al., 2014). In the following sections, the methods applied to get the data ready for experiments will be discussed.

4.1.1 Data Transformation

Data transformation is the first and one of the most important steps during data preprocessing. The purpose of this step is to populate intuitive features from raw historical data. For the dataset in hand (Chapter 3.1), data collected from [redacted] [1] for 6 financial market indicator originally has the following columns: [redacted]. With basic transformation methodologies such as calculating [redacted] (4.1)

[1] https://finance.yahoo.com

and [redacted] (4.2), [redacted] features have been populated from [redacted] financial market indicator.

$$[redacted] = log([redacted]_t / [redacted]_{t-1}) \tag{4.1}$$

$$[redacted] = [redacted]_t - [redacted]_t \tag{4.2}$$

Exemplary daily market making performance dataset is not transformed and used without any manipulation.

4.1.2 Data Integration

In order to train the regression algorithms, the data which is collected from different sources is merged into one dataset.

The internal data and the data generated by data transformation process (Chapter 4.1.1) are both indexed with days that each observation belongs to. Then, data belongs to each observation date is merged with respect to their indexes. Table 4.1 shows the number of features and the number of observation after data integration. So, there are [redacted] independent and [redacted] dependent variable and [redacted] observations in hand.

	FEATURES		**OBSERVATIONS**
	Features from External Data Sources	**Features from Internal Data Sources**	
Number	[redacted]	[redacted]	[redacted]

TABLE 4.1: Number of Features and Observations

4.1.3 Data Cleaning

Null Values

As it can be observed in many of the other datasets in finance, the dataset used for this research consists some null values (DiCesare, 2006). There are two main causes for having null values. Firstly, there may be a connection problem between the data source and the model which causes missing information. Second reason of having a null value of a certain financial market indicator could be a national holiday causing the exchange and the financial market being closed on that specific day.

There are many different methodologies that can be applied to deal with the null values. Those methods can be grouped as data imputation and data deletion. Data imputation is to replace the null values with a non-null value based on a rule. Some common techniques are Mean Substitution (MS), Iterative robust model-based imputation (IRMI), Multiple Imputation of Incomplete Multivariate Data (MIIMD), and Random Imputation of Missing Data (RIMD) (Al-Mudhafar and Al-Mudhafar, 2014). Besides, it is also possible to use the value from the previous day. It means that if $indicator_x(t)$ is not available, $indicator_x(t)$ may be populated with $indicator_x(t-1)$.

On the other hand, data deletion technique simply disregards non-complete observations. This means that any observation where at least one of the features is null is removed from the dataset. Even if this approach reduces the number of observations in hand, it guarantees that the data that the model fits is complete and fully reflects the behavior.

Data imputation techniques may be useful in other scopes but it is undesired with financial data since it compromises the structure of the data. Indeed, it is reported that data deletion method is the most popular treatment in financial data based on a survey of papers on financial journals (Kofman and Sharpe, 2003).

Therefore, it is decided to apply data deletion technique to the integrated data (Chapter 4.1.2). In order to validate this decision, a t-test is applied to see if the deletion significantly affects the mean value of features and the response variable. If this is the case, it means that an unnecessary bias is added which would compromise the model. Welch's t-test is chosen to be the specific t-test method as it does not assume equal variance and equal sample size between the test sets (Welch, 1951). The change in means and standard deviations of the features can be observed in Appendix B.1 and standard deviations of the features and the t-test results can be seen in Appendix B.2.

As all of the p-values are higher than 0.05, it can be concluded that null hypothesis that mean of datasets before and after the deletion are same cannot be rejected. Therefore, it is safe to apply data deletion.

Table 4.2 depicts the change in the number of observations after data deletion.

Number of Total Data Points	[redacted]
Number of Null Values	[redacted]
Number of Non Null Values	[redacted]
Number of Observations Before Data Deletion	[redacted]
Number of Observations After Data Deletion	[redacted]

TABLE 4.2: Summary of Number of Observations

Outliers

Occurrence of outliers is a common situation, especially in financial datasets. Therefore, a formal methodology should be applied to handle these outliers values. Two common techniques are found to be truncation and winsorization. Research showed that presence of outliers are handled 55% with winsorization and 40% with truncation (Leone, Minutti-Meza, and Wasley, 2012).

Truncation is removing outlier values where winsorization is substituting outlier values with upper and lower extreme limit values (Leone, Minutti-Meza, and Wasley, 2012). The main difference between truncation and winsorization is simply the fact that truncation results in a smaller data set where winsorization keeps the number of observation same.

As the performance of the regression models are affected by the presence of outliers, both winsorization and truncation methods are found to be beneficial. However, the tails of the distribution of the financial data are extremely important, and indiscriminately modifying some large and small values invalidates many of the statistical analyses that are taken for granted.

The box plot (Figure 4.1) suggests that there exists outliers in the dataset.

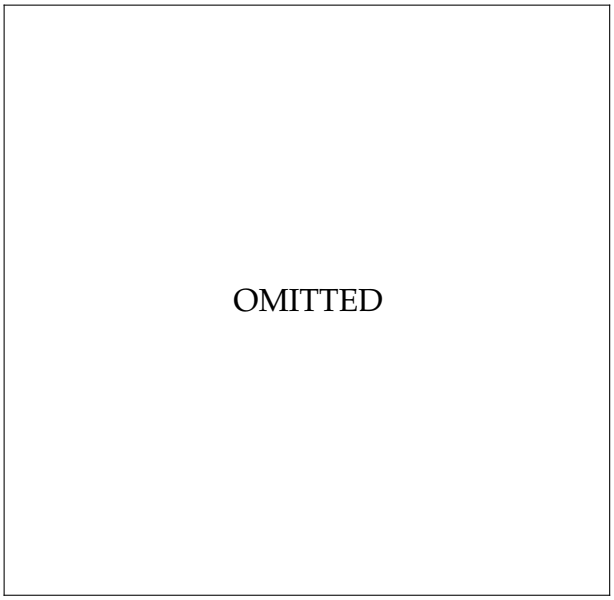

FIGURE 4.1: The Box Plot of the Dependent Variable Before Winsorization or Truncation

Interquartile range method is used for detecting exact outliers. The method calculates the 25th and 75th percentiles and denotes the difference between those values as the interquartile range. Then, any value lower than 25th quartile minus 1.5 times interquartile range or higher than 75th quartile plus 1.5

times interquartile range is perceived as outlier. This methodology is first proposed by John Tukey where he named the interquartile range as the inner fence (Tukey, 1977).

