DEGREE PROJECT IN MATHEMATICS,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2017

Comparing fast- and slow-acting features for short-term price predictions

ERIK PÄRLSTRAND

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ENGINEERING SCIENCES

Comparing fast- and slow-acting features for short-term price predictions

ERIK PÄRLSTRAND

Degree Projects in Mathematical Statistics (30 ECTS credits)
Degree Programme in Applied and Computational Mathematics (120 credits)
KTH Royal Institute of Technology year 2017
Supervisor at Lynx Asset Management: Per Hallberg and Martin Rehm
Supervisor at KTH: Timo Koski
Examiner at KTH: Timo Koski

TRITA-MAT-E 2017:24
ISRN-KTH/MAT/E--17/24--SE

Royal Institute of Technology
School of Engineering Sciences
KTH SCI
SE-100 44 Stockholm, Sweden
URL: www.kth.se/sci

Comparing fast- and slow-acting features for short-term price predictions

Abstract

This thesis compares two groups of features for short-term price predictions of futures contracts; fast- and slow-acting features. The fast-acting group are based on limit order book derived features and technical indicators that reacts to changes in price quickly. The slow-acting features constitute of technical indicators that reacts to changes in price slowly.

The comparison is done through two methods, group importance and a mean cost calculation. This is evaluated for different forecast horizons and contracts. Furthermore, two years of data was provided to do the analysis. Moreover, the comparison is modelled with an ensemble method called random forest. The response is constructed using rolling quantiles and a volume weighted price.

The finding implies that fast-acting features are superior at predicting price changes on smaller time scales, while long-acting features are better at predicting prices changes on larger time scales. Furthermore, the multivariate model results were similar to the univariate ones. However, the results are not clear-cut and more investigation ought to be done in order to confirm these results.

Jämförelse av snabba och långsamma variabler för kortsiktig prisprediktion

Sammanfattning

Den här uppsatsen jämför två typer av variabler för kortsiktig pris prediktion av terminskontrakt; snabba och långsamma variabler. De snabba variablerna är en sammansättning av limit order boks härledda variabler och tekniska indikatorer som svarar snabbt på prisförändringar. De långsamma variablerna utgörs av tekniska indikatorer som svarar långsamt på prisförändringar.

Jämförelsen är gjord med två metoder, "group importance" och genomsnittskostnad. Detta har gjort för olika prediktionshorisonter och kontrakt. Två-års data användes för att göra analysen. Detta modellerades med en gruppmetod, kallad "random forest". Responsvariabel är konstruerad med rullande kvantiler och ett volymviktat pris.

Resultaten indikerar att snabba variabler är bättre på att prediktera prisändringar på korta horisonter medan långsamma är bättre på att prediktera prisändring på långa horisonter. Dessutom efterliknade de multivariata resultaten de univariata. Dock var resultaten inte entydiga och mer undersökning krävs för att säkerställa dessa resultat.

Acknowledgements

I would first like to express my sincere gratitude to my supervisors at Lynx Asset Management, Per Hallberg and Martin Rehn for their interesting suggestions, comments and feedback, without which this thesis could not have been completed. Moreover, I would like to thank Lynx Asset Management for providing the necessary data in order to do the analysis.

I wish to express my thankfulness to my supervisor at KTH, Timo Koski for his valuable comments and feedback.

I'm also thankful for all my fellow students at KTH for these interesting study years.

Last but not least, I would like to thank my better half, Palma Toth for helping me proofread this thesis and always being there for me.

Contents

List of Figures

List of Algorithms

List of Tables

Nomenclature

Acronyms

Symbol	**Description**
ADL	Accumulation Distribution Line
ATR	Average True Range
CHV	Chaikin Volatility
CHO	Chaikin Oscillator
CPC	Conditional Price Change
DEL	Difference in Expected Level
EMA	Exponential Moving Average
FSD	Fast Stochastic D
FSK	Fast Stochastic K
MACD	Moving Average Convergence Divergence
OFI	Order Flow Imbalance
PB	Percent Bollinger
RSI	Relative Strength Index
SMA	Simple Moving Average
UO	Ultimate Oscillator
VI	Volume Imbalance
VRV	VWAP of Reference Volume

Mathematical notations

Symbol	Description
$\mathbb{Z}^+$	The positive integers (excluding zero)
$\mathbb{R}$	The real numbers
$\mathbb{P}(A)$	Probability of an event A
$\{\cdot\}$	A set of elements
$\mathcal{A} \setminus \mathcal{B}$	Difference of sets $\mathcal{A}$ and $\mathcal{B}$
$\lfloor x \rfloor$	The floor function of x
$\log(x)$	The natural logarithm of x
$\mathbb{1}(\cdot)$	The indicator function

Machine learning notations

Symbol	Description
$\mathcal{D}$	The training set of size n
$\mathcal{T}$	The test set of size m
x_i^j	The value of feature j for example i.
y_i	The i-th data point's response
$\hat{y}_i$	The i-th predicted response
$\mathcal{Y}$	The response space
$\mathcal{X}$	The feature space
p	Number of features, i.e. $\dim(\mathcal{X}) = p$
$\mathcal{F}_s$	The slow-acting features
$\mathcal{F}_f$	The fast-acting features
P	The confusion matrix
C	The cost matrix

Series notations

Symbol	**Description**
$\{Z_t\}$	A generic time series
$\{p_t\}$	The price series
$\{p_t^c\}$	The closing price of bars series
$\{p_t^o\}$	The opening price of bars series
$\{p_t^h\}$	The high price of bars series
$\{p_t^l\}$	The low price of bars series
$\{p_{t,i}^a\}$	The ask price series at level i
$\{p_{t,i}^b\}$	The bid price series at levels i
$\{V_{t,i}^a\}$	The ask volume series at level i
$\{V_{t,i}^b\}$	The bid volume series at level i

Chapter 1

Introduction

In the last two decades enormous advances have been made within communication technology, which allowed the tremendous increase in financial electronic trading. For example, in 2015, 99% of all futures contracts traded on Chicago Mercantile Exchange were done electronically [15].

All trades that are executed electronically, are recorded and therefore produce an abundance of data that can be used by machine learning algorithms to predict future prices. This increase in data and the developments in computer processing turned the machine learning application to financial data into a hot topic.

When applying learners on financial data, one builds a model that tries to predict future prices based on market information. This market information is compressed into a set of features and the underlying assumption is that the features are correlated with future prices. The most common learners in finance are supervised, i.e. one exposes the learners to a number of training examples (future prices and features), which the learners adapt to. This is called the training phase. When the learners are trained, one evaluates the models on unseen data, i.e. data that was not used for training. There are numerous machine learning algorithms, however in this thesis the random forest algorithm will be used. It is a very powerful method that has performed well on many different tasks [11].

Traditionally, computerized trading has been done on longer time horizons, such as days or even weeks. However, during the last decade more interest were shown towards short-term trading (also known as high frequency trading) [31] [12] [43]. According to the Aldridge and Krawciw estimate, in 2016 high frequency trading initiate 10-40% of trading volume in equities, and 10-15% of the trading volume in foreign exchange and commodities [3].

Applying machine learning algorithms to high frequency data, one can divide the features into two groups based on the time period they operate on, namely features that act on shorter time periods and those that act on longer time periods. For example, one can use a moving average based on one minute of data or a moving average based on thirty minutes of data.

However, there is little documented information in the scientific literature on which feature groups perform better for short-term price prediction. Furthermore, does the performance change as the prediction horizon increases/decreases? A lot of questions regarding short-term price prediction are still unanswered in the literature and therefore I choose this as my topic of this thesis.

1.1 Objectives

In this section, the main objectives of this thesis are presented. At the end of the thesis, chapter 8, each objective will be reviewed and related to the results. The objectives of this thesis are:

1. **Compare the importance for short and long time-acting features**
   Given historical time series of price and volume for different futures contracts, investigate if the importance for the short time-acting features differs from the longer ones on different forecast horizons.
2. **Compare the mean cost for the short and long time-acting features**
   Investigate how the two groups perform as one increases the punishment for making an incorrect direction prediction (i.e. predicting an increase in price, but the outcome is a decrease and vice versa) on different forecast horizons.
3. **To do the previous analysis for a multivariate model**
   To investigate the group importance and mean cost calculations using a multivariate model. Moreover, examine if the results for the multivariate model differ from the univariate ones.

1.2 Scope and Limitations

As stated in section 1.1, the main objective of this thesis is the comparison of two feature groups, namely slow-acting and fast-acting features. However, no hyper-parameter optimization was done for the random forest algorithm, only the default values were used as it is described in section 5.3. Moreover, no other machine learning algorithms were considered.

Additionally, the technical indicators' parameters were not optimized. Finally, the used contracts for the analysis were limited to four, see section 5.1 for the list of contracts, and the number of forecast horizons were restricted to six.

1.3 Previous studies

The literature on machine learning application to short-term price prediction is quite extensive. The features used here can be broadly divided into three categories: limit order book derived features, technical indicators and combinations of the two.

Alec N. Kercheval et al. applied support vector machines with features derived from the limit order book to predict the mid price movements and spread crossing 5-15 tick-events ahead [31]. Additionally, Frédéric Abergel et al. used similar features, with logistic regression, to predict the sign of the mid price at the next event [43].

German Creamer used gradient boosting with technical indicators to predict mid price movement 10-600 seconds ahead [17]. Hao Chen et al. used a double layered neural network with technical indicators to predict the mid price movement five minutes ahead [12]. Youngdoo Sona et al. used logistic regression, neural networks and support vector machines with technical indicators to predict the mid price movement one minute ahead [39].

Ash Booth used a random forest model with features derived from the limit order book and technical indicators to predict the price impact of an order [8].

1.4 Outline

This thesis has the following layout:

In chapter 2 the limit order book is presented along with a short description of futures contracts. In chapter 3 the mathematical background is covered. Some of the statistical methods that are used in this thesis are introduced along with the machine learning theory. Furthermore, time series and how they relate to financial prediction are described. In chapter 4 the used features are described. The chapter opens with the limit order book derived features and it is followed by the technical indicators.

In chapter 5 the methodology is introduced. Here, the reader finds a description on the data processing methods and how the features are built. Furthermore, the used methods in the results section are described. In chapter 6 the results are presented. Here, the objective questions are investigated, i.e. the group importance for the short- and long-time-acting groups. Lastly, the mean cost for the two groups are explored. In chapter 7 the results are discussed and related to the objective questions. Finally, in chapter 8 conclusions are drawn and some ideas of future work are discussed.

Chapter 2

Financial Background

In this chapter, some financial background is presented. It begins with the limit order book. Afterwards the different order types are described, and how one can define the price in a limit order book. Finally, forward and futures contracts are introduced.

2.1 The limit order book

A limit order book is an electronic waiting list of buy and sell orders for equities, futures or other listed derivatives at a given market place. It keeps track of all orders and their price, quantity, other market dependent information and the time of arrival. Therefore, a limit order book contains at a given time a list of all the possible transactions that a market participant could perform on that market.

A market, where buyers and sellers meet via a limit order book is called order-driven market. In an order-driven market buy and sell orders are matched as they arrive over time and are subjected to some priority rules. The priority is always based on the price and secondly, in most markets according to time with the first in, first out rule [1].

Figure 2.1 are a visualization of the limit order book at a given time.

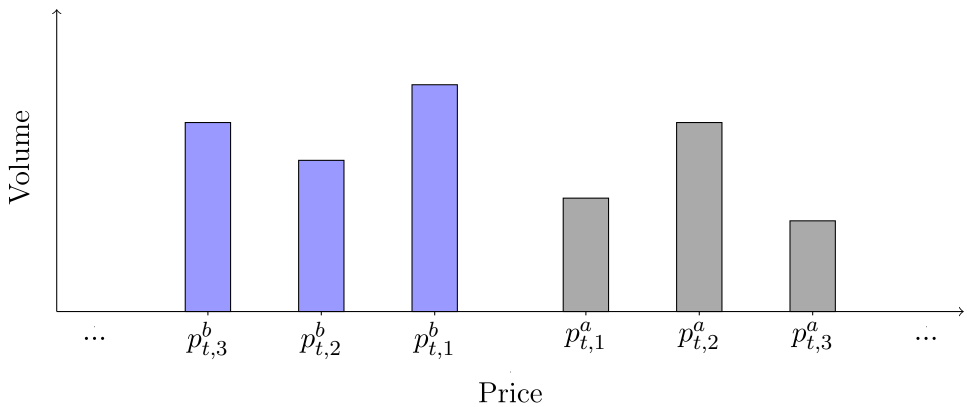


Figure 2.1: Visualization of a limit order book at time t. The blue regions represents the bid side and the grey regions represents the ask side.

2.1.1 Order types

There are three basic order types: market order entry, limit order entry and limit order cancellation.

Market order entries will be executed immediately at the best available price for a given quantity, therefore the market orders demand liquidity and have more uncertainty risks in terms of price.

Limit order entries will buy/sell a quantity from an asset for a specified limit price that the order cannot surpass. When the exchange receives a limit order the matching engine will compare the order's price and quantity with opposing orders from the book. If there is a resting order matching the incoming orders' price, a trade will occur.

Finally, a cancellation order is a type of order that cancels an existing limit order.

One should note, that most exchanges offer several types of orders, not only these, for instance orders with limited lifetime or orders wih partially visible volume [1]. However, these order-types will not be covered in this thesis.

2.1.2 The price in a limit order book

In a limit order book the definition of the price is not obvious. One way to define the price is to weight the bid and ask price with the volumes:

$$p_t = p^b_{t,1} + (p^a_{t,1} - p^b_{t,1}) \cdot \frac{V^b_{t,1}}{V^b_{t,1} + V^a_{t,1}} \tag{2.1.1}$$

where $p^b_{t,1}$ is the bid price on the first level at time t, $p^a_{t,1}$ is the ask price on the first level at time t, $V^b_{t,1}$ and $V^a_{t,1}$ is their respective volumes. The reason behind this definition is to get a more accurate description on the demands for the bid and ask side. Furthermore, one notices that if $V^b_{t,1} \gg V^a_{t,1}$ then $p_t \approx p^a_{t,1}$, i.e. since the volume on the bid side is much larger than on the ask side, the ask price is more likely the price, and vice versa. In this thesis the actual price is defined as above.

Moreover, mid price is a commonly used price definition, described in Appendix A.1.

2.2 Forward and futures

2.2.1 Forward contracts

A forward contract is a customized contract between two parties to buy/sell an asset at a specified future time at a price agreed upon today making it a derivative instrument. Forward contracts are not being traded on a centralized exchange (due to its customized nature) and are regarded as over-the-counter instruments.

The party that is buying the underlying asset in the future is said to have a long position and the party agreeing to sell the asset in the future has a short position [26].

2.2.2 Futures contracts

A futures contract can be viewed as a standardized forward contract and therefore it can easily be traded between parties including others than the two original ones. The parties initially agree to buy/sell an asset for a price agreed upon today, however unlike forward contracts the futures are marked-to-market daily, i.e. daily changes are settled day by day until the end of the contract.

These contracts are traded at exchanges and cleared at a clearing house. This addresses one of the problems with forward contracts since the clearing houses guarantee the transactions and therefore drastically reduces the counterpart risk. On the other hand, forward contracts are private agreements between two parties and therefore there is always a chance that one party may default on the agreement [26].

2.2.3 Futures contracts for speculation

Futures contracts are usually used for speculation. If a market participant expects a price increments of an underlying asset in the future he/she could potentially gain a profit by purchasing the asset in a futures contract and selling it later at a higher price on the spot market and profiting from the favorable price difference through cash settlement. Alternatively, the speculator could sell the futures which should now trade at a higher price.

Also, futures are desirable in speculation since they can go both long and short when one expects the price of the underlying asset to be higher or lower.

2.2.4 Futures contracts rollover

Since all futures contracts have expiration dates, investors who want to hold their exposures for a longer time period than the maturity date, need to close the contracts before they expiration and open new ones with later expiration date to avoid the obligations associated with settlement of the contracts. This is called rollover of futures contracts and makes it possible to produce a continuous time series for the futures price (see section 3.4 for the definition of a time series).

Chapter 3

Mathematical Background

In this chapter, the mathematical background is presented. It begins with some statistical methods that are used later in this thesis. Afterwards, decision trees and ensemble learning are presented. Lastly, time series and how they relate to financial prediction is described.

3.1 Statistical Theory

3.1.1 Quantiles

Quantiles are cut-points that divide observations from a sample into q-contiguous intervals with roughly equal-size. There is one less quantile than the number of groups created. For example, tertiles are the two cut-points that divide a sample into three roughly equal-sized groups. Figure 3.1 is an example of quantiles that divides the samples into three parts:

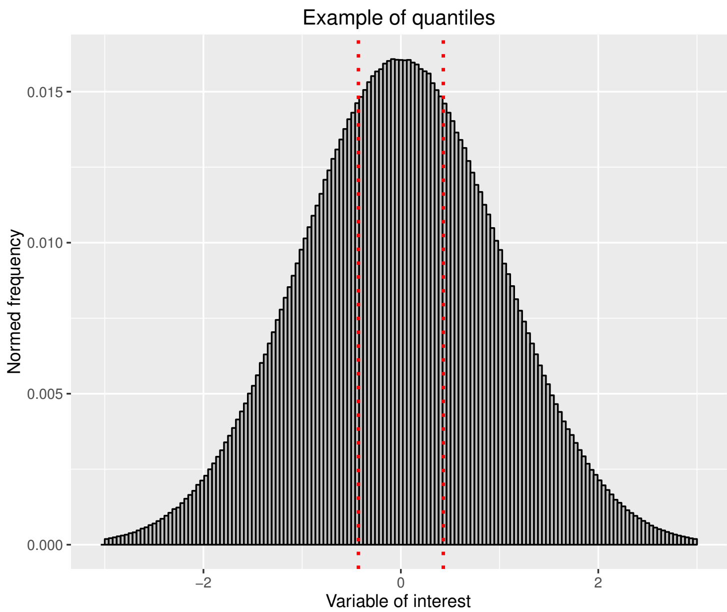


Figure 3.1: Example of quantiles (the red dotted lines) that has been estimated from a sample.

When the samples are drawn from an unknown population, the cumulative distribution function and quantile function of the underlying population are not known and the task is to estimate these quantiles. In essence, the task is to compute Q_r, the estimate for the k-th q-quantile, where $r = k/q$, $k = 1, ..., q-1$. This is done from a sample of size n' by computing a real valued index h. If h is an integer, the h-th smallest data point z_h is the quantile estimate. Otherwise an interpolation scheme is used to compute the quantile estimate from h [27]. Assuming that $\{z\}_{i=1}^{n'}$ is the data set, one common interpolation scheme is:

$$h = (n' - 1)r + 1$$

$$Q_r = z_{\lfloor h \rfloor} + (h - \lfloor h \rfloor)(z_{\lfloor h \rfloor + 1} - z_{\lfloor h \rfloor}) \tag{3.1.1}$$

3.1.2 Bias variance trade-off

The bias variance trade-off is characterized by trying to minimize two sources of errors, bias and variance, that prevent a supervised learner to generalize beyond its training set [40].

The bias in the model comes from erroneous assumptions in the learner. Large bias can arise when relevant relationships between the features and the response are missed i.e. underfitting; the model is not complex enough to fit the data. Underfitted models do not perform well on the training, nor the test set.

The variance in the model arises from sensitivity to small fluctuations in the training set. High variance can be caused by overfitting; modeling the random noise in the training set rather than the intended response and therefore usually generalize poorly outside of the training set.

Figure 3.2 is an illustration of the training and test set errors as the model complexity increases:

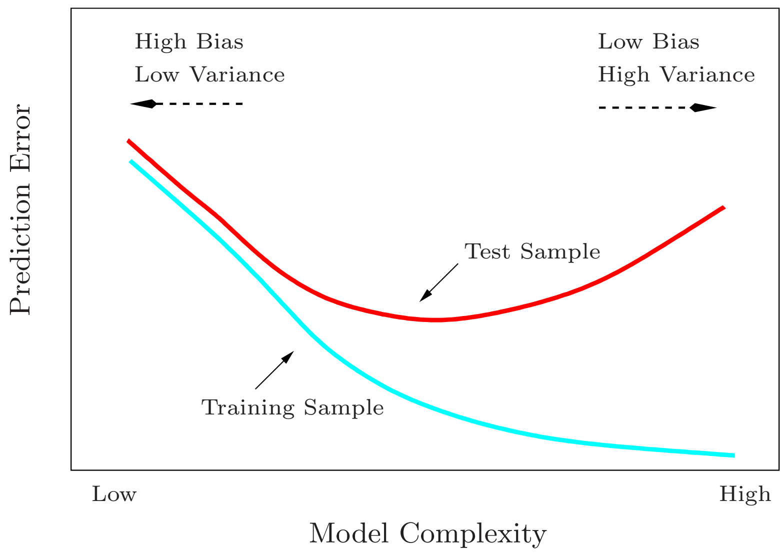


Figure 3.2: A visualization of the bias variance trade-off [25].

From figure 3.2 one can see that high bias, low variance and low bias, high variance classifiers has large out of sample error. However, by balancing the variance and bias, out of sample performance can be optimized.

3.1.3 Confusion matrix

In classification problems it is important to be able to visualize the performance of classification methods. This can be done with a confusion matrix. The rows of the matrix represent true classes and the columns represent the predicted classes. The confusion matrix will be denoted as P in this thesis. The table below is an example of a confusion matrix for a classification problem with three classes $(-1, 0, 1)$:

		Predicted $\hat{y}$		
		1	0	-1
Actual y	1	99	44	21
	0	33	105	24
	-1	11	33	87

Table 3.1: Example of a confusion matrix P. The columns represents the predicted classes and on rows represents the actual classes.

Examining the confusion matrix above (table 3.1) yields that there are $99 + 44 + 21 = 164$ examples of class one and $99 + 33 + 11 = 143$ instances of predicted class one. This is a useful visualization tool of the performance of a classifier, since it shows which classes are being predicted [25].

3.1.4 Cost matrix

Often, the performance measure of classification learners are based on the accuracy. While this is an useful performance metric, it may be important to take additional factors into account, some types of misclassifications may be worse than others. For example, rejecting a valid credit card transaction may cause an inconvenience, but approving a large fraudulent transaction may have substantial negative consequences. In situations such as this, it is important to take the cost of every type of misclassification into account to avoid the costliest errors [19].

To represent the differing cost of each type of classification, a cost matrix can be used. It will be denoted as C in this thesis. The matrix entry C_{ij} is the cost of predicting the i-th class when the j-th class is actually correct. In general, $C_{ij} > C_{jj}$ when $i \neq j$, i.e. a correct prediction is less costly than an incorrect prediction. Often the entries C_{jj} along the main diagonal will be zero. The table below is an example of a cost matrix for a classification problem with three classes $(-1, 0, 1)$:

		Predicted $\hat{y}$		
		1	0	−1
Actual y	1	0	1	2
	0	1	0	1
	−1	2	1	0

Table 3.2: An example of a cost matrix C. The columns represents the predicted classes and on rows represents the actual classes.

For the example in table 3.2, predicting −1 when the true response is 1 is twice as costly as predicting 0. Now, given a confusion matrix P and a cost matrix C, the out of sample mean cost is given by:

$$\hat{c} = \frac{1}{m} \sum_{i,j} P_{ij} C_{ij}$$

3.1.5 Feature scaling

Usually the value range of the features varies extensively. This can cause problems for some machine learning algorithms. For example KNN (K-nearest neighbors) calculates the Euclidean distance between data points. If one of the features has a broad range of values, the distance between data points will be governed by this particular features and that is not desirable. This can be solved by normalizing the features.

One common scaling method is the standardization. Feature standardization makes the values have zero mean and unit variance [40]. Let X_j be a feature of interest with the corresponding outcomes $\boldsymbol{x}^j$ with n data points. Now, the standardized feature $\tilde{\boldsymbol{x}}$ is defined as:

$$\tilde{x}_i^j = \frac{x_i^j - \bar{x}^j}{s^j} \tag{3.1.2}$$

where $\bar{x}^j = \frac{1}{n} \sum_{i=1}^{n} x_i^j$ and $s^j = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} \left(x_i^j - \bar{x}^j\right)^2}$.

3.2 Classification trees

Let us consider a supervised classification problem. The aim is to derive a prediction $f(\boldsymbol{X})$ for the response Y, with both $f(\boldsymbol{X})$ and Y taking values in $\mathcal{Y}$. Here, the features $\boldsymbol{X} = \big(X_1, ..., X_p\big)$ comprise of p-dimensional random variable from the feature space $\mathcal{X}$ and Y is a scalar random variable from the response space $\mathcal{Y}$.

The paired outcomes of $\boldsymbol{X}$ and Y, $(\boldsymbol{x}, y)$ are called data points. The function f is estimated from the training set: $\mathcal{D} = \{(\boldsymbol{x}_1, y_1), ..., (\boldsymbol{x}_n, y_n)\}$.

In this thesis, f will be approximated by tree based models (classification trees). Therefore, let us define a tree properly [6]:

Definition 3.2.1. A tree is a graph $\mathcal{G}$ in which any two nodes are connected by exactly one path.

Definition 3.2.2. A rooted tree is a tree in which one of the nodes has been assigned as the root. Furthermore, it is assumed in this thesis that the rooted tree is a directed graph, i.e. all edges are directed away from the root.

Definition 3.2.3. If there is an edge from η_1 to η_2 then node η_1 is said to be the parent of node η_2 while node η_2 is said to be a child of node η_1.

Definition 3.2.4. In a rooted tree, a node is said to be internal if it has one or more children and terminal if it has no children.

Definition 3.2.5. A binary tree is a rooted tree where all internal nodes have exactly two children.

The trees in this thesis will be rooted binary trees. Figure 3.3 is an example of a rooted binary tree:

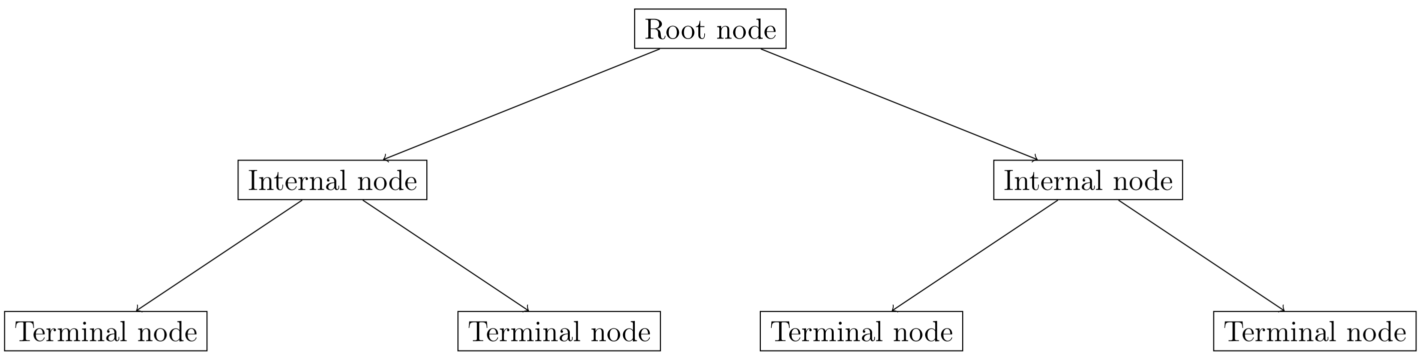


Figure 3.3: Example of a rooted binary tree structure.

A classification tree can be defined as a model $f : \mathcal{X} \to \mathcal{Y}$ represented by a rooted tree, where any node η represents a subspace $\mathcal{X}_\eta \subseteq \mathcal{X}$ of the feature space, with the root node η_0

corresponding to $\mathcal{X}$ itself. Internal nodes η are labeled with a split s_η, which divides the space $\mathcal{X}_\eta$ that node η represents into disjoint subspaces.

Learning a classification tree ideally amounts to determine the tree structure producing the partition which is closest to the partition caused by $\mathcal{Y}$ over $\mathcal{X}$. Since the partitions are unknown, the construction of a classification tree is driven with the objective of finding a model which partitions the training set $\mathcal{D}$ as well as possible.

Let $\mathcal{H}$ be the hypothesis space. Among all classification trees $f \in \mathcal{H}$, there may exist several of them that explain $\mathcal{D}$ equally well. Following Occam's Razor principles of preferring the explanation which makes as few assumptions as possible, i.e. to favor the simplest solution which fits the data set $\mathcal{D}$. While this assumption makes sense from a generalization point of view, it also makes sense regarding interpretability. A classification tree which is small is easier to understand than a large and complex tree [25].

Let us define an impurity measure $i(\eta)$ which evaluates the goodness of any node η. Let us assume that the smaller $i(\eta)$, the purer the node and the better the predictions $\hat{y}_i$ is, for all $\boldsymbol{x}_i \in \mathcal{D}_\eta$, where $\mathcal{D}_\eta$ is the subset of training samples such that $\boldsymbol{x}_i \in \mathcal{X}_\eta$, where $\{\boldsymbol{x}_i, y_i\} \in \mathcal{D}$.

Starting from a single node representing the whole training set $\mathcal{D}$, near-optimal decision trees can then be grown greedily by iteratively dividing nodes into purer nodes. That is, by iteratively dividing $\mathcal{D}$ into smaller subsets, until a stopping criteria is meet, for example the impurity decrease is not large enough in a given node. The greedy assumption is to divide each node η using the split s that locally maximizes the decrease of impurity of the resulting child nodes. Formally, the decrease of impurity of a binary split s is defined as follows [25]:

Definition 3.2.6. The impurity decrease of a binary split s dividing node η in a left node η_L and a right node η_R is:

$$\Delta i(s, \eta) = i(\eta) - \frac{\bar{\eta}_L}{\bar{\eta}} i(\eta_L) - \frac{\bar{\eta}_R}{\bar{\eta}} i(\eta_R)$$

where $\bar{\eta}_* = |\mathcal{D}_{\eta_*}|$, where $*$ indicates the node index.

In this thesis the Gini impurity will be used as the impurity measure. The Gini impurity measures the inequality among classes of a frequency distribution. If the Gini impurity is close to one it indicates almost perfect equality (i.e. the class frequency is roughly the same). If the Gini impurity is close to zero it has maximal inequality among classes (i.e. one class has a frequency of 100% while the others have 0%). The Gini impurity is defined as:

$$i_G(\eta) = 1 - \sum_{k=1}^{K} \hat{\nu}_k^2 \tag{3.2.1}$$

where $\hat{\nu}_k^2$ is the frequency of class k amongst the population in node η and K is the total number of classes [35]. $\hat{\nu}_k$ can be calculated in node η as:

$$\hat{\nu}_k = \frac{1}{|\mathcal{D}_\eta|} \sum_{(\boldsymbol{x}_i, y_i) \in \mathcal{D}_\eta} \mathbb{1}\big(y_i = k\big) \tag{3.2.2}$$

In order to build the tree one has to be able to determine where the splits should be made. The splits used in this thesis are binary splits defined on a single feature X_j and results in non-empty subsets, i.e. $\mathcal{D}_{\eta_L} \neq \emptyset \wedge \mathcal{D}_{\eta_R} \neq \emptyset$ [9].

Now, if X_j is a categorical feature then one can simply compare the different possible splits (assuming p is of reasonable size). For example, assume that the feature X_j is a discrete feature with three outcomes, $\{-1, 0, 1\}$. Then the splits to consider would be $X_j \geq 0$ and $X_j \geq 1$. Now, one would choose the split that yields the highest Gini gain according to equation (3.2.1).