This method is applied to the dependent (response) variable of the regression and ■ outliers have been detected.

Both truncation and winsorization methods are applied and box plots are observed (Figure 4.2 and Figure 4.3).

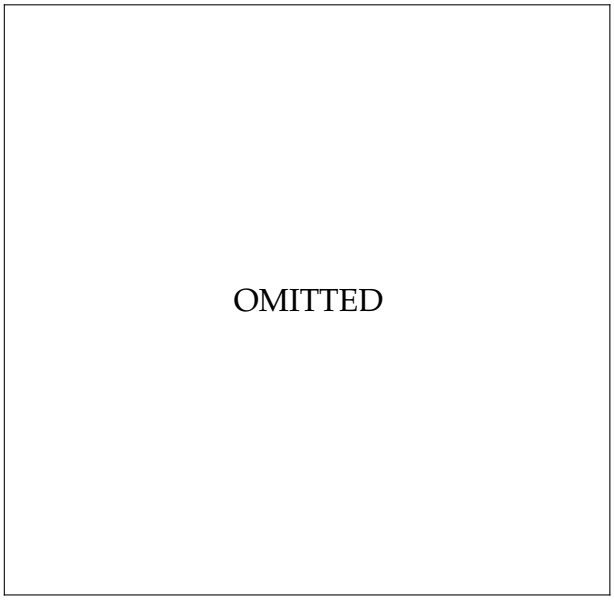

FIGURE 4.2: The Box Plot of the Dependent Variable After Winsorization

Apparently, both methods manage to obtain noise-free datasets. In order to observe and compare the effect of each techniques on the experiment results, experiment is conducted for both truncated version and the winsorized version of the datasets. The change in the number of observations after outlier truncation can be observed in Table 4.3.

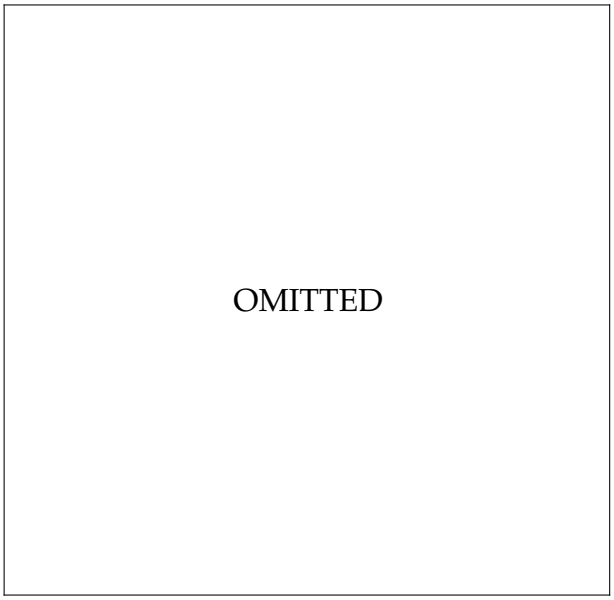

FIGURE 4.3: The Box Plot of the Dependent Variable After Truncation

Number of Observations Before Outlier Truncation	[redacted]
Number of Outlier Observations	[redacted]
Number of Observations After Outlier Truncation	[redacted]

TABLE 4.3: Number of Observations Before and After Outlier Truncation

4.1.4 Data Standardization (z-score)

Another data preprocessing step is to handle the lack of scaling in the dataset. It is quite often to have different features with very large and very small values together. For example, [redacted] of a stock may vary from zero to millions where [redacted] of its price may be in very small scales. Yet, [redacted] may well be a better indicator compared to [redacted] (Olden, 2016). Therefore, it is logical to apply a scaling to the dataset for the sake of a valid machine learning model.

In order to scale down all of the data, standardization is applied. This method basically calculates the ratio of the difference between the observation

and the mean of the observations to the standard deviation of the observation (Aksoy and Haralick, 2016). This number is called as z-score or standard score of the observation as well.

$$x_{i_{standardized}} = \frac{x_i - x_{mean}}{x_{std}} \tag{4.3}$$

Expression 4.3 is applied to both dependent and the independent variables.

4.2 Model Building

Having applied all of the data preprocessing steps to the dataset, model building is initiated. Before training the algorithms mentioned in Chapter 3.2.1, two more actions are taken.

Firstly, a feature selection procedure mentioned in Chapter 3.3.4 is applied to the preprocessed data. As mentioned in Chapter 3.2.1 and in Chapter 3.3.4, LASSO regression already includes a natural feature selection during its training process. The rest of the algorithms (Linear Regression, MLPR and SVR) (Chapter 3.2.1) is subsetted manually with the mentioned feature selection procedure. The number of features left after the feature selection process can be observed in Chapter 5.1.

Moreover, parameter tuning (Chapter 3.3.2) is also handled just before the actual training of each algorithm. A grid search procedure (Chapter 3.3.2) is applied on the dataset so that optimal parameters (i.e. hyperparameters) can be obtained. Parameter grids for LASSO is displayed in Table 4.4 where parameters grids for MLPR can be observed in Table 4.5 and parameter grids for SVR is shown in Table 4.6. Hyperparameters for each algorithm can be seen in Chapter 5.2.

	Parameter Grid for LASSO
Alpha	[redacted]

TABLE 4.4: Parameter grids for LASSO Regression

	Parameter Grids for MLPR
Hidden Layers	[redacted]
Activation Function	[redacted]
Solver	[redacted]
Alpha	[redacted]
Optimization Tolerance	[redacted]

TABLE 4.5: Parameter grids for MLPR

	Parameter Grid for SVR
Error Penalty	[redacted]
Kernel Function	[redacted]
Optimization Tolerance	[redacted]

TABLE 4.6: Parameter Grids for SVR

Finally, models are built with the hyperparameters and the performance metrics are observed (Chapter 4.3). In order to assess the overall improvement suggested by each algorithm, a dummy model is also introduced. This dummy model would constantly predict the dependent variable as the median of all observations. Since this model does not offer any learning, comparison of its performance with the actual learning models would reveal the learning improvement.

All results related to the experiments can be observed for each of the algorithms can be observed in Chapter 5.