However, determining the split for a continuous feature X_j is a bit harder. The most common approach for determining the split for a continuous feature is to group the data set into bins. The following algorithm can be used to find the best splitting point on a continuous feature X_j [13]:

Algorithm 1: Finding the best split s_j on feature X_j in node η.

Assume that $\{x_i^j\} \in \mathcal{D}_\eta$
Let Q be the number of bins.
Create Q bins from $\{x_i^j\}$, $\mathcal{B}_1, ..., \mathcal{B}_Q$
for $q = 1, ..., Q-1$ **do**
    Create the split s_j^q according to the largest value in bin q, i.e. $s_j^q = \max\left(x_i^j | x_i^j \in \mathcal{B}_q\right)$
    Compute the Gini gain for the split s_j^q, i.e. $\Delta i_G(s_j^q, \eta)$
end
Choose the split s_j that maximizes the Gini gain, i.e. $s_j = \arg\max_q \Delta i_G(s_j^q, \eta)$

Algorithm 1 can also be used for categorical features when p is large, with some modifications.

Now, the following algorithm can be employed to determine the best split amongst all features:

Algorithm 2: Finding the best split s_η in node η.

$\Delta = -\infty$
for $j = 1, ..., p$ **do**
    Find the best binary split s_j defined on X_j according to algorithm 1
    if $\Delta i_G(s_j, \eta) > \Delta$ **then**
        $\Delta = \Delta i_G(s_j, \eta)$
        $s_\eta = s_j$
    end
end

Now, equipped with the methods described earlier, one can finally define the process of building a classification tree:

Algorithm 3: The algorithm for building a classification tree T_b.

for *each internal node* η, *recursively* **do**
    Find the best split s_η according to algorithm 2
    Split the node η into two child nodes (η_L, η_R) according to s_η
    Partition the data $\mathcal{D}_\eta$ according to s_η, i.e.
    $\mathcal{D}_{\eta_L} = \{\boldsymbol{x_i}, y_i\} \big| \{\boldsymbol{x}_i, y_i\} \in \mathcal{D}_\eta \wedge x_i^j \leq s_\eta$
    $\mathcal{D}_{\eta_R} = \mathcal{D}_\eta \setminus \mathcal{D}_{\eta_L}$
end

Figure 3.4 is an example of a classification tree which partition the feature space according to a tree model into four disjoint regions:

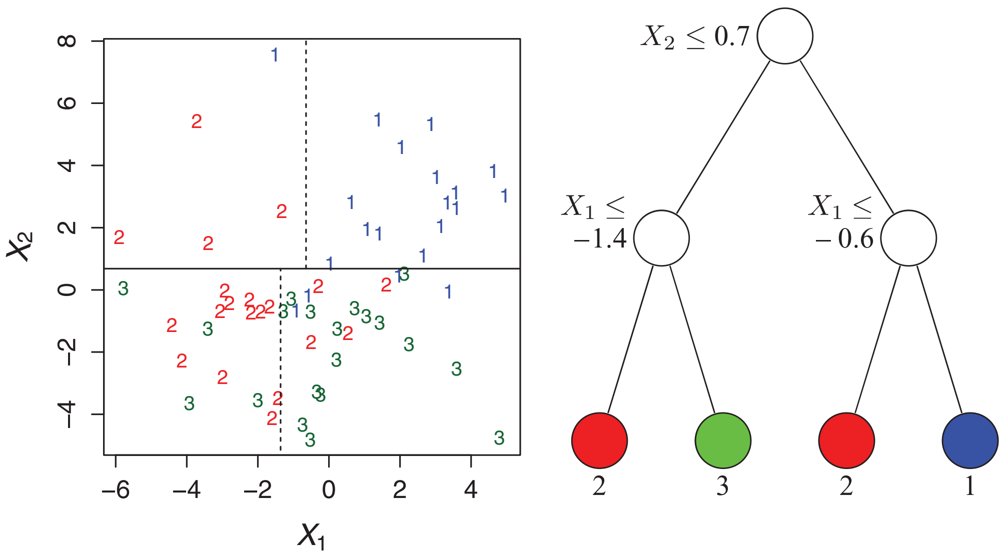


Figure 3.4: An example of a classification tree and the corresponding feature space partitions [34].

Assuming that the classification tree is built, one can predict a new point $\boldsymbol{x}_*$ by:

Algorithm 4: Predicting $\hat{y}_*$ from a classification tree.

$\eta = \eta_0$
while η *is not a terminal node* **do**
    $\eta =$ the child node η' of η such that $\boldsymbol{x}_* \in \mathcal{X}_{\eta'}$
end
$\hat{y}_* = \arg\max_k \sum_{(\boldsymbol{x}_i, y_i) \in \mathcal{D}_\eta} \mathbb{1}(y_i = k)$

Unfortunately, classification trees tend to create over-complex trees that do not generalize well out of sample (i.e. overfitting). In terms of the variance-bias trade off, they usually produce low bias, high variance classifiers [25]. This can be solved with an ensemble learner.

3.3 Ensemble learning

An ensemble learner contains a set of learners which are usually called base learners. The generalization ability of an ensemble is usually much larger than the base learners. Actually, ensemble learning is appealing because they are able to boost weak learners which are just slightly better than random guesses to strong learners which can make very accurate predictions.

The base learners f_i are generated from a training set by a base learning algorithm which can be decision trees [40]. Bagging of trees and random forest are two examples of ensemble methods based on decision trees.

Typically, an ensemble is constructed in two steps. First, a number of base learners are produced, which can be generated in a parallel style or in a sequential style where the generation of base learners has influence on the generation of subsequent learners. Then, the base learners are combined, usually by majority voting for classification:

$$\hat{F}(\boldsymbol{x}) = \arg\max_{k} \sum_{i} \mathbb{1}\big(\hat{f}_i(\boldsymbol{x}) = k\big) \tag{3.3.1}$$

To understand why the generalization ability of an ensemble is usually much stronger than that of a single learner, [18] give three reasons by viewing the nature of machine learning as searching a hypothesis space $\mathcal{H}$ for the most accurate hypothesis.

The first reason is that the training set might not provide sufficient information for choosing a single best learner. For example, there may be many learners that perform equally well on the training set. Thus, combining these learners may be a better choice.

The second reason is that the search processes of the learning algorithms might be imperfect. For example, even if there exists a unique best hypothesis, it might be difficult to find it since running the algorithms results in sub-optimal hypotheses. Thus, ensembles can compensate for such imperfect search processes.

The third reason is that, the hypothesis space being searched might not contain the true target function, while ensembles can give a good approximation. For example, it is well-known that the classification boundaries of decision trees are linear segments parallel to coordinate axes. If the target classification boundary is a diagonal line, using a single decision tree cannot lead to a good result. A good approximation can be achieved by combining a set of decision trees [18].

Figure 3.5 is an example of an ensemble learner with three base learners.

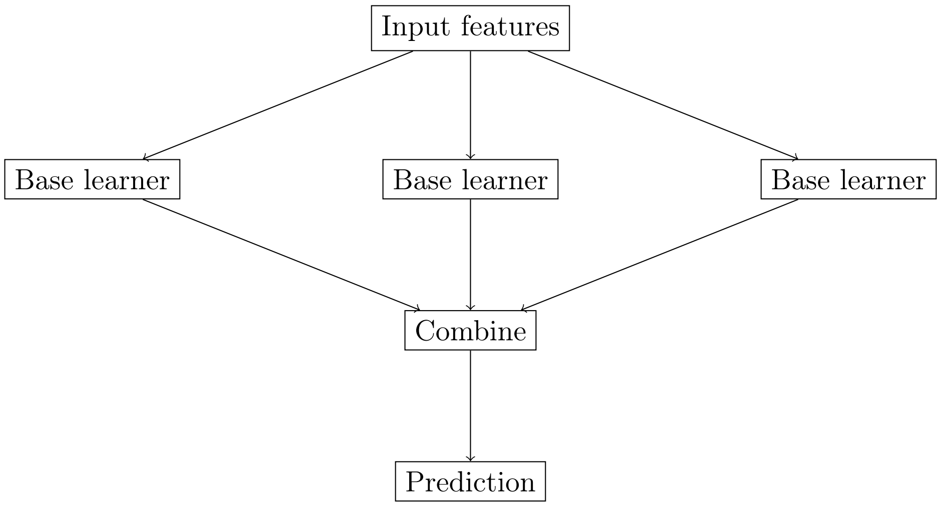


Figure 3.5: Example of an ensemble learner with three base learners.

3.3.1 Bagging

Bagging (also known as bootstrap aggregating) is an ensemble learner designed to lower the variance and thereby increase prediction accuracy by combining multiple classification trees.

Given a training set $\mathcal{D}$ of size n, the bagging method generates B new training sets $\{\mathcal{D}'_b\}_{b=1}^{B}$ with size n' by sampling from $\mathcal{D}$ through bootstrap cases. These new training sets $\mathcal{D}'_b$ are being used to train the new models $\hat{f}_b$ and combined by majority vote to produce a prediction:

$$\hat{F}_{bag}(\boldsymbol{x}) = \arg\max_{k} \sum_{b=1}^{B} \mathbb{1}\big(\hat{f}_b(\boldsymbol{x}) = k\big) \tag{3.3.2}$$

A critical factor to whether bagging will improve accuracy is the stability of the procedure for constructing $\hat{f}$, i.e. the base learner. If changes in $\mathcal{D}$, i.e. a replicate $\mathcal{D}$, produces small changes in $\hat{f}$ then $\hat{F}$ will be close to $\hat{f}$. Improvement will occur for unstable procedures where a small change in $\mathcal{D}$ can result in large changes in $\hat{f}$. Therefore, bagging unstable classifiers usually improves them. Bagging stable classifiers is not a good idea.

In terms of bias variance trade-off, the bagging procedure leads to a decrease in variance and a small increase in bias. The prediction of a single tree is highly sensitive to noise in the training set. The average of many trees are not sensitive to noise, as long as the tress are not correlated. Training multiple trees on the same data set would lead to highly correlated trees, and choosing bootstrap samples is a way to de-correlate the tress by showing them different training sets [40].

The number of trees B is a free parameter that is chosen by the user. The algorithm for bagging follows:

Algorithm 5: The bagging algorithm.

for $b = 1$ **to** B **do**
  Select n' observations from $\mathcal{D}$ with replacement to form the bootstrap sample $\mathcal{D}'_b$
  Grow a classification tree T_b on the bootstrap sample $\mathcal{D}'_b$ according to algorithm 3, which represent the function $\hat{f}_b$
end
Majority vote the output from all B trees, i.e. $\hat{F}_{\text{bag}}(\boldsymbol{x}) = \arg\max_k \sum_{b=1}^{B} \mathbb{1}\big(\hat{f}_b(\boldsymbol{x}) = k\big)$

3.3.2 Random forest

Random forest is an extension of the bagging algorithm. It uses the same bootstrap procedure as bagging but, at each split, a random subset of features with size $p_{try} \leq p$ is selected to build the trees. The reason for doing this is to de-correlate the trees even more. When one uses ordinary bootstrap samples, some of the features can be strong predictors for the response and they will be selected in many of the B trees which will cause them to become correlated. By choosing a random subset one might break up this dependency and make them more de-correlated.

Usually, in a classification setting with p features, one chooses $p_{try} = \lfloor \sqrt{p} \rfloor$ [25]. However, this parameter can be determined by the user according to the problem specification. Furthermore, random forest have more hyper-parameters that can be tuned, for example, number of trees that are grown B, depth of the tree and more.

The algorithm for random forest follows:

Algorithm 6: The random forest algorithm.

for $b = 1$ **to** B **do**
  Select n' observations from $\mathcal{D}$ with replacement to form the bootstrap sample $\mathcal{D}'_b$
  Grow a classification tree T_b on the bootstrap sample by recursively
  for *each internal node* η **do**
    Randomly select p_{try} features
    Select the best split s_η among the p_{try} features according to algorithm 2 in node η
    Split the node η into two child nodes according to the best split s_η
  end
  This tree T_b represent the function $\hat{f}_b$
end
Majority vote the output from all B trees, i.e. $\hat{F}_{\text{rf}}(\boldsymbol{x}) = \arg\max_k \sum_{b=1}^{B} \mathbb{1}\big(\hat{f}_b(\boldsymbol{x}) = k\big)$

3.3.3 Feature importance for random forest

Usually, only a few of the features substantial influence on the response in a machine learning setting. The vast majority of features are irrelevant and could be eliminated. Therefore, it is often useful to learn the importance of each feature.

Now, introduce a random variable X'_j, which is an independent replication of X_j and it is also independent of the response and all other features. According to [28] the feature importance for the feature X_j can be defined by:

$$\mathcal{I}_j = \mathbb{P}\big(Y \neq f(X_1, ..., X'_j, ..., X_p)\big) - \mathbb{P}\big(Y \neq f(X_1, ..., X_j, ..., X_p)\big) \tag{3.3.3}$$

The variable X_j is called relevant if $\mathcal{I}_j > 0$, i.e. the probability of an incorrect prediction is larger when X'_j is used rather than X_j. It is important to note that this relevant feature definition also includes those variables that do not have their own effects on the response, but they are associated with the response due to their correlation with influential features.

In order to estimate $\mathcal{I}_j$ one tries to mimic the independent replication X'_j by permuting the j-th feature, i.e. the j-th feature' data points are randomly mixed (however still retaining the distribution of the feature values since it is just a permutation). The predictions and their respective error rates are obtained from the modified and original data set that are used in the estimation of equation (3.3.3). Therefore, the estimation of $\mathcal{I}_j$ for the tree T_b is given by:

$$\hat{\mathcal{I}}_j(T_b) = \frac{1}{m(T_b)} \sum_{i=1}^{m(T_b)} \Big(\mathbb{1}\big(y_i \neq \hat{y}'_i\big) - \mathbb{1}\big(y_i \neq \hat{y}_i\big) \Big) \tag{3.3.4}$$

where $\hat{y}'_i$ is the predicted response when the j-th feature' data points are permuted and $\hat{y}_i$ is when it is not. Here, the predictions are for the out of sample data, i.e. data that was not used to train the model and $m(T_b)$ is the size of the out of sample data for tree T_b [28].

This importance measure can easily be generalized for ensemble methods with a decision tree base learner by simply averaging over all trees:

$$\hat{\mathcal{I}}_j = \frac{1}{B} \sum_{b=1}^{B} \hat{\mathcal{I}}_j(T_b) \tag{3.3.5}$$

Since averaging gives a stabilizing effect, the ensemble measure is more reliable than a single tree measure.

3.3.4 Group importance for random forest

The feature importance in section 3.3.3 can be extended to a group of features. Now, assume that $J = \{j_1, ..., j_k\}$ is a k-tuple of indices and $k \leq p$. Let us introduce a random vector $\boldsymbol{X'}_J$, which is an independent replication of $\boldsymbol{X}_J$ and it is also independent of the response and all other features. The importance for group J is defined by:

$$\mathcal{I}_J = \mathbb{P}\big(Y \neq f(\boldsymbol{X'}_J, \boldsymbol{X}_{\bar{J}})\big) - \mathbb{P}\big(Y \neq f(\boldsymbol{X}_J, \boldsymbol{X}_{\bar{J}})\big) \tag{3.3.6}$$

where $\boldsymbol{X}_{\bar{J}} = \boldsymbol{X} \setminus \boldsymbol{X}_J$. In order to estimate $\mathcal{I}_J$ one tries to mimic the independent replication $\boldsymbol{X'}_J$ by permuting the J features' data points, independently [24] [23]. The predictions and their respective error rates are obtained from the modified and original data set that are used in the estimation of equation (3.3.6). Therefore, the estimation of $\mathcal{I}_J$ for the tree T_b is given by:

$$\hat{\mathcal{I}}_J(T_b) = \frac{1}{m(T_b)} \sum_{i=1}^{m(T_b)} \Big(\mathbb{1}\big(y_i \neq \hat{y}'_i\big) - \mathbb{1}\big(y_i \neq \hat{y}_i\big) \Big) \tag{3.3.7}$$

where $\hat{y}'_i$ is the predicted response when the J features' data points are permuted and $\hat{y}_i$ is when it is not [28]. As in section 3.3.3, it can be generalized for ensemble methods with a decision tree base learner by simply averaging over all trees:

$$\hat{\mathcal{I}}_J = \frac{1}{B} \sum_{b=1}^{B} \hat{\mathcal{I}}_J(T_b) \tag{3.3.8}$$

It is worth noting that, in general, $\mathcal{I}_J \neq \sum_{j \in J} \mathcal{I}_j$.