The overall flow during the model build is also itemized below:

⇒ Apply feature selection

⇒ Split data into k folds

⇒ Set a parameter grid space for the potential values of parameters of the algorithm

⇒ Use each k fold as validation dataset at each iteration, and the rest k-1 folds as training datasets

⇒ Apply a grid search to each validation dataset and find out parameter sets with the highest accuracies on validation dataset

⇒ Select the parameter set which appears in most of the selected parameter sets among k iterations as hyperparameters

⇒ Use hyperparameters to train the whole training dataset to fit the final model

⇒ Observe performance metrics and plots

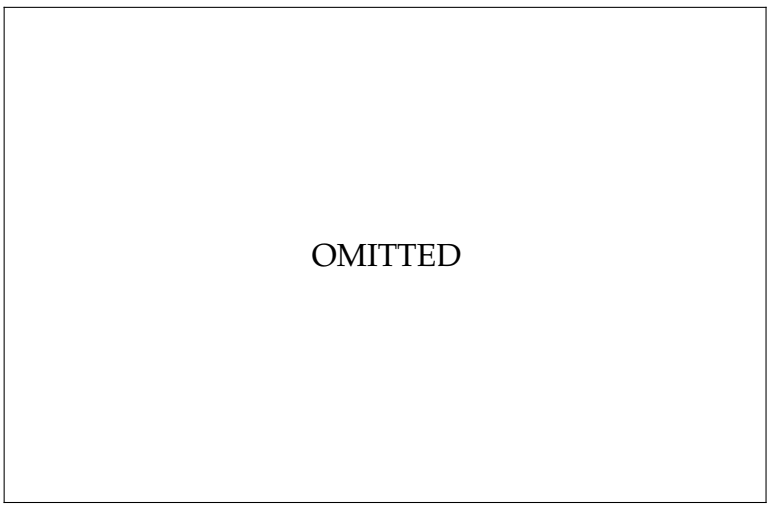

FIGURE 4.4: Example Cross Validated Procedure (Raschka, 2013)

4.3 Evaluation Measures

The performance of the experiments are assessed with the following common evaluation metric:

4.3.1 Mean Squared Error

One of the most popular metric to measure the performance of a regression model is mean squared error.

$$\mathrm{MSE}(\mathrm{y}, \hat{\mathrm{y}}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{4.4}$$

Expression 4.4 calculates the squared average of the difference between the actual data and the prediction.

4.3.2 Computation cost

Computation time is measured for each of the regressor to measure their time expenses by setting a timer. Time is a important criteria when it becomes real time applications which have large scale data and limited time for training the model. Time expenses are computed and compared to give insights about algorithms regarding their computation costs. As this study is designed to be used in daily data in a real time application, computational cost is not the primary performance criteria.

Chapter 5

Experiment and Results

The experiment is conducted with the algorithms mentioned in Chapter 3.2.1 on the preprocessed dataset (Chapter 4.1) and evaluated with measures explained in Chapter 4.3. All of the results will be shown in two versions. In one of the versions; outliers in the dataset are truncated and in the other, they are winsorized (Chapter 4.1.3).

Since one of the aims of this study is find the best accurate model, in this section, all the possible implementation and their results will be shown.

5.1 Feature Selection Results

Feature selection procedure is applied to Linear Regression, MLPR and SVR algorithms and the following changes are observed (Table 5.1):

	Truncated Outliers	**Winsorized Outliers**
Number of Features Before Feature Selection	■	■
Number of Features After Feature Selection	■	■

TABLE 5.1: Number of Features Before and After Feature Selection for Linear Regression, MLPR and SVR Algorithms

As LASSO regression applies its own feature selection, abovementioned procedure is not applied during its model building. Yet, the number of features whose coefficients are shrunk during the training process of LASSO is observed for comparison purposes and displayed in Table 5.2.

	Truncated Outliers	**Winsorized Outliers**
Number of Features Before Feature Selection	[redacted]	[redacted]
Number of Features After Feature Selection	[redacted]	[redacted]

TABLE 5.2: Number of Features Before and After Natural Feature Selection of LASSO

5.2 Hyperparameters

Hyperparameters obtained via parameter tuning process (Chapter 3.3.2) are displayed in this section.

	Truncated Outliers	**Winsorized Outliers**
Alpha	[redacted]	[redacted]

TABLE 5.3: Hyperparameters of LASSO Model with Truncated Outliers and with Winsorized Outliers

	Truncated Outliers	**Winsorized Outliers**
Hidden Layers	[redacted]	[redacted]
Activation Function	[redacted]	[redacted]
Solver	[redacted]	[redacted]
Alpha	[redacted]	[redacted]
Optimization Tolerance	[redacted]	[redacted]

TABLE 5.4: Hyperparameters of MLPR Model with Truncated Outliers and with Winsorized Outliers

	Truncated Outliers	Winsorized Outliers
Error Penalty	[redacted]	[redacted]
Kernel Function	[redacted]	[redacted]
Optimization Tolerance	[redacted]	[redacted]

TABLE 5.5: Hyperparameters of SVR Model with Truncated Outliers and with Winsorized Outliers

5.3 Accuracy Results

Table 5.6 summarizes the accuracy results of each regression algorithm where outliers are truncated. On the other hand, Table 5.7 has the MSE results for each regression algorithm where outliers are winsorized.

	Training MSE	Test MSE
Linear Regression	[redacted]	[redacted]
LASSO Regression	[redacted]	[redacted]
MLPR	[redacted]	[redacted]
SVR	[redacted]	[redacted]

TABLE 5.6: Accuracy Results of Regression Algorithms with Truncated Outliers

	Training MSE	Test MSE
Linear Regression	[redacted]	[redacted]
LASSO Regression	[redacted]	[redacted]
MLPR	[redacted]	[redacted]
SVR	[redacted]	[redacted]

TABLE 5.7: Accuracy Results of Regression Algorithms with Winsorized Outliers

	Truncated Outliers MSE	Winsorized Outliers MSE
Dummy Model	[redacted]	[redacted]

TABLE 5.8: Accuracy Results of Dummy Model

5.4 Computational Cost

Table 5.9 shows the system computation durations of each algorithm.