3.4 Time series

Let $\{Z_t\}$, $t = 1, ..., T$ be a discrete time series where $t = 1, ..., T$ are certain discrete time points. In most cases the time is taken at equally spaced intervals. Time series are used in signal processing, weather forecasting, but most importantly for this thesis, econometrics/mathematical finance (prices of commodities or assets produce time series) [10].

Time series analysis can be used to forecast future values. Here one uses information about historical values and related patterns to predict future values.

3.4.1 Financial time series

Since trading is done on a continuous basis one would like to create discrete time bars that contain the essential information. This is done by creating bars which is a form of aggregation of trades. These bars contain the open, close, high and low price for a set period of time. Furthermore, noise in the tick-by-tick data is removed when the trades are being aggregated.

This aggregated data can be used to build features or to estimate the volatility.

3.4.1.1 Volatility estimation

In many financial applications there is a need to estimate the historical daily volatility of an asset. A naive approach is the close-to-close volatility estimation:

$$\sigma_{t,cc} = \sqrt{\frac{F}{n_1 - 1} \sum_{i=1}^{n_1} \left(p^c_{t-i+1} - p^c_{t-i}\right)^2} \tag{3.4.1}$$

where p^c_t is the closing price at day t, n_1 is the number of data points used to estimate the volatility and F is the number of trading days in a year.

Unfortunately, the close-close estimator does not handle opening jumps well. However, Yang-Zhang extension of the Garman-Klass volatility does, denoted as the Yang-Zhang volatility in this thesis. Now, let:

$$\Lambda_t = \left(p^o_t - p^c_{t-1}\right)^2 + \frac{1}{2}\left(p^h_t - p^l_t\right)^2 - \left[2\log(2) - 1\right]\left(p^c_t - p^o_t\right)^2 \tag{3.4.2}$$

Then, the Yang-Zhang volatility is given by:

$$\sigma_{t,YZ} = \sqrt{F}\sqrt{\mathrm{EMA}(\Lambda_t, \lambda)} \tag{3.4.3}$$

where EMA(...) is defined in section 4. The Yang-Zhang volatility assumes zero drift and will therefore overestimate the volatility if an asset has a non-zero mean return [14]. It tends to create lower bias estimates of volatility than the close-close estimator. Therefore, the Yang-Zhang volatility is used as historical volatility measure in this thesis.

3.4.2 Machine learning using time series

When using machine learning methods on time series one needs to be careful not to change the time order of the training, validation and test set. For instance, assume that the price at time t and $t+1$ are strongly correlated. Now, if one would sample the data randomly and the price at $t+1$ would be in the training set and the price at t would be in the test set this correlation would "leak" information about the test set and therefore would introduce a look ahead bias in the model.

3.4.3 Properties of time series features

It is useful if the features are invariant in price and/or volume when applying classification learners on time series data. Invariance means that the feature remains unchanged when it is transformed under a certain operation. For example, if the bid volume is 10 and the ask volume is 20 then a volume imbalance feature should have the same value if the bid and ask volume were multiplied by a factor of 1000 [22].

Another useful property is that the features' range are symmetric around their centers. For example if it is centered around 0 then the feature's interval would be $[-c, c]$, $c \in \mathbb{R}$.

Furthermore, one would like the features to be anti-symmetric, with respect to the price. If one assumes that a feature has a value of 0.5 at a given time, if one would flip the limit order book and the order book volumes would reverse (bid with ask) then the feature would change its value to -0.5.

Chapter 4

Features description

In this chapter, the used features are defined. It begins with the limit order book derived features. These are included in the fast-acting group $\mathcal{F}_f$. Afterwards, the technical indicators are presented. These are included in both the fast-acting group $\mathcal{F}_f$ and the slow-acting group $\mathcal{F}_s$, however with different parameters.

4.1 Order book features

4.1.1 Order flow imbalance

Rama Cont et al. proposed that price changes could be driven by order flow imbalance [16]. Let:

$$e_i = \mathbb{1}(p^b_{i,1} \geq p^b_{i-1,1})V^b_{i,1} - \mathbb{1}(p^b_{i,1} \leq p^b_{i-1,1})V^b_{i-1,1} - \mathbb{1}(p^a_{i,1} \leq p^a_{i-1,1})V^a_{i,1} + \mathbb{1}(p^a_{i,1} \geq p^a_{i-1,1})V^a_{i-1,1}$$

Consider the bid side. If $V^b_{i,1}$ increases but $p^b_{i,1}$ remains the same, one assign $e_i = V^b_{i,1} - V^b_{i-1,1}$, representing the size that was added to the bid side. If $V^b_{i,1}$ decreases, one also assign $e_i = V^b_{i,1} - V^b_{i-1,1}$, representing the size that was removed from the bid side, regardless if it was due to a market sell order or a cancel buy order. If $p^b_{i,1}$ increases, then $e_i = V^b_{i,1}$, representing the size of a price-improving limit order. If $p^b_{i,1}$ decreases, let $e_i = V^b_{i-1,1}$, representing the size that was removed, regardless if it was due to a market order or a cancellation. The same holds for events on the ask side, but with signs reversed.

The order flow imbalance over the time interval $[t_{k-L}, t_k]$ is defined by:

$$\text{OFI}_k(L) = \sum_{i=t_{k-L}}^{t_k} e_i \tag{4.1.1}$$

4.1.2 Volume order book imbalance

Alexander Lipton et al. used an order book imbalance to describe the trade arrival dynamics of the limit order book [33]. The imbalance at time t on level i in the limit order book is defined as:

$$\mathrm{VI}_t^i = \frac{V_{t,i}^b - V_{t,i}^a}{V_{t,i}^b + V_{t,i}^a} \tag{4.1.2}$$

4.1.3 Difference in expected level

An informative feature could be the difference in expected level of volume in the limit order book between bid and ask side. Let the limit order book on bid and ask side have I visible levels (it is important to note that the limit order book typically has more levels, but these are not visible in a the public data). Now, let:

$$q_{t,i}^b = \frac{V_{t,i}^b}{\sum_{i=1}^{I} V_{t,i}^b}$$

Then the expected level on bid side is defined as:

$$\mathrm{EL}_t^b = \sum_{i=1}^{I} i \cdot q_{t,i}^b \tag{4.1.3}$$

Now, the same procedure can be applied to the ask side. Then the difference in expected level is defined as:

$$\mathrm{DEL}_t = \mathrm{EL}_t^b - \mathrm{EL}_t^a \tag{4.1.4}$$

4.1.4 Conditional probability of VWAP

Let the volume weighted price over the time period $\tau = [t_k, t_{k'}]$ be defined as:

$$p_\tau^{VWAP} = \frac{\sum_{t\in\tau} p_t^\iota \cdot V_t^\iota}{\sum_{t\in\tau} V_t^\iota} \tag{4.1.5}$$

Here p_t^ι is the traded price at time t and V_t^ι is the respective volume. Furthermore, let:

$$\Delta_\tau = p_\tau^{VWAP} - p_{\tau'}^{VWAP} \tag{4.1.6}$$

where $p_{\tau'}^{VWAP}$ means the VWAP of the previous time period before τ. Michael Rechenthin et al. showed that the probability that $\Delta_\tau > 0$ conditioned on $\Delta_{\tau'} < 0$ is informative on shorter time periods (1-5 seconds) [38]. Therefore the following feature could be used for prediction:

$$\mathrm{CPC}_\tau = \begin{cases} 1 \text{ if } & \Delta_\tau > 0 \\ 0 \text{ if } & \Delta_\tau = 0 \\ -1 \text{ if } & \Delta_\tau < 0 \end{cases} \tag{4.1.7}$$

4.1.5 VWAP spread

A common feature from the limit order book is the price spread [8] [31], i.e. the difference between best bid and ask price. However, for most liquid futures contracts this spread is almost always constant and therefore not very informative.

However, one can define the VWAP spread instead, i.e. the difference between the bid VWAP and ask VWAP [37]. Let us assume that there are I levels on the bid and ask side. Then the VWAP spread is defined as:

$$\mathrm{VS}_t = \frac{\sum_{i=1}^{I} p^a_{t,i} V^a_{t,i}}{\sum_{i=1}^{I} V^a_{t,i}} - \frac{\sum_{i=1}^{I} p^b_{t,i} V^b_{t,i}}{\sum_{i=1}^{I} V^b_{t,i}} \tag{4.1.8}$$

4.1.6 Best price versus VWAP

One informative feature could compare the best price and the price one would have to pay in order to buy α percent of a reference volume [29]. This reference volume is set as an exponential moving average of the total volume of both bid and ask side. The average is over the same time-stamp, for example at the "09:01" today, yesterday and so on. Now, let the total volume series be defined as:

$$V^{\mathrm{Tot}}_t = \sum_{i=1}^{I} \frac{V^a_{t,i} + V^b_{t,i}}{2}$$

Define the reference volume as:

$$V^{\mathrm{Ref}}_t = \alpha \cdot \mathrm{EMA}\big(V^{\mathrm{Tot}}_t, L\big)$$

where EMA(...) is defined in section 4. The bid side will now be investigated. Let $\Upsilon = \{1, 2, ...\}$ be the largest set s.t:

$$\sum_{i \in \Upsilon} V^b_{t,i} = V^{\mathrm{Ref}}_t$$

Note that if the last level in Υ has more volume then the required one (in terms of obtaining V^{Ref}_t), one only simply include the volume from the last level in Υ s.t. V^{Ref}_t is filled. Now, the difference between the best bid price and VWAP is defined as:

$$\mathrm{VRV}^b_t = p^b_{t,1} - \frac{\sum_{i \in \Upsilon} p^b_{t,i} \cdot V^b_{t,i}}{\sum_{i \in \Upsilon} V^b_{t,i}} \tag{4.1.9}$$

The same procedure can be applied to the ask side.

4.2 Technical indicators

Traders use technical indicators as tools to determine future trends in security prices. Technical indicators are not connected to the intrinsic value of securities, instead they focus on matters such as trade volume, price and volatility to predict future prices [36].

There are several different types of technical indicators, one usually puts them into the following categories:

4.2.1 Trend indicators

Since markets tend to oscillate a lot, it can be difficult to disguise trends in the market from normal oscillations and noise. Trend indicators measure the direction and strength of a trend using some form of price average to establish a baseline. When prices move above the average it can be a sign of an uptrend and when the prices fall below the average it can be a sign of a downtrend. This averaging of price can be referred to as "smoothing" since it removes oscillations and helps to identify trends. Some examples of trend indicators are moving averages and macd [21].

4.2.1.1 Simple moving average

Simple moving average (sma) can be used as a technical indicator. It is calculated as an average over the last L series point of the time series $\{Z_t\}$. It is used to smooth out short-term fluctuations and highlight longer-term trends or cycles. It is calculated as [30]:

$$\mathrm{SMA}\big(Z_t, L\big) = \frac{1}{L}\sum_{i=0}^{L-1} Z_{t-i} \tag{4.2.1}$$

A commonly used value for the length parameter is $L = 12$, when it is applied to daily series.

4.2.1.2 Exponential moving average

An exponential moving average (ema) can be viewed as an infinite simple moving average with exponential decaying weights. It is defined as:

$$\mathrm{EMA}\big(Z_t, \lambda\big) = \lambda\sum_{i=0}^{\infty}(1-\lambda)^i Z_{t-i} \tag{4.2.2}$$

where $0 < \lambda < 1$. The ema can be computed recursively by:

$$\mathrm{EMA}\big(Z_t, \lambda\big) = \lambda Z_t + (1-\lambda)\mathrm{EMA}\big(Z_{t-1}, \lambda\big) \tag{4.2.3}$$

It is practical to define the ema in terms of a length L. One can show that if $\lambda = \frac{2}{L+1}$ then the centre of gravity for the weights forming $\mathrm{EMA}\big(Z_t, L\big)$ will be the same as for those forming $\mathrm{SMA}\big(Z_t, L\big)$ [36]. In this thesis, ema will be expressed in terms of a length:

$$\mathrm{EMA}\big(Z_t, L\big) = \lambda\sum_{i=0}^{\infty}(1-\lambda)^i Z_{t-i} \tag{4.2.4}$$

$$\lambda = \frac{2}{L+1}$$

A commonly used value for the length parameter is $L = 12$, when it is applied to daily series.

4.2.1.3 Moving average convergence divergence

The moving average convergence divergence (macd) is based on the convergence and divergence of two exponential moving averages. Convergence occurs when the two exponential moving averages move towards each other and divergence occurs when the they move away from each other. It is defined as:

$$\text{MACD}\big(Z_t, L_1, L_2\big) = \text{EMA}\big(Z_t, L_1\big) - \text{EMA}\big(Z_t, L_2\big) \tag{4.2.5}$$

The shorter exponential moving average $\text{EMA}\big(Z_t, L_1\big)$ is faster and responsible for most of the macd movements while the longer $\text{EMA}\big(Z_t, L_2\big)$ is slower and less reactive to price changes in the underlying security [4]. A commonly used value for the length parameters are $L_1 = 12$ and $L_2 = 26$, when it is applied to daily series.

4.2.1.4 Bollinger bands

Bollinger bands are volatility bands positioned above and below a simple moving average, called upper/lower Bollinger bands. The bands automatically expands when the volatility increases and narrows when the volatility decreases. They are defined as:

$$\text{BBU}\big(p_t, s, L_1, L_2\big) = \text{SMA}\big(p_t, L_1\big) + s\sigma\big(p_t, L_2\big) \tag{4.2.6}$$

$$\text{BBL}\big(p_t, s, L_1, L_2\big) = \text{SMA}\big(p_t, L_1\big) - s\sigma\big(p_t, L_2\big) \tag{4.2.7}$$

$$\sigma\big(p_t, L_2\big) = \sqrt{\frac{1}{L_2 - 1}\sum_{i=0}^{L_2-1}\big(p_{t-i} - \bar{p}\big)^2}$$

$$\bar{p} = \text{SMA}\big(p_t, L_2\big)$$

One version of the Bollinger bands is the percent Bollinger (pb) that shows where the price is in relation to the bands. At the upper band pb equals 1 and at the lower band pb equals 0 [7]. It is defined as:

$$\text{PB}_t\big(p_t, s, L_1, L_2\big) = \frac{p_t - \text{BBL}\big(p_t, s, L_1, L_2\big)}{\text{BBU}\big(p_t, s, L_1, L_2\big) - \text{BBL}\big(p_t, s, L_1, L_2\big)} \tag{4.2.8}$$

A commonly used value for the length parameters are $L_1 = L_2 = 20$, when it is applied to daily series. Furthermore, one usually set $s = 2$.

4.2.2 Momentum indicators

Momentum indicators identify the speed of price movements by comparing prices over time. Some examples of momentum indicators are relative strength index, fast stochastic k and fast stochastic d [21].

It is important to note that most of these indicators require bar data since they operate on high, low and closing prices.

4.2.2.1 Relative strength index

Relative strength index (rsi) is an indicator that measures the speed and price change movements, which oscillates between 0 and 1. In the literature, rsi is considered to be overbought when it is above 0.7 and oversold when it is below 0.3 [41]. Let:

$$\Delta_t = p_t - p_{t-1}$$

$$U_t = \Delta_t \mathbb{1}\big[\Delta_t > 0\big]$$

$$D_t = \Delta_t \mathbb{1}\big[\Delta_t < 0\big]$$

Now, let the relative strength index is defined by:

$$\mathrm{RSI}_t(L) = \frac{\mathrm{SMA}\big(U_t, L\big)}{\mathrm{SMA}\big(D_t, L\big) + \mathrm{SMA}\big(U_t, L\big)} \tag{4.2.9}$$

A commonly used value for the length parameter is $L = 14$, when it is applied to daily series.

4.2.2.2 Fast stochastic K

The fast stochastic k (fsk) represents a percent measure of the last closing price in relationship to the highest and lowest price of the last L periods. When fsk is above 0.5, the closing price is in the upper half of the price range and below 0.5 when it is in the lower half of the price range. Low values (below 0.2) indicate that the price is near the lowest low for that time period. High values (above 0.8) indicate that price is near the highest high for that time period [32]. Let:

$$\mathrm{MAX}_t(L) = \max\Big[\{p_i^h | i \in [t, t-L+1] \cap \mathbb{Z}^+\}\Big]$$

$$\mathrm{MIN}_t(L) = \min\Big[\{p_i^l | i \in [t, t-L+1] \cap \mathbb{Z}^+\}\Big]$$

Then fsk is defined as:

$$\mathrm{FSK}_t(L) = \frac{p_t - \mathrm{MIN}_t(L)}{\mathrm{MAX}_t(L) - \mathrm{MIN}_t(L)} \tag{4.2.10}$$

A commonly used value for the length parameter is $L = 14$, when it is applied to daily series.

4.2.2.3 Fast stochastic D

The fast stochastic d (fsd) is a simple moving average of the fsk, that indicates trends of the fast stochastic k [32]. It is defined as:

$$\mathrm{FSD}_t\big(L_1, L_2\big) = \mathrm{SMA}\big(\mathrm{FSK}_t\big(L_1\big), L_2\big) \tag{4.2.11}$$

A commonly used value for the length parameters are $L_1 = 14$ and $L_2 = 3$, when it is applied to daily series.

4.2.2.4 Ultimate oscillator

The ultimate oscillator (uo) is a momentum oscillator that captures momentum on three different time frames [36]. Let:

$$\mathrm{BP}_t = p_t^c - \min\left(p_t^l, p_{t-1}^c\right)$$

$$\mathrm{TR}_t = \max\left(p_t^h, p_{t-1}^c\right) - \min\left(p_t^l, p_{t-1}^c\right)$$

Where bp stands for buying pressure and tr for true range. Now, let:

$$A_t(L) = \frac{\mathrm{SMA}\big(\mathrm{BP}_t, L\big)}{\mathrm{SMA}\big(\mathrm{TR}_t, L\big)}$$

One can finally define the ultimate oscillator as:

$$\mathrm{UO}_t(\boldsymbol{w}, \boldsymbol{L}) = \frac{w_1 A_t(L_1) + w_2 A_t(L_2) + w_3 A_t(L_3)}{\displaystyle\sum_{i=1}^{3} w_i} \tag{4.2.12}$$

In this thesis $\boldsymbol{w} = [4, 2, 1]^T$. A commonly used value for the length parameters are $\boldsymbol{L} = [7, 14, 28]^T$, when it is applied to daily series.

4.2.3 Volatility indicators

Volatility measures the speed of increases and decreases in price. For technical analysis, volatility indicators measure the rate of price movements and it is independent of direction. It is usually based on changes in the highest and lowest historical prices in an instrument. Two examples of volatility indicators are Chaikin volatility and average true range [21].

Like momentum indicators, these indicators operate on bar data.

4.2.3.1 Chaikin volatility

The Chaikin volatility (chv) evaluates the breadth of the range between high and low prices. It also calculates the rate of change of an exponential moving average of the difference between the high and low prices. Often, a very fast increase/decrease of this indicator is a sign of the near bottom/top of the market [2]. It is defined as:

$$\mathrm{CHV}_t(L_1, L_2) = \frac{\mathrm{EMA}(p_t^h - p_t^l, L_1)}{\mathrm{EMA}(p_{t-L_2}^h - p_{t-L_2}^l, L_1)} - 1 \tag{4.2.13}$$

A commonly used value for the length parameters are $L_1 = L_2 = 10$, when it is applied to daily series.

4.2.3.2 Average true range

Average true range (atr) is based on the true range which measures absolute price changes and therefore atr reflects volatility in absolute terms [41]. Now, let:

$$\mathrm{TR}_t = \max\left(p_t^h, p_{t-1}^c\right) - \min\left(p_t^l, p_{t-1}^c\right)$$

Then atr is defined as:

$$\mathrm{ATR}_t(L) = \mathrm{SMA}(\mathrm{TR}_t, L) \tag{4.2.14}$$

A commonly used value for the length parameter is $L = 14$, when it is applied to daily series.

4.2.4 Volume indicators

The volume represents the amount of contracts that change hands in a given time interval. Most volume indicators are based on some forms of smoothing of raw volume. When volume levels move above their averages, it can suggest an upward trend or confirmation of a trading direction. The strongest trends often occur while volume increases, an increase in trading volume can lead to large movement in price [21]. Examples of volume indicators are accumulation distribution line and Chaikin oscillator.

As for momentum indicators, these indicators operate on bar data.

4.2.4.1 Accumulation distribution line

The accumulation/distribution line (adl) is calculated by using the close location value. This indicator compares the close price with the range of prices from the same period. A positive value indicates an increase in buying pressure and a negative indicates a decrease in buying pressure [2]. Let:

$$\mathrm{CLV}_t = \frac{2p_t^c - p_t^l - p_t^h}{p_t^h - p_t^l} \tag{4.2.15}$$

Now, the adl can be calculated:

$$\mathrm{ADL}_t = \sum_{i=0}^{\infty} \mathrm{CLV}_{t-i} V_{t-i} \tag{4.2.16}$$

$$\mathrm{ADL}_0 = 0$$

4.2.4.2 Chaikin oscillator

The Chaikin oscillator (cho) is the macd of the adl and therefore has similar interpretation as a macd [36]. It is defined as:

$$\mathrm{CHO}_t\big(L_1, L_2\big) = \mathrm{MACD}\big(\mathrm{ADL}_t, L_1, L_2\big) \tag{4.2.17}$$

A commonly used value for the length parameters are $L_1 = 10$ and $L_2 = 3$, when it is applied to daily series.

Chapter 5

Methodology

In this chapter, the methodology used in the results section is presented. This chapter begins with a description of the data and how it was manipulated, then the features and their respective parameters are presented. Afterwards, the learner and its hyper-parameters are specified. Lastly, the methods for the group importance and the mean cost calculations are clarified.

5.1 Data processing

Two years of data was given to do the analysis, 2015-01-04 to 2016-12-30 for four futures contracts. The contracts are:

1. Crude oil - Futures contracts on the crude oil price. The underlying asset is a commodity.
2. GBP - Futures contracts on the British pound to U.S. dollar. The underlying asset is a foreign exchange index.
3. T-bond - Futures contracts on the U.S. treasury bond with a duration of $10 - 30$ years. The underlying asset is a fixed-income security.
4. Dow Jones index futures (mini) - futures contract that represents a portion of a standard Dow Jones index. The underlying asset is a stock market index.

5.1.1 Raw data

The data was provided by Lynx Asset Management and each contract's data consists of three parts:

1. Continuous quote data - every quote update on the first level only. Here, the data contained date, time, price and volume for both bid and ask at the top level.
2. Continuous trade data - all completed trades. Here, the data contained date, time, price and volume for the trades.
3. Discrete quote data - A depth snapshot every minute for the first ten levels. Here, the data contained date, time, price and volume for bid and ask at each level.

Since some features required limit order book depth data, one data point was constructed each minute. Furthermore, none-overlapping regions were constructed in time. This was done to avoid correlation (due to time series) in the prediction of the overlapping regions. For example, if the forecast horizons is one minute, one would get the time periods 09:00, 09:01, 09:02,... (assuming one starts at 09:00). Figure 5.1 is an example of these none-overlapping regions.

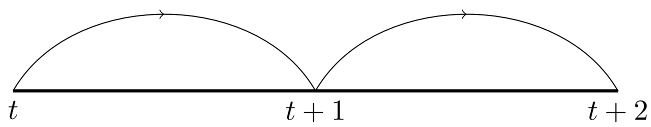


Figure 5.1: Example of none-overlapping regions.

The quote and trade data was aggregated into discrete buckets as discussed in section 3.4. Six different prediction horizons were chosen: $3, 9, 30, 90, 300, 900$ seconds. Since the depth snapshots were recorded each minute the $3, 9, 30$ seconds horizons were aligned to whole minutes, i.e. making a prediction each minute. For the 90, 300 and 900 seconds prediction horizons the time periods were 120, 300 and 900 seconds respectively.

Furthermore, 75% of the data was assigned as the training set $\mathcal{D}$ and the remaining 25% was assigned to the test set $\mathcal{T}$. The training and test set were selected in time order, i.e. the first 75% of the data set was used for training and the last 25% of the data set was used for testing, according to the theoretical arguments in section 3.4.2.

The same assignment was done for the multivariate analysis, i.e. the first 75% of the data set was used for training and the last 25% of the data set was used for testing, within each contract. Then, these four training set from each contract were combined into one training set. The same was done for the test set.

5.1.2 Response

Since the price definition in section 2.1.2 is continuous, one wishes to discretize it into three classes; down, neutral and up, denoted by $\{-1, 0, 1\}$. These classes were constructed by looking at how the price difference between t and $t+1$ falls compared to the quantiles computed from historical price differences. The quantiles are introduced in section 3.1.1, however the used quantiles were rolled, explained below.

The rolling quantiles are estimated from the price differences over a two week period (historically). This yields the two cutting points $Q_{t,1/3}$ and $Q_{t,2/3}$. The classes are defined as:

$$y_t = \begin{cases} -1 \text{ if } & Q_{t,1/3} > p_{t+1} - p_t \\ 0 \text{ if } & Q_{t,1/3} \leq p_{t+1} - p_t \leq Q_{t,2/3} \\ 1 \text{ if } & Q_{t,2/3} < p_{t+1} - p_t \end{cases} \tag{5.1.1}$$

There are a few reasons for defining the response this way. Firstly, comparison between different time horizons becomes possible. If one would define the price according to, for example mid price (take the average between best bid and ask price), the class distribution would become different between shorter forecast horizons and longer ones. For example, at 3 seconds forecast horizon, the class distribution might be: 33% up, 33% neutral and 33% down. For the longer horizon, the distribution might be 49% up, 2% neutral and 49% down. This would make the comparison between forecast horizons difficult.

Another reason for using this approach is that the machine learning part becomes less troublesome. Most learners have problems when the class distribution is skewed, since they usually try to maximize the accuracy. For some contracts, for example t-bond, the class distribution for the mid price on three seconds forecast horizon is skewed, with 15% up, 70% neutral and 15% down.

Now, the learner could only predict neural and get an acceptable accuracy, however only predicting no price changes is obviously not desirable. To account for this some alternative approaches would have to be employed, such as to under/over-sample the majority/minority class etc. However this tends to produce suboptimal results.

One should keep in mind that estimating the quantiles from historical data does not ensure equal class frequency, however it should be near equal.

5.2 Selecting and building features

In this section all the used features are presented with their respective parameters. The definition of the features can be found in chapter 4.

Since one objective of this thesis was to examine the group importance for short and long-acting features, one would like to divide the features into two groups; fast-acting and slow-acting. The shorter features parameters were chosen to be the "default" values when they are applied to daily series. As the longer acting technical indicators should span a longer time period, their parameters were increased by a factor of ten. The longer features parameters were increased substantially in order to differentiate them from the shorter ones.

The technical indicators were built upon data sampled every ten seconds. This was done since the order flow imbalance feature had a length of ten seconds in the original paper [16] and the shorter technical indicators ought be on the same time scale as the limit order book derived features.

Additionally, the technical indicators were only calculated on data from the same day, except for the vrv features. This was done in order to avoid jumps in the series. These jumps, called "overnight gaps" come from the flow of information during closed market periods.

The following two groups were created:

Features	Parameters
ema	$L = 120$
rsi	$L = 140$
fsk	$L = 140$
fsd	$L_1 = 140,\ L_2 = 30$
pb	$L_1 = 200,\ L_2 = 400,\ s = 2$
macd	$L_1 = 120,\ L_2 = 260$
chv	$L = 100$
atr	$L = 140$
cho	$L_1 = 100,\ L_2 = 30$
uo	$\boldsymbol{L} = [70, 140, 280]^T$

Table 5.1: The slow-acting features $\mathcal{F}_s$.

Features	Parameters
ema	$L = 12$
rsi	$L = 14$
fsk	$L = 14$
fsd	$L_1 = 14,\ L_2 = 3$
pb	$L_1 = 20, L_2 = 40,\ s = 2$
macd	$L_1 = 12,\ L_2 = 26$
chv	$L = 10$
atr	$L = 14$
cho	$L_1 = 10, L_2 = 3$
uo	$\boldsymbol{L} = [7, 14, 28]^T$
ofi	$L = 1$
vi	$i = 1, 2, 3$
cpc	$\tau = 1$
del	$I = 10$
vs	$I = 10$
vrv (bid & ask)	$L = 12,\ \alpha = 0.25$

Table 5.2: The fast-acting features $\mathcal{F}_f$.

Note that the vrv feature has a length parameter L that spans over L days (at the same time-stamp). However, every other parameters with a length L are built upon ten seconds interval. Also, the ema features were built from the difference between the price at time t and the opening price of that day. Furthermore, the volatility time length for the percent Bollinger feature was increased by a factor of two, in order to get a more accurate estimation of the volatility.

5.3 Choice of machine learning method

The chosen machine learning method was random forest because of the following reasons:

1. It is robust. Usually performs well even if the underlying assumptions are somewhat violated by the model [40].
2. Easy to parallelise. Since each tree is built separately, it is easy to parallelise the computation on multiple computers/cores which reduces the computational time [20].
3. Performs well. Rich Caruana et al. compared a number of supervised learning methods including support vector machines, artificial neural nets, logistic regression and random forest on different data sets. They concluded that random forest performed very well amongst these data sets. The only method that was better than random forest was gradient boosting [11].
4. Deals with multi-class problem naturally. Since one only takes a majority vote in each leaf, the random forest algorithm has a natural way of dealing with multi-class problems. In other learners, such as support vector machines, one would need to employ additional functionality to deal with multi-classes.
5. Few parameters to tune. Random forest has relatively few hyper-parameters to tune compared to other methods, such as artificial neural nets. The hyper-parameters for random forest can be found in section 3.3.2.

Random forest default parameters were chosen, i.e. $p_{try} = \lfloor \sqrt{p} \rfloor$, $n = n'$. However there is no default value for the number of trees B, but according to [5], 64-128 trees are usually enough. Since financial data are notoriously noisy, B was set to 500 for the $3, 9, 30$ seconds forecast horizons. Furthermore, it was increased by 500 trees at each following forecast horizon, i.e. 1000 trees were used for the 90 seconds forecast horizon, 1500 for the 300 seconds forecast horizon etc. This was done to compensate for the decrease in data size.

It is worth pointing out that the accuracy tends to increase as B increases. However, this accuracy increase will decay as B increases and therefore it becomes a trade-off between small increases in accuracy and computational time.

Furthermore, all continuous features were scaled to standardization, according to section 3.1.5. Random forests do not usually have any problems with differences in feature scales, unlike other methods such as artificial neural networks, however it is a good practice.