	Regression Algorithms with Truncated Outliers	**Regression Algorithms with Winsorized Outliers**
Linear Regression	[redacted]	[redacted]
LASSO Regression	[redacted]	[redacted]
MLPR	[redacted]	[redacted]
SVR	[redacted]	[redacted]

TABLE 5.9: Computation Times of Regression Algorithms

5.5 Visualization

In order to visualize the results, prediction results of the algorithms are deployed to a dashboard together with the actual trading performances. This dashboard enables comparison of any algorithm per observation, interactively. For this purpose Qlik data visualization tool[1] is utilized.

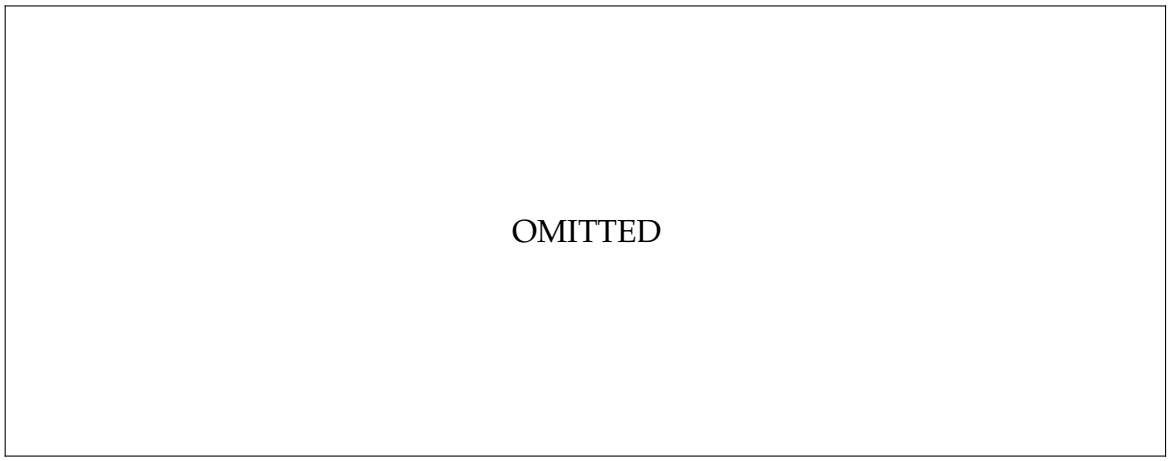

FIGURE 5.1: Visualization of Regression Models on Dashboard

[1] https://www.qlik.com

Chapter 6

Discussion

The motivation behind this study was to apply machine learning algorithms to understand the underlying factor of a business and predict the performance of the business using these factors.

[redacted]

Feature selection procedure (Chapter 5.1) applied to Linear Regression, MLPR and SVR resulted in a significant amount of decrease in the number of features for the dataset where outliers are winsorized. [redacted]

[redacted]

[redacted]. In order to assess if feature selection served its purpose of limiting overfitting, learning curves of models are investigated. Basically, a learning curve indicates the behavior of the training and cross-validated test MSE's as the number of observations trained increases. Training MSE and cross-validated test MSE curve should converge in an optimal learning curve. Appendix C depicts the learning curves of Linear Regression, LASSO regression, MLPR and SVR for both datasets (outliers truncated and outliers winsorized, respectively). As all of the learning curves behave in a convergence pattern, it is concluded that feature selection procedure is successfully served its purpose.

Hyperparameters obtained for each of the algorithm are visualized in Chapter 5.2. [redacted]

Accuracy results (Chapter 5.3) indicates that [redacted]

Comparison of test mean squared errors with dummy model (Table 5.8) clearly indicates that [redacted]

MSE observations suggest that [redacted]. A bar chart that compares MSE results of the algorithms can be observed in Figure 6.1.

Even if it is not the major achievement of this study, computational costs

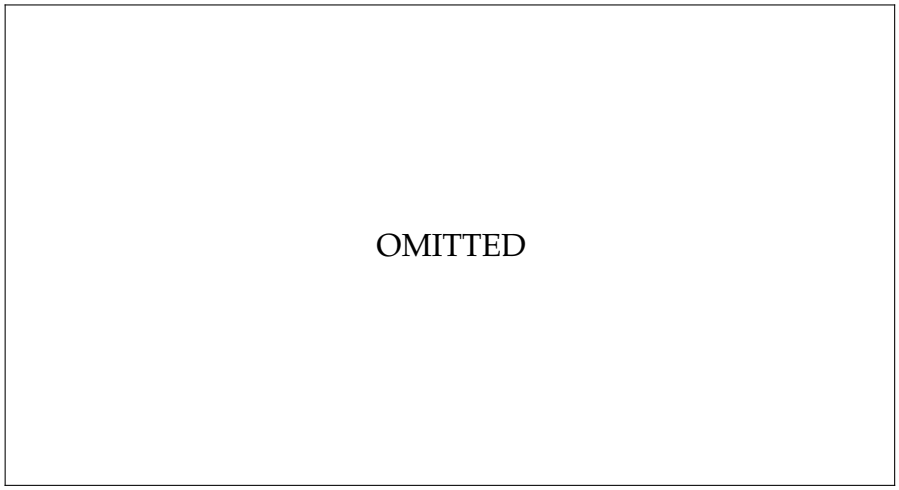

FIGURE 6.1: The Learning Curve of Linear Regression with Truncated Outliers

are also observed and recorded (Chapter 5.9). [redacted]

Lastly, residual Q-Q plots of each model have been plotted. The purpose of this plot is to observe if residuals (the differences between the estimated output and the actual value) of the algorithms are randomly distributed. If this is the case, it means that algorithms successfully absorbed all of the information on the dataset and there are no pattern left in the residuals. Q-Q plots basically compare quantile distribution of each residual with a normal distribution. Appendix D visualizes the Q-Q plots of each model. [redacted]

[redacted].

6.1 Limitations

In this section, the limitations of the research will be discussed. It should be noted that most of the limitations are related to the dataset. To begin with, one of the most challenging limitations is the insufficient historical training data that is available regarding high frequency daily market making performance. As discussed in Chapter 3.1; this was a choice to keep dataset consistent, yet it limited the overall success of the model.

Another limitation is that there exist infinite number of financial market indicators that may have affect on high frequency market making performance. The logic here was to select the most relevant factors. However, only a limited number of market indicators could be tried due to time constraints.

Also, it should be noted that this research is only conducted to find out potential affects and relationship between market factors on high frequency market making performance. Therefore, it is never possible completely estimate other non-financial factors that has affect on trading performance. Even the market offers the opportunity to perform better, there may be technical or behavioral reasons that keeps the performance low. Those effects are not within the scope of this study.