5.3.1 Computer software

The R programming language was chosen, it is a open source programming language used for statistical computation and more. Furthermore, it was decided that the ranger package for R would be used. It contains a fast implementation of the random forest algorithm and allows for distributed computation [42].

5.4 Group importance

In this section the methods and algorithms that were used to compute the group importance for $\mathcal{F}_f$ and $\mathcal{F}_s$ are presented.

The following algorithm was used to estimate the group importance for the slow- and fast-acting features:

Algorithm 7: The algorithm for computing the group importance for the slow- and fast-acting features.

for $i = 1$ **to** N **do**
    Use $\mathcal{D}$ to build a random forest according to algorithm 6
    Compute the group importance $\hat{\mathcal{I}}^i_J$, $J = \{\mathcal{F}_s, \mathcal{F}_f\}$, on the test set $\mathcal{T}$, according to equation (3.3.8)
end
Compute the mean and standard deviation for the slow and fast-acting group, $\hat{\mu}_f$, $\hat{\mu}_s$, $\hat{\sigma}_f$ and $\hat{\sigma}_s$ (see definition below)

$$\hat{\mu}_* = \tfrac{1}{N}\sum_{i=1}^{N}\hat{\mathcal{I}}^i_* \qquad \hat{\sigma}_* = \sqrt{\tfrac{1}{N-1}\sum_{i=1}^{N}\left(\hat{\mathcal{I}}^i_* - \hat{\mu}_*\right)^2}$$

The N iterations were computed with different random seed numbers. It was done in order to investigate how much the randomness effected the results.

Furthermore, one can calculate the feature importance for a single feature j. Now, compute the feature importance $\hat{\mathcal{I}}^i_j$ according to equation (3.3.5) for each iteration i. The mean feature importance is given by:

$$\hat{\mu}_j = \frac{1}{N}\sum_{i=1}^{N}\hat{\mathcal{I}}^i_j$$

5.5 Mean cost

One of the research questions was related to how a feature group's performance changes as the penalty for an incorrect prediction of direction increases. From a trading point of view, predicting an increase in price (1), but the outcome is a decrease (-1) is worse than predicting an increase in price (1), but in reality it was a neutral (0) and vice verse. Therefore, the following cost matrix was proposed:

Actual y \ Predicted $\hat{y}$	1	0	-1
1	0	1	a
0	1	0	1
-1	a	1	0

Table 5.3: The proposed cost matrix C.

Here, a dictates how much one punishes the model for making a wrong direction prediction (predicting 1 but in reality it is -1 etc). If $a = 1$ then a wrong direction prediction will have the same cost as an incorrect neutral prediction. Now, given a confusion matrix P of predictions on the test set $\mathcal{T}$ and the cost matrix C described above, the mean cost is given by:

$$\hat{c} = \frac{1}{m} \sum_{i,j} P_{ij} C_{ij}$$

The goal here is to minimize the cost. random forest was compared to a benchmark. The benchmark will only predict the zeros class (no movements) for all data points in the test set. This benchmark will have a mean cost independent of a.

First, three feature setups with random forest were compared against the benchmark for different a-values. These three setups were: use only the slow features, use only the fast features and use both the fast and slow features. It was decided that two a-values would be used, $a = 1$ and $a = 1.3$.

Secondly, the a-values were varied to see where the random forest would have the same mean cost as the benchmark. This was done for the three feature setups described above. Since random forest predicts 1 and -1's, as one increases a the mean cost will increase. However, it should outperform the benchmark when $a = 1$. Now, let the mean cost difference between random forest and the benchmark be defined as:

$$\Delta(a) = \hat{c}_{\text{rf}}(a) - \hat{c}_{\text{oz}}$$

where $\hat{c}_{\text{rf}}$ is the mean cost for random forest and $\hat{c}_{\text{oz}}$ is the mean cost for the benchmark. Then, one wishes to find a_{crit} such that:

$$\Delta(a_{\text{crit}}) = \hat{c}_{\text{rf}}(a_{\text{crit}}) - \hat{c}_{\text{oz}} = 0$$

Chapter 6

Results

In this chapter, the results are presented. It begins with the investigation of the response definition. Afterwards, the results from the group importance for slow- and fast-acting features are presented. Lastly, the mean cost is investigated.

6.1 Response definition

Section 5.1.2 described how the rolling quantiles would be used to classify the price changes into discrete responses. Below, the quantiles are displayed as a function of time:

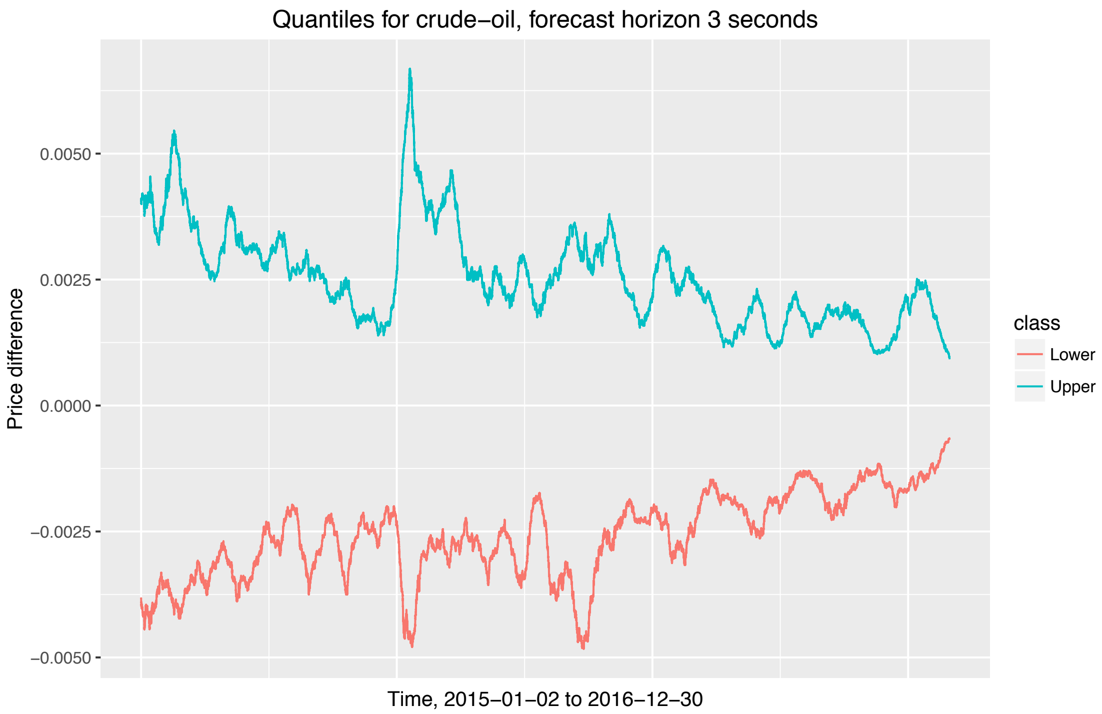


Figure 6.1: Quantiles for the time period 2015-2016, at 3 seconds forecast horizon. The contract is crude oil.

As one can see in figure 6.1, the quantiles seem to have a drift. This is not desirable, since the dynamic in which the prices are classified changes. This is the case for all forecast horizons in crude oil (which is displayed in appendix, B.1). One reason for this drift could be that the volatility has changed during the time period 2015-2016.

To address this, one approach is to scale the prices with their volatility. However, there was a concern that this normalization would remove the tick size structure, i.e. one expects that there should be an increase in frequency of price changes at the tick size. The tick size is the lowest allowed price difference between the levels in a limit order book.

However, there were no significant increases in frequency at the tick size. Figure B.7 in Appendix is an example of a histogram at a given time. It shows no apparent sign of an increase at the tick size. Therefore, the prices were normalized by their volatility, namely the Yang-Zhang volatility, described in section 3.4.1.1. This was done for all contracts and all forecast horizons. The price changes were defined as:

$$\tilde{p}_t = \frac{p_{t+1} - p_t}{\sigma_{YZ}} \tag{6.1.1}$$

The Yang-Zhang volatility had been estimated from the day before, i.e. all data points within the same day and contract were normalized by the same volatility value. Moreover, λ was set to $\frac{1}{22}$. This yields the following quantiles figure:

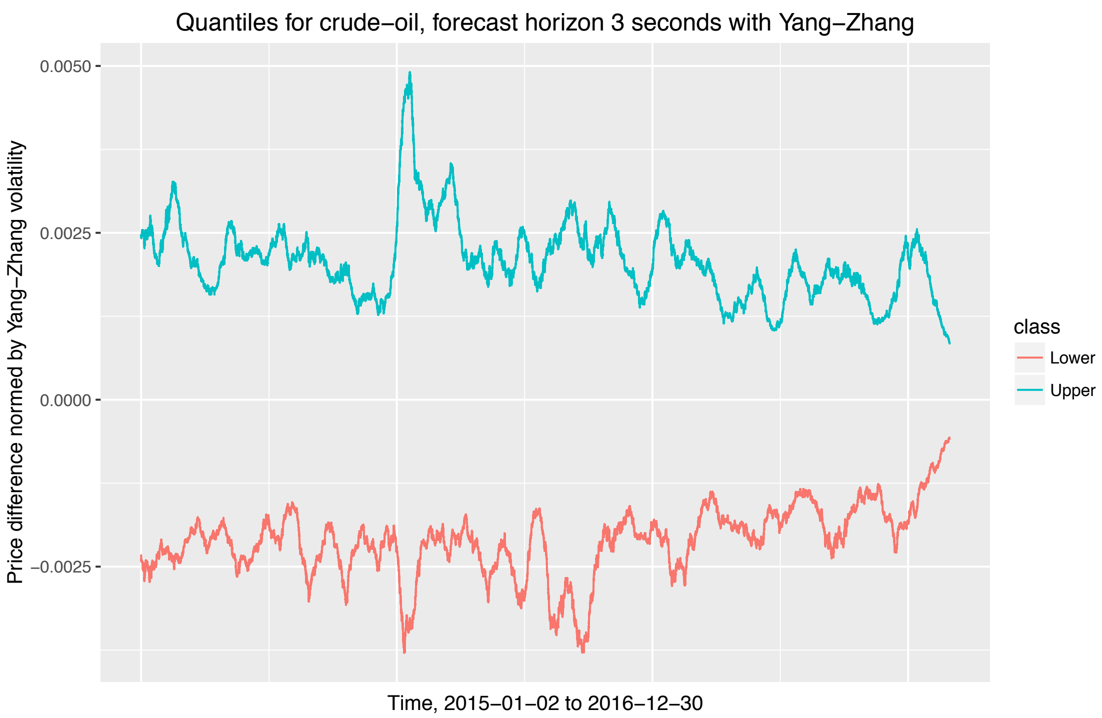


Figure 6.2: Quantiles for the time period 2015-2016, at 3 seconds forecast horizon and normalized with the Yang-Zhang volatility. The contract is crude oil.

One can see from figure 6.2 that the quantiles seem to have a lower drift than in figure 6.1 (the quantiles with/without Yang-Zhang on all forecast horizons for crude oil is the in Appendix, B.1). Therefore, equation (6.1.1) was used to compute the price changes. Furthermore, the same procedure as in section 5.1.2 was used to group the data points into discrete values (i.e. quantiles), however the normalized price changes in equation (6.1.1) were used instead.

Note that most technical indicators were built upon price and therefore these features were also normalized with the Yang-Zhang volatility. Furthermore, the vrv and vs features were normalized by the Yang-Zhang volatility.

6.2 Group importance

In this section, the group importance results are presented. The group importance was calculated using algorithm 7. The number of iterations was chosen to be $N = 10$ in order to understand how much the randomness effected the results. Afterwards, the individual feature-importance was computed. This is done for the univariate models and the multivariate one.

Lastly, a mid price analysis of the group importance and individual feature importance were evaluated. This was done since the mid price is commonly used to define the price in a limit order book setting. The results are in Appendix, A.1.

6.2.1 Univariate models

Below, in figures 6.3, 6.5, 6.7 and 6.9 the reader finds the estimated group importance for the fast- and slow-acting features for different forecast horizons and contracts. The figures display their means and one standard deviation error bars.

After that, the individual feature importance were computed. Moreover, the features were ranked, i.e. the feature with the highest feature-importance got rank 1, etc. This is displayed in figures 6.4, 6.6, 6.8 and 6.10 for different forecast horizons and contracts.

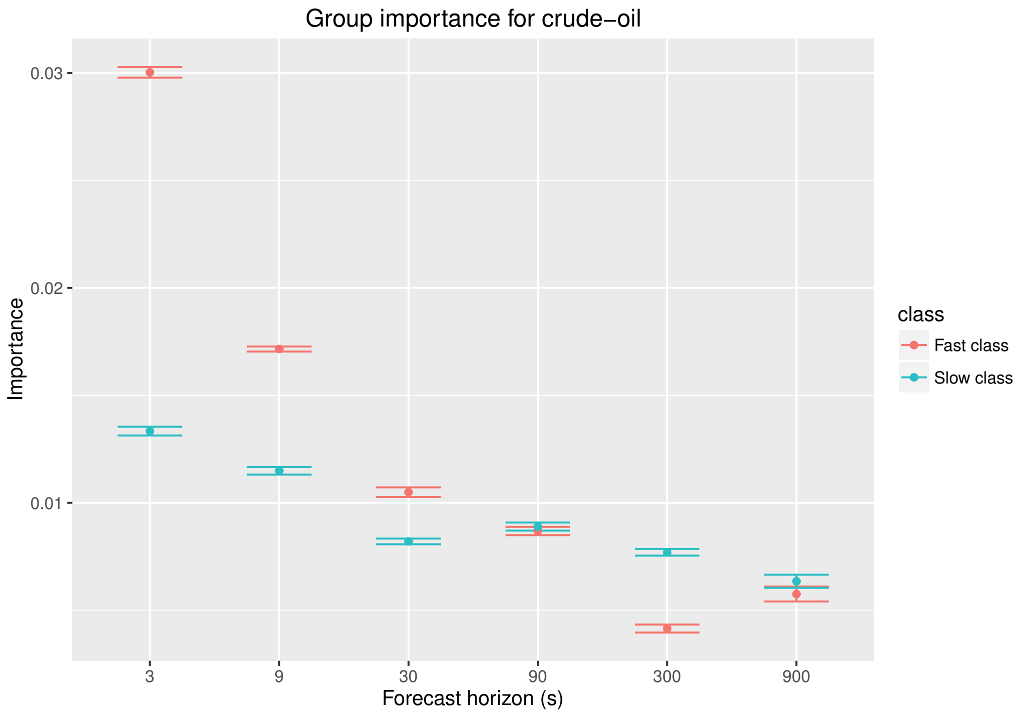


Figure 6.3: Group importance for the fast- and slow-acting features on different forecast horizons. The contract is crude oil.

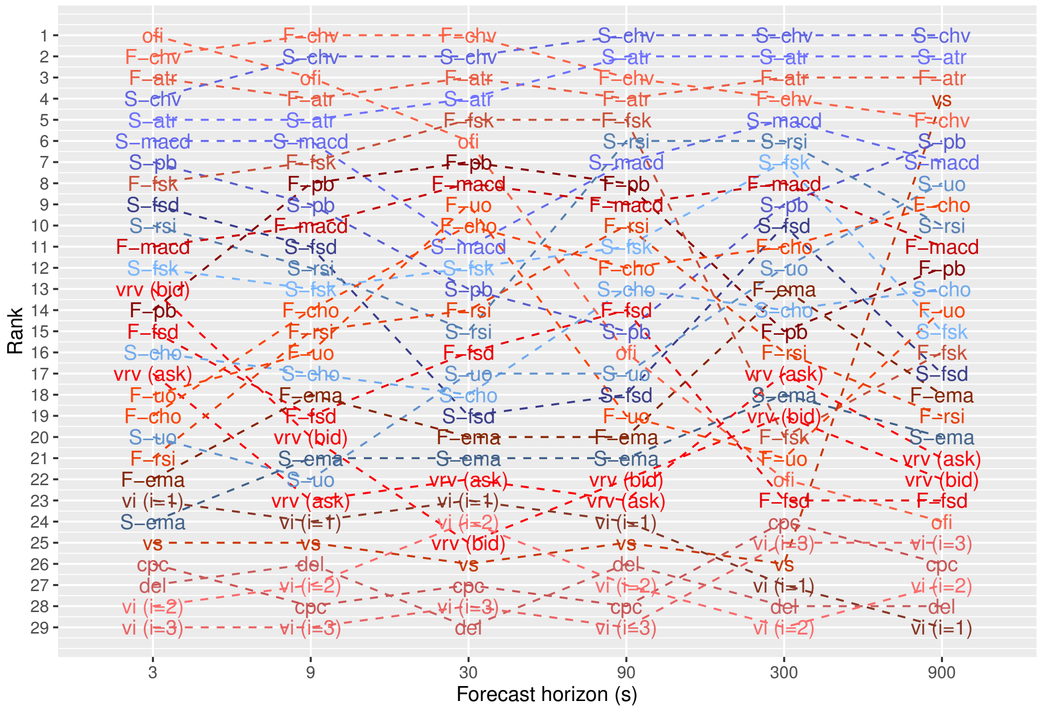


Figure 6.4: The ranking of individual features on different forecast horizons. The contract is crude oil.

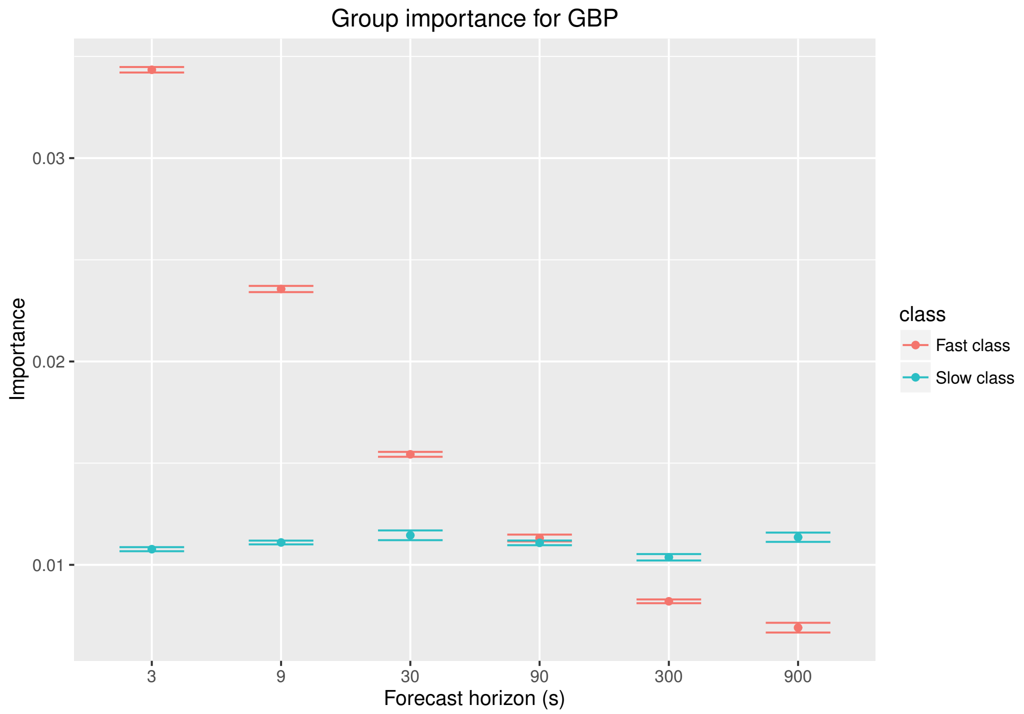


Figure 6.5: Group importance for the fast- and slow-acting features at different forecast horizons. The contract is GBP.

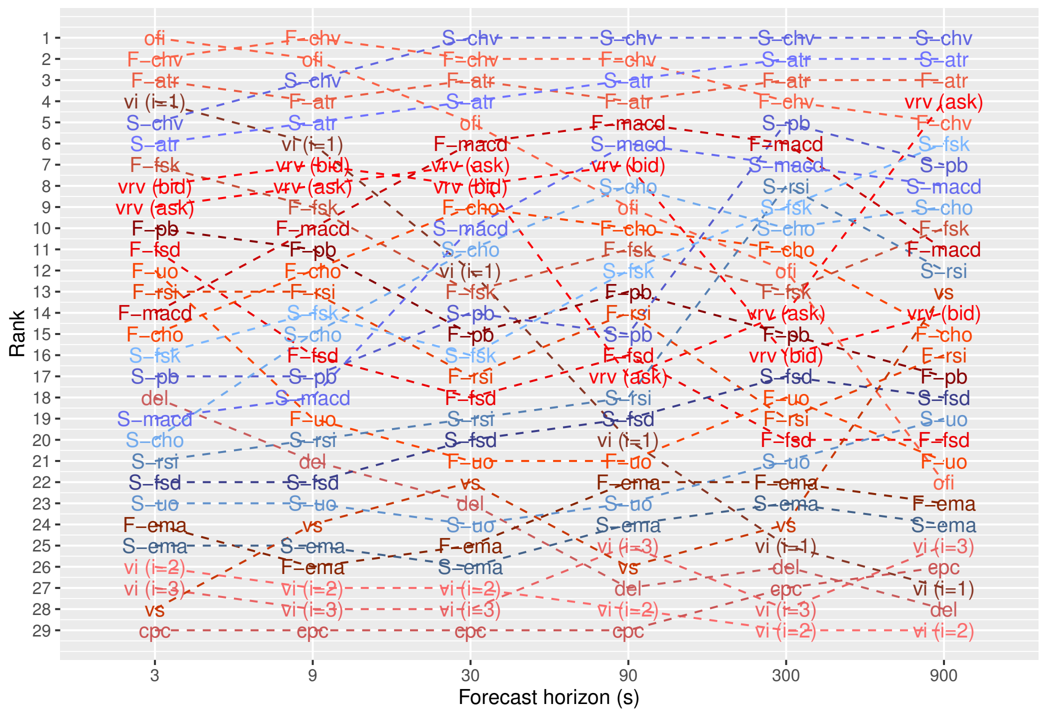


Figure 6.6: The ranking of individual features on different forecast horizons. The contract is GBP.

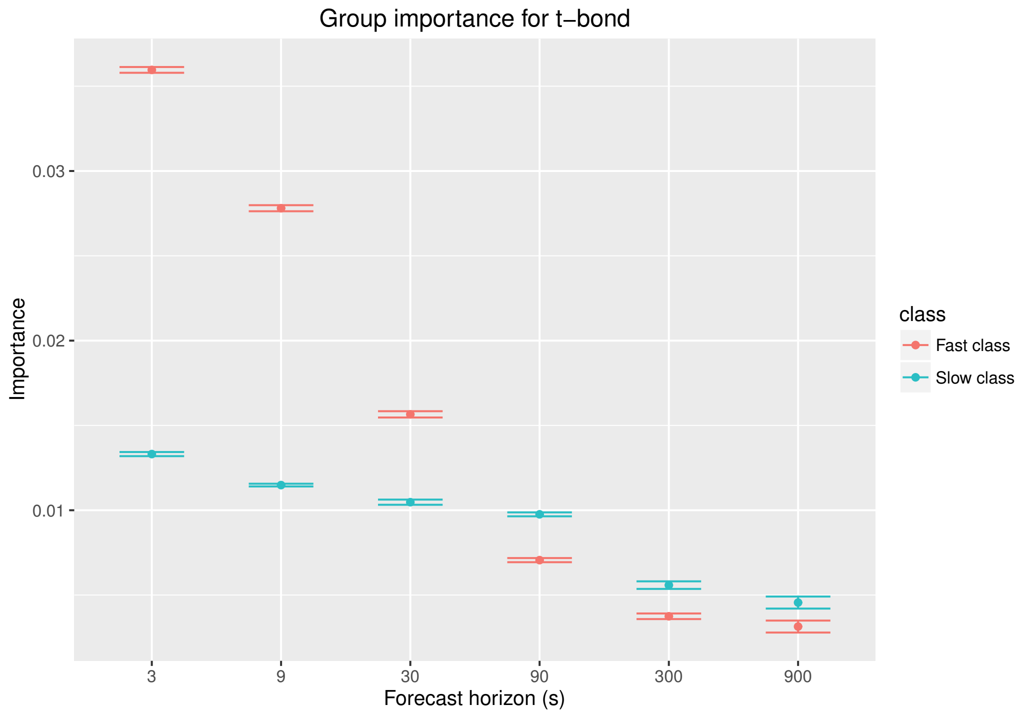


Figure 6.7: Group importance for the fast- and slow-acting features on different forecast horizons. The contract is t-bond.

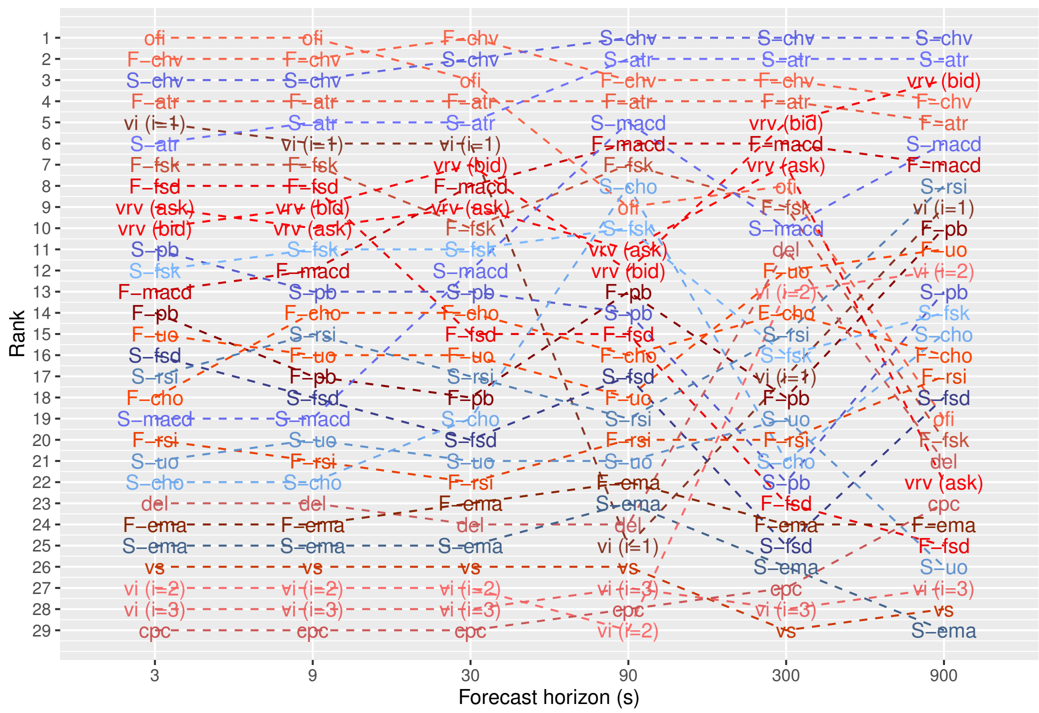


Figure 6.8: The ranking of individual features on different forecast horizons. The contract is t-bond.

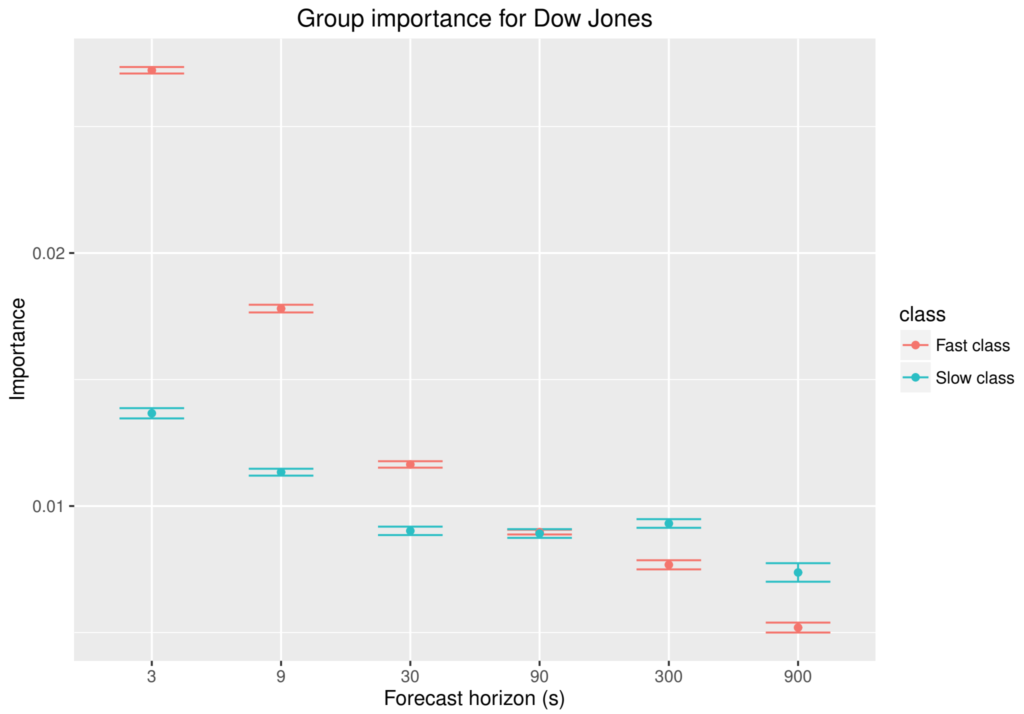


Figure 6.9: Group importance for the fast- and slow-acting features on different forecast horizons. The contract is Dow Jones.

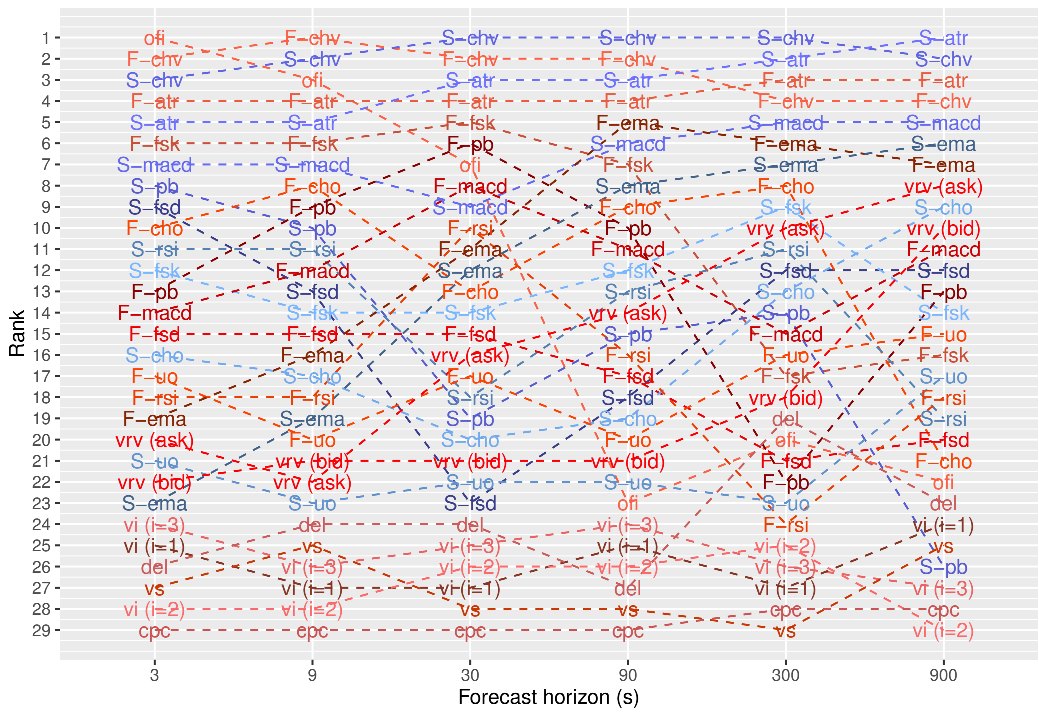


Figure 6.10: The ranking of individual features on different forecast horizons. The contract is Dow Jones.