6.2 Future Work

As mentioned in Chapter 6.1, there exist a few limitations that level the overall success of the study down. Among them, the insufficiency of historical data and market indicators that have not been tried out yet could be good candidates for future work.

Assuming the market conditions and trading strategies won't change drastically, it may be wise to re-run the experiments when whole data from 2018 becomes available and observe if the accuracies of the implemented models change. They may or may not improve as it is not a trivial task to understand if the models have already reached to a stable state. Learning curves (Appendix C) may offer a solution since it depicts whether there is room for more convergence in the model.

Secondly, it may be an helpful work to list more and more financial market indicators and utilize their historical data to seek for any new indicator that has affect on trading performance. Please note that a proper feature selection procedure (Chapter 3.3.4) is a must as it is quite dangerous to use too many features where number of observations are limited.

Lastly, it is planned to conduct another study on the same dataset (high frequency trading daily market making) which perceives the dataset as time-series and aims to apply a future forecasting process. A preview of this study can be observed in Chapter 7.

Chapter 7

Preview: Time-Series Forecasting

7.1 Introduction & Goal

This chapter is reserved for a preview of a study conducted on the same business scope of high frequency trading daily market making performance. As it is also clarified in Chapter 1.1, one of the main goals of this study is to investigate business performance from variety of aspects and thus this preview is one of those aspects in which the problem is approached with a time-series perception.

This approach led to the question of whether trading performances in different regions (i.e. different financial markets across the world) have any affect on each other and whether it is possible to forecast future high frequency trading daily market making performances on each financial market in different regions.

So, the main goal of this chapter is to provide a preview of a causality analysis across business performances in different geographical regions and to forecast future trading performances.

7.2 Data

For this specific purpose (Chapter 7.1), new datasets from same business scope are introduced. In order to analyze any causality or in order to forecast future performances, high frequency daily market making performances are obtained for Region1, Region2 and Region3 all between 2nd January 2015 and 14th May 2018. In order to align the number of observations, the days where at least one of the data is missing are deleted (similar approach with Chapter 4.1.3). Table 7.1 depicts the number of observations used for the analysis.

	Number of Observations Before Data Deletion	**Number of Observations After Data Deletion**
Region1	[redacted]	[redacted]
Region2	[redacted]	[redacted]
Region3	[redacted]	[redacted]

TABLE 7.1: Number of Observations for each Region Before and After Data Deletion

Obtained datasets are utilized without further data preprocessing steps. However, a simple data shifting is applied due to the nature of the handled problem. As the purpose of this specific study is to reveal relational connections, used data should be aligned properly. For example, any effect of the performance on Region1 to performance on Region3 comes with one day lag since new trading day starts from Region3, but potential effect of Region3 region performance on Region2 performance could be observed within same day. Therefore, below version of each dataset (Table 7.2) are prepared and used at necessary steps of the analysis.

7.3 Methodology

Both analyses conducted have the precondition of datasets being stationary. Therefore, a proper stationarity test is applied on datasets in Chapter 7.3.1.

Dataset Name	Description
$Region1$	will be used when needed directly
$Region1_{lagged}$	will be used when needed with 1-day lag
$Region2$	will be used when needed directly
$Region2_{lagged}$	will be used when needed with 1-day lag
$Region3$	will be used when needed directly
$Region3_{lagged}$	will be used when needed with 1-day lag

TABLE 7.2: Prepared Datasets for Time-Series Analyses

Then, any potential causality is sought in Chapter 7.3.2 and future forecasting is applied in Chapter 7.3.3.

7.3.1 Stationarity

Stationarity is the term used for time-series whose mean, variance, autocorrelation etc. stays constant over time (Nau, 2014). Stationarity is quite important since any pattern in mentioned statistics of datasets would result in misguiding model results.

Therefore, it is a prerequisite to find out if time-series' are stationary. For this purpose, a unit root test is applied on each dataset ($Region1$, $Region2$, $Region3$). Results of ADF test (Augmented-Dickey-Fuller test)[1] can be observed in Appendix E.1. Null hypothesis in ADF test is that the datasets are not stationary. The results suggest that null hypothesis can be rejected as none of the p-values are significant. Thus, datasets are stationary.

7.3.2 Granger Causality

Having proved that datasets are stationary, a causality test is applied to find out if trading performances in different regions affect each other. The most

[1] http://www.statsmodels.org/dev/generated/statsmodels.tsa.stattools.adfuller.html

popular method for testing causality is Granger Causality.

Granger causality is a test designed for revealing insights of whether a time-series is driven by one other. Moreover, it provides an estimation whether a forecast on a time-series could be improved with the information in another time-series (Granger, 1969).

Granger causality test is applied to all combinations of datasets with taking the respective orders into consideration. As mentioned in Chapter 7.2, causality of a region's performance on a performance whose region is in a later time zone could only be observed by utilizing one-day lagged version of the latter dataset. Table 7.3 indicates the Granger Causality tests applied with the dataset versions used. For example, the first test aims to find out if performance of Region1 has any affect on the performance of Region2. Since markets in Region1 closes and then Region2 markets trade on one trading day afterwards, a lagged version of $Region1$ performance is used.

Causee	**Causer**	**Used Datasets**
$Region2$	$Region1$	$Region2$, $Region1_{lagged}$
$Region3$	$Region2$	$Region3$, $Region2_{lagged}$
$Region3$	$Region1$	$Region3$, $Region1_{lagged}$
$Region1$	$Region2$	$Region1$, $Region2$
$Region2$	$Region3$	$Region2$, $Region3$
$Region1$	$Region3$	$Region1$, $Region3$

TABLE 7.3: List of Granger Causality Tests Applied

The result of each test can be observed in Appendix E.2. All 6 tests have rejected the null hypothesis with very small p-values. Therefore, it is concluded that each of the performances have affect on each other with the proper timing order.

7.3.3 Forecasting

ARMA Model

ARMA (Autoregressive Moving Average) model is a time-series model which combines moving averages with autoregressions (Steel, 2014). Moving average model simply claims that the model output is linearly relational with the current and past observations of the time-series where autoregressive models suggest that the model output is a linear combination of its past observations. Therefore, an ARMA(p,q) model simply use historical p observations in the auto-regression term and the moving averages derived from the last q observations (Steel, 2014).

ARMA(p,q) models have precondition of time-series being stationary. In case of a non-stationary time-series, an ARIMA (AR Integrated MA) model should be utilized where parameter d stands for differencing which handles non-stationarity problem. Fortunately, stationarity test conducted in Chapter 7.3.1 concluded that none of the datasets are non-stationary. So, an ARMA(p,q) model is sufficient.

Having decided on the model, the other very important step is to determine parameters of ARMA (p,q). Autocorrelation and Partial-Autocorrelation are significant measures for the decision of p and q (Mills and Markellos, 2008). Autocorrelation is the correlation between the value of current observation and the value of historical observations. Therefore, an ACF (Autocorrelation Function) plot simply visualizes significant correlations of the current observations with historical observations. Any correlation close to zero is assumed to indicate randomness. Therefore, q parameter of an ARMA(p,q) model could be estimated by observing the latest significant historical correlation. For example if ACF plot becomes insignificant after 5 lags, this means that time-series' current observation has correlation with up to 5 past observations and q parameter could be used as 5. PACF (Partial Autocorrelation Function) is the plot of partial autocorrelations of the time-series observations. Similar to ACF,

p value could be estimated from PACF plot by observing the lag level where partial autocorrelation becomes insignificant.

ACF and PACF plots for each dataset can be observed in Appendix E.3. Table 7.4 lists respective parameters decided with the help of plots.

ARMA Model	**p**	**q**
Region1 Performance Forecast	[redacted]	[redacted]
Region2 Performance Forecast	[redacted]	[redacted]
Region3 Performance Forecast	[redacted]	[redacted]

TABLE 7.4: Parameters of ARMA Models Built

Then, models are built with the parameters in Table 7.4. Preliminary results obtained from the model is presented in Chapter 7.4.

ARMA with Exogenous Regressors

Result of Granger Causality test guided the study to also focus on an ARMA model where information that could be obtained from other datasets is also included. For this purpose, ARMA models with exogenous variables are built.

Each model is built where endogenous variable is set to be the performance of the region that is desired to be forecasted and exogenous variables are set to be the remaining two datasets, again with proper timing order. Parameters of p and q are used as mentioned in Table 7.4. Table 7.5 lists the built models with endogenous and exogenous variables.

The results and comparison with ARMA model is presented in Chapter 7.4.

7.4 Preliminary Results

In order to assess performance of the models on different regions, three measurement metric is introduced.

ARMA Model	Endogenous Variable	Exogenous Variables
Region1 Performance Forecast	*Region*1	*Region*2, *Region*3
Region2 Performance Forecast	*Region*2	$Region1_{lagged}$, *Region*3
Region3 Performance Forecast	*Region*3	$Region1_{lagged}$, $Region2_{lagged}$

TABLE 7.5: Built ARMA Models

The first metric is AIC (Akaike Information Criterion) (Akaike, 1974). This metric is a statistical estimator of the relative quality of a model. Therefore, it is used for model selection means. Basically, it claims that the model with the smallest AIC is the best among the possible models and thus should be selected. Indeed, AIC metric of ARMA models are observed with variety of p and q parameters as a cross-check and the smallest AIC has been observed with p and q values in Table 7.4. The other metric is BIC (Bayesian Information Criterion) (Schwarz, 1978) and is again an estimator of relative quality.

The final metric is MAPE (Mean Absolute Percentage Error) (Goodwin and Lawton, 1999). The metric is a common error measure for forecasting models and therefore used here.

Table 7.6 provides the preliminary test results of ARMA models and ARMA models with exogenous regressors.

Region	Model	AIC	BIC	MAPE
Region1	ARMA	[redacted]	[redacted]	[redacted]
	ARMA with Exogenous Regressors	[redacted]	[redacted]	[redacted]
Region2	ARMA	[redacted]	[redacted]	[redacted]
	ARMA with Exogenous Regressors	[redacted]	[redacted]	[redacted]
Region3	ARMA	[redacted]	[redacted]	[redacted]
	ARMA with Exogenous Regressors	[redacted]	[redacted]	[redacted]

TABLE 7.6: Preliminary Test Results of Built ARMA Models

7.5 Discussion

In this preview, another approach is performed to analyze business performance. High frequency trading performances in different regions are utilized as time-series. On those time-series', Granger Causalities are observed and ARMA models and ARMA models with exogenous variables are built. Then, initial results are presented. What Table 7.6 claims is that [redacted].

This study will continue by focusing on more complex models that could improve the preliminary results obtained so far. An exemplary approach may well be an hybrid of ARMA and a neural network (Zhang, 2003).

Chapter 8

Conclusion

This thesis has investigated an application of machine learning. More specifically, it sought for a suitable algorithm that could be utilized for understanding/explaining a business performance. It has focused on high frequency trading daily market making performance and ultimately aimed to reveal any relationship between certain financial market indicators and the performance.

For the purpose, an intensive literature research is conducted and current applications are observed for guidance. Furthermore, cutting-edge machine learning methods are analyzed and a work-flow with fully scientific approach is planned. All necessary preprocessing and processing are applied to data and decided algorithms are trained. In the end, stable models and moderately promising results are obtained. Therefore, it is possible to state that there exists a relationship between the utilized financial market indicators and high frequency daily market making performance.

For the same purpose, another study is conducted where the scope is to find out causal relationships between high frequency daily market making performances across different geographical regions and to forecast future performances. A preview of this study is presented and its preliminary results are discussed. The initial findings indicate that market making performance across regions indeed affects each other. Besides, it is possible to build forecasting models.

However, work is not yet done. This study plays an introductory role and therefore it is quite possible to improve what has been done. As machine learning techniques are improving and new approaches become available each day, this study could offer better outcomes with the introduction of more suitable technical approaches to this specific business case.

Appendix A

Example Raw Financial Market Indicator Data

REDACTED DUE TO CONFIDENTIALITY

Appendix B

Null Value Deletion

B.1 Mean and Standard Deviation of Features Before and After Data Deletion

REDACTED DUE TO CONFIDENTIALITY

B.2 t-test Results for the Effect of Data Deletion

REDACTED DUE TO CONFIDENTIALITY

Appendix C

Learning Curves

REDACTED DUE TO CONFIDENTIALITY

Appendix D

Residual Q-Q Plots

REDACTED DUE TO CONFIDENTIALITY

Appendix E

Test Results for Preview: Time-Series Forecasting

E.1 Stationarity Test Results

REDACTED DUE TO CONFIDENTIALITY

E.2 Causality Test Results

REDACTED DUE TO CONFIDENTIALITY

E.3 ACF and PACF Plots

REDACTED DUE TO CONFIDENTIALITY

E.4 Actual vs. Forecasted Plots

REDACTED DUE TO CONFIDENTIALITY

E.5 Residual Q-Q Plots

REDACTED DUE TO CONFIDENTIALITY

E.6 Autocorrelations of Residuals

REDACTED DUE TO CONFIDENTIALITY

Bibliography

Akaike, H. (1974). "A new look at the statistical model identification". In: *IEEE Transactions on Automatic Control* 19.6, pp. 716–723. ISSN: 0018-9286. DOI: 10.1109/TAC.1974.1100705.

Aksoy, Selim and Robert M. Haralick (2016). "Feature Normalization and Likelihood-based Similarity Measures for Image Retrieval". In: URL: http://www.cs.bilkent.edu.tr/~saksoy/papers/prletters01_likelihood.pdf.

Al-Mudhafar, Watheq and Ali Al-Mudhafar (2014). "Comparative Statistical Algorithms for Imputation of Missing Measurements in Petrophysical Data". In:

Aldridge, Irene (2011). In: *High-Frequency Trading: A Practical Guide to Algorithmic Strategies and Trading Systems*. URL: https://books.google.nl/books?id=8QpIsVUMhmEC&printsec=frontcover.

Alexander, Matt et al. (2016). *Revenue management and predictive analytics-Illuminate the future*. URL: http://www.ey.com/Publication/vwLUAssets/ey-illuminate-the-future/$FILE/ey-illuminate-the-future.pdf.

Avdalović, Snežana Milošević and Ivan Milenković (2017). "Impact Of Company Performances On The Stock Price: An Empirical Analysis On Select Companies In Serbia". In: *Economics of Agriculture*. URL: https://scindeks-clanci.ceon.rs/data/pdf/0352-3462/2017/0352-34621702561M.pdf.

Bergstra, James and Yoshua Bengio (2012). "Random Search for Hyper-parameter Optimization". In: *J. Mach. Learn. Res.* 13.1, pp. 281–305. ISSN: 1532-4435. URL: http://dl.acm.org/citation.cfm?id=2503308.2188395.

Biais, Bruno (2011). "High Frequency Trading". In: URL: http://www.eifr.eu/files/file2220879.pdf.

Bishop, Christopher M. (1995). *Neural Networks for Pattern Recognition*. New York, NY, USA: Oxford University Press, Inc. ISBN: 0198538642.

Bottou, Léon (2010). "Large-Scale Machine Learning with Stochastic Gradient Descent". In: *Proceedings of COMPSTAT'2010*. Ed. by Yves Lechevallier and Gilbert Saporta. Heidelberg: Physica-Verlag HD, pp. 177–186. ISBN: 978-3-7908-2604-3.

Cakraci, Necmettin (2017). *Basic concepts in deep learning applications: perceptron, score function and error function (loss function)*. URL: https://tr.linkedin.com/pulse/derin-ÃűÄ§renme-uygulamalarÄśnda-temel-kavramlar-skor-ve-Ãğarkacİś.

Conerly, Bill (2014). *High Frequency Trading Explained Simply*. URL: https://www.forbes.com/sites/billconerly/2014/04/14/high-frequency-trading-explained-simply/#122127cb3da8.

Deng, Kan (1998). "OMEGA: Online Memory-Based General Purpose System Classifier". In: pp. 117–125. URL: https://www.cs.cmu.edu/~kdeng/thesis/thesis.pdf.

DiCesare, Giuseppe (2006). "Imputation, Estimation and Missing Data in Finance". In:

Diederik P. Kingma, Jimmy Lei Ba (2015). "ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION". In: URL: https://arxiv.org/pdf/1412.6980.pdf.

Gajewar, Amita and Gagan Bansa (2016). "Revenue Forecasting for Enterprise Products". In: URL: https://arxiv.org/pdf/1701.06624.pdf.

Goodwin, Paul and Richard Lawton (1999). "On the asymmetry of the symmetric MAPE". In: *International Journal of Forecasting* 15.4, pp. 405 –408. ISSN: 0169-2070. DOI: https://doi.org/10.1016/S0169-2070(99)00007-2. URL: http://www.sciencedirect.com/science/article/pii/S0169207099000072.

Gordon, James A. (2017). "Algorithmic Trading". In: URL: https://www.linkedin.com/pulse/algorithm-trading-james-a-gordon/.

Granger, C. W. J. (1969). "Investigating Causal Relations by Econometric Models and Cross-spectral Methods". In: *Econometrica* 37.3, pp. 424–438. ISSN: 00129682, 14680262. URL: http://www.jstor.org/stable/1912791.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman (2017). In: *The Elements of Statistical Learning Data Mining, Inference, and Prediction*, pp. 68–69. URL: https://web.stanford.edu/~hastie/ElemStatLearn/printings/ESLII_print12.pdf.

Henke, Nicolaus et al. (2016). *THE AGE OF ANALYTICS: COMPETING IN A DATA-DRIVEN WORLD*. URL: https://www.mckinsey.com/~/media/McKinsey/Business%20Functions/McKinsey%20Analytics/Our%20Insights/The%20age%20of%20analytics%20Competing%20in%20a%20data%20driven%20world/MGI-The-Age-of-Analytics-Full-report.ashx.

Hoerl, Arthur E. and Robert W. Kennard (1970). "Ridge Regression: Biased Estimation for Nonorthogonal Problems". In: *Technometrics* 12.1, pp. 55–67. DOI: 10.1080/00401706.1970.10488634. eprint: https://www.tandfonline.com/doi/pdf/10.1080/00401706.1970.10488634. URL: https://www.tandfonline.com/doi/abs/10.1080/00401706.1970.10488634.

Joachims, Thorsten (1997). "Text Categorization with Support Vector Machines". In:

Kaiping, Li (2017). *A discussion on electricity forecasting technology for demand bidding of joint users*. URL: http://www.syscom.com.tw/ePaper_Content_EPArticledetail.aspx?id=634&EPID=241&j=3&HeaderName=çăŤçŹijæŰřèęŰçŢŒ.

Kanagal, Kapil, Yu Wu, and Kevin Chen (2017). In: *Market Making with Machine Learning Methods*. URL: https://web.stanford.edu/class/msande448/2017/Final/Reports/gr4.pdf.

Kofman, Paul and Ian G. Sharpe (2003). "Using Multiple Imputation in the Analysis of Incomplete Observations in Finance". In: *Journal of Financial Econometrics* 1.2, pp. 216–249. DOI: 10.1093/jjfinec/nbg013. eprint: /oup/backfile/content_public/journal/jfec/1/2/10.1093/jjfinec/nbg013/2/nbg013.pdf. URL: http://dx.doi.org/10.1093/jjfinec/nbg013.

Lacey, Michelle (1997-1998). "STAT101 Course Notes - University of Yale". In: URL: http://www.stat.yale.edu/Courses/1997-98/101/linmult.htm.

Leone, Andrew J., Miguel Minutti-Meza, and Charles Wasley (2012). "Outliers and Inference in Accounting Research". In:

Mills, T.C. and R.N. Markellos (2008). *The Econometric Modelling of Financial Time Series*. Cambridge University Press. ISBN: 9781139470810. URL: https://books.google.nl/books?id=nV93bk3mhNoC.

Mohamad, H.H., A.H. Ibrahim, and H.H. Massoud (2013). "Assessment of the expected construction company's net profit using neural network and multiple regression models". In: *Ain Shams Engineering Journal* 4.3, pp. 375 – 385. ISSN: 2090-4479. DOI: https://doi.org/10.1016/j.asej.2012.11.008. URL: http://www.sciencedirect.com/science/article/pii/S2090447912001165.

Nau, Robert (2014). *Stationarity and differencing*. URL: https://people.duke.edu/~rnau/411diff.htm.

Ohlsson, Henrik (2010). "Regularization for Sparseness and Smoothness - Applications in System Identification and Signal Processing". In: p. 6. URL: http://users.isy.liu.se/en/rt/ohlsson/HOthesis.pdf#page35.

Ojemakinde, Bukola Titilayo (2006). "Support Vector Regression for Non-Stationary Time Series". In: p. 28. URL: http://trace.tennessee.edu/cgi/viewcontent.cgi?article=3107&context=utk_gradthes.

Olden, Magnus (2016). "Predicting Stocks with Machine Learning". In:

Quilumba, F. L. et al. (2014). "An overview of AMI data preprocessing to enhance the performance of load forecasting". In: pp. 1–7. ISSN: 0197-2618. DOI: 10.1109/IAS.2014.6978369.

Raschka, Sebastian (2013). *Machine Learning FAQ*. URL: https://sebastianraschka.com/faq/docs/evaluate-a-model.html.

Saputro, Dewi and Purnami Widyaningsih (2016). *NEWTON, BROYDEN-FLETCHER-GOLDFARB-SHANNO (BFGS) AND LIMITED MEMORY BROYDEN-FLETCHER-GOLDFARB-SHANNO (L-BFGS) METHODS FOR THE PARAMETER ESTIMATION IN GEOGRAPHICALLY WEIGHTED ORDINAL LOGISTIC REGRESSION MODEL (GWOLR)*.

Schwarz, Gideon (1978). "Estimating the Dimension of a Model". In: *Ann. Statist.* 6.2, pp. 461–464. DOI: 10.1214/aos/1176344136. URL: https://doi.org/10.1214/aos/1176344136.

Shen, Shunrong, Haomiao Jiang, and Tongda Zhang (2012). *Stock Market Forecasting Using Machine Learning Algorithms*.

Steel, Dr. Allan (2014). *Predictions in Financial Time Series Data*. URL: http://www.dataminingmasters.com/uploads/studentProjects/TimeSeriesData.pdf.

Stone, M. (1974). "Cross-validatory choice and assessment of statistical predictions". In: *Journal of the Royal Statistical Society: Series B* 1.2, pp. 111–147. DOI: 10.1093/jjfinec/nbg013. URL: http://www.scirp.org/(S(vtj3fa45qm1ean45vvffcz55))/reference/ReferencesPapers.aspx?ReferenceID=997688.

Sunthornjittanon, Supichaya (2015). "Linear Regression Analysis on Net Income of an Agrochemical Company in Thailand". In: URL: https://pdxscholar.library.pdx.edu/cgi/viewcontent.cgi?article=1156&context=honorstheses.

Tibshirani, Robert (1996). "Regression Shrinkage and Selection via the Lasso". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 58.1, pp. 267–288. ISSN: 00359246. URL: http://www.jstor.org/stable/2346178.

Troeger, Vera (2012-2013). "The simple linear Regression Model". In: *PO906: Quantitative Data Analysis and Interpretation - The University of Warwick* Week 5-6-7. URL: https://warwick.ac.uk/fac/soc/economics/staff/vetroeger/teaching/po906_week567.pdf.

Tukey, J.W. (1977). *Exploratory Data Analysis*. Addison-Wesley series in behavioral science. Addison-Wesley Publishing Company. ISBN: 9780201076165. URL: https://books.google.nl/books?id=UT9dAAAAIAAJ.

Welch, B. L. (1951). "On the Comparison of Several Mean Values: An Alternative Approach". In: *Biometrika* 38.3/4, pp. 330–336. ISSN: 00063444. URL: http://www.jstor.org/stable/2332579.

Wilson, H. James, Narendra Mulani, and Allan Alter (2017). "Sales Gets a Machine-Learning Makeover". In: *MIT Sloan Management Review*. URL: https://sloanreview.mit.edu/article/sales-gets-a-machine-learning-makeover.

Yunus, Muhammad (2018). *By 2050, machine learning and AI will outsmart humans*. URL: https://economictimes.indiatimes.com/magazines/panache/

by-2050-machine-learning-and-ai-will-outsmart-humans-nobel-laureate-muhammad-yunus/articleshow/62616673.cms.

ZeBlemoyer, Luke (2012). "Bias / Variance Tradeoff". In: *CSE546: Linear Regression - University of Washington*. URL: https://courses.cs.washington.edu/courses/cse546/12wi/slides/cse546wi12LinearRegression.pdf.

Zhang, G.Peter (2003). "Time series forecasting using a hybrid ARIMA and neural network model". In: *Neurocomputing* 50, pp. 159 –175. ISSN: 0925-2312. DOI: https://doi.org/10.1016/S0925-2312(01)00702-0. URL: http://www.sciencedirect.com/science/article/pii/S0925231201007020.