As one can see from the group importance figures above, the fast-acting features were clearly better for the shorter forecast horizons, across contracts. The difference between the two groups shrinks as one increases the forecast horizon. At the 90 seconds forecast horizon the slow-acting features had approximately the same group importance as the fast-acting features, except for the t-bond contract. There, the slow-acting features had a higher group importance than the faster ones. Furthermore, for the remaining forecast horizons, 300 and 900 seconds the slow-acting features had a higher group importance than the fast-acting ones.

Now, if one examine the individual feature importance one notices that the ofi feature was the best for the shortest forecast horizon (3 seconds) and the volatility indicators also performed well, across contracts. As one increases the forecast horizon the ofi feature importance dropped while the volatility indicators still performed well.

Furthermore, there is a hierarchy between volatility indicators; the faster volatility indicators performed better at the short-forecast horizons. This changes as one increases the forecast horizon and the longer ones became better. On the contrary, most of the technical indicators do not have this structure for the majority of contracts, with the except for the GBP contract.

For for the limit order book derived features, some of them perform relatively poorly across forecast horizons, with the exception of the ofi feature. However, for the t-bond contact and the GBP contract they performed better.

The full list of individual feature importance values can be found in Appendix, B.3.

6.2.2 Multivariate model

Since the group importance and the individual feature-importances were rather similar amongst the contracts, it was decided to the same analysis using all data from the four contracts in one model, i.e. from four univariate models to one multivariate model.

The group importance for the two groups was estimated, as described earlier. In figure 6.11, their means and one standard deviation error bars are displayed:

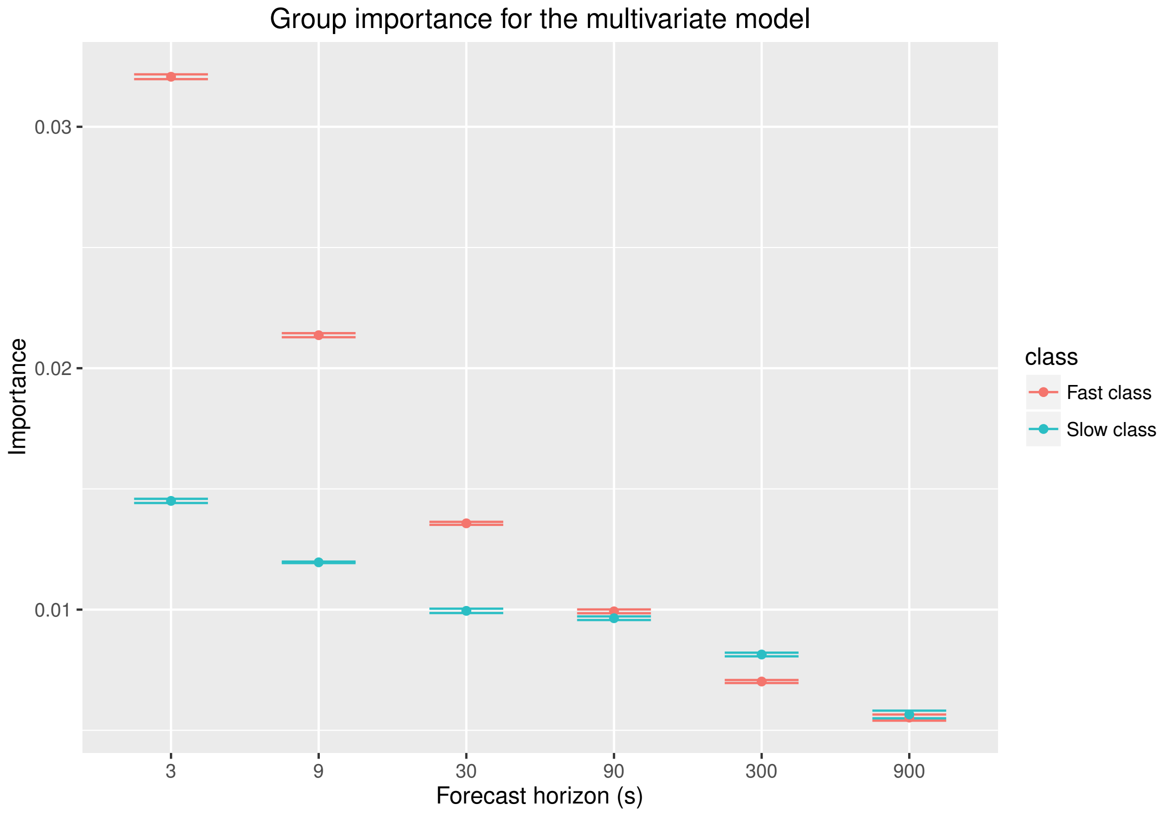


Figure 6.11: Group importance for the fast- and slow-acting features on different forecast horizons for the multivariate model.

As one can see from figure 6.11, the fast-acting features were clearly better for the shorter forecast horizons. The difference between the two groups shrinks as one increases the forecast horizon. Additionally, for the 300 seconds forecast horizon the group importance for slow-acting features became higher than the fast-acting features. However, the difference between the group becomes very small for the 900 seconds forecast horizon.

Now, the individual feature importance were computed as described earlier. This is displayed in figure 6.12 for different forecast horizons:

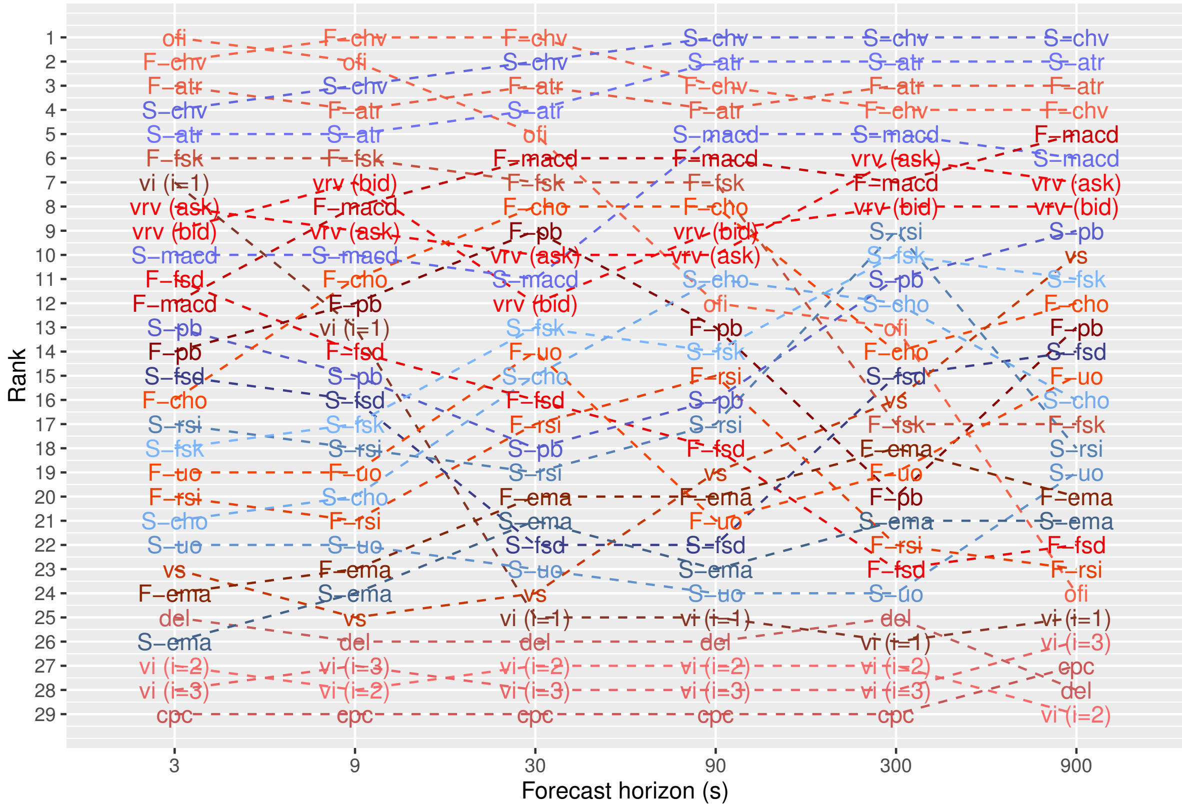


Figure 6.12: The ranking of individual features on different forecast horizons for the multivariate model.

As one can see from figure 6.12, the ofi feature was the best for the shortest forecast horizon (3 seconds) and the volatility indicators also performed well. As one increases the forecast horizon the ofi feature importance dropped while the volatility indicators still performed well.

Furthermore, there is a hierarchy between volatility indicators; the faster volatility indicators performed better at the short forecast horizons. This changes as one increases the forecast horizon and the longer ones became better. On the contrary, most of the technical indicators do not have this structure.

The limit order book derived features performed better for the multivariate model than it did for the univariate ones.

The list of individual feature importance values for the multivariate model can be found in Appendix, section B.3, table B.5.

6.3 Mean cost

The mean cost of the random forest models was compared to the benchmark, as described in section 5.5. In the first step, the three feature setups ($\mathcal{F}_f$, $\mathcal{F}_s$, $\{\mathcal{F}_f, \mathcal{F}_s\}$) were compared to the benchmark for two a-values. The first one was $a = 1$. This corresponds to the "usual" error rate, i.e. how often does the method predict incorrect, regardless of class. Secondly, the a-value was increased to 1.3 in order to investigate how the models perform. Random forest's mean cost will increase, however this is not the case for the benchmark. A lower value of the mean cost implies better performance. This was done for the four contracts. Moreover, the average mean cost for the four contracts were compared to the multivariate model.

Finally, the a_{crit}-values was investigated for the three feature setups, as described in section 5.5. A higher value of a_{crit} implies better performance. Furthermore, this was done for the same contracts and forecast horizons described in the section 5.1.

6.3.1 Univariate models

Below, in figures 6.13, 6.16, 6.19 and 6.22 the reader finds the estimated mean cost with $a = 1$ for the three models at different forecast horizons and contracts.

After that, the mean cost was estimated for $a = 1.3$. This is displayed in figures 6.14, 6.17, 6.20 and 6.23.

Lastly, the a_{crit}-values was estimated. This is displayed in figures 6.15, 6.18, 6.21 and 6.24.

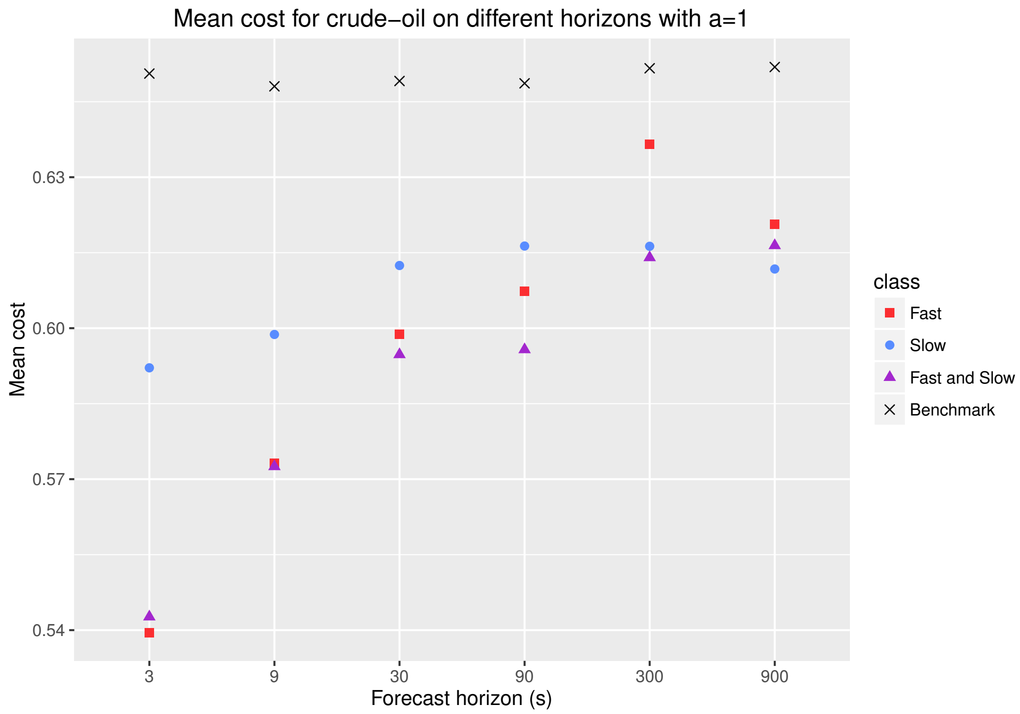


Figure 6.13: The mean cost for the four models on different forecast horizons with $a = 1$. The contract is crude-oil.

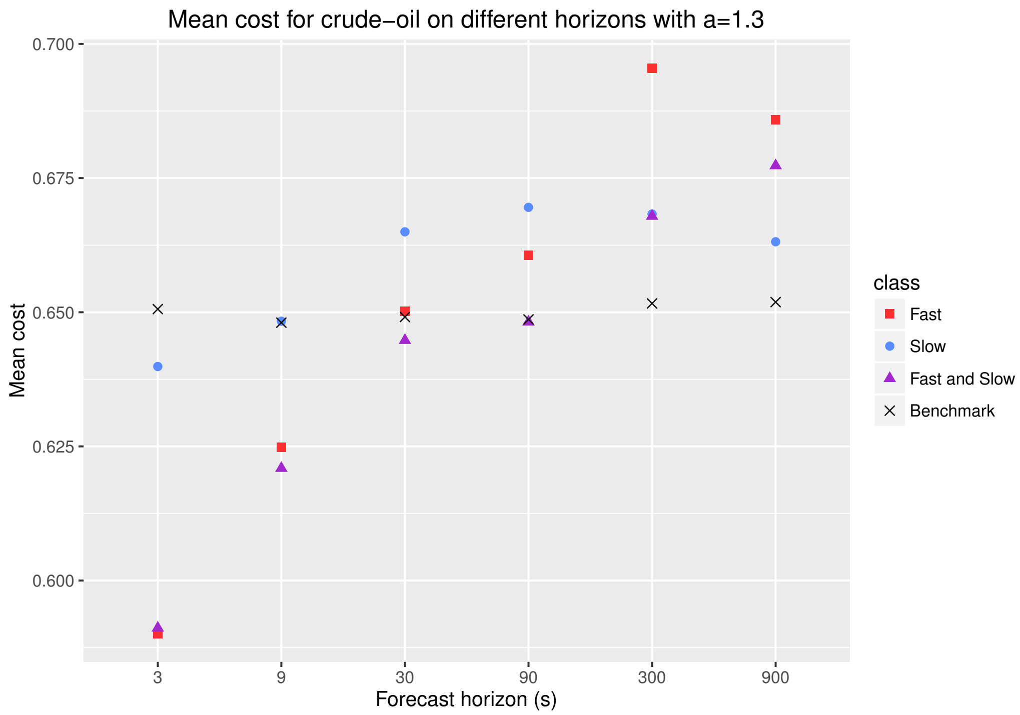


Figure 6.14: The mean cost for the four models on different forecast horizons with $a = 1.3$. The contract is crude-oil.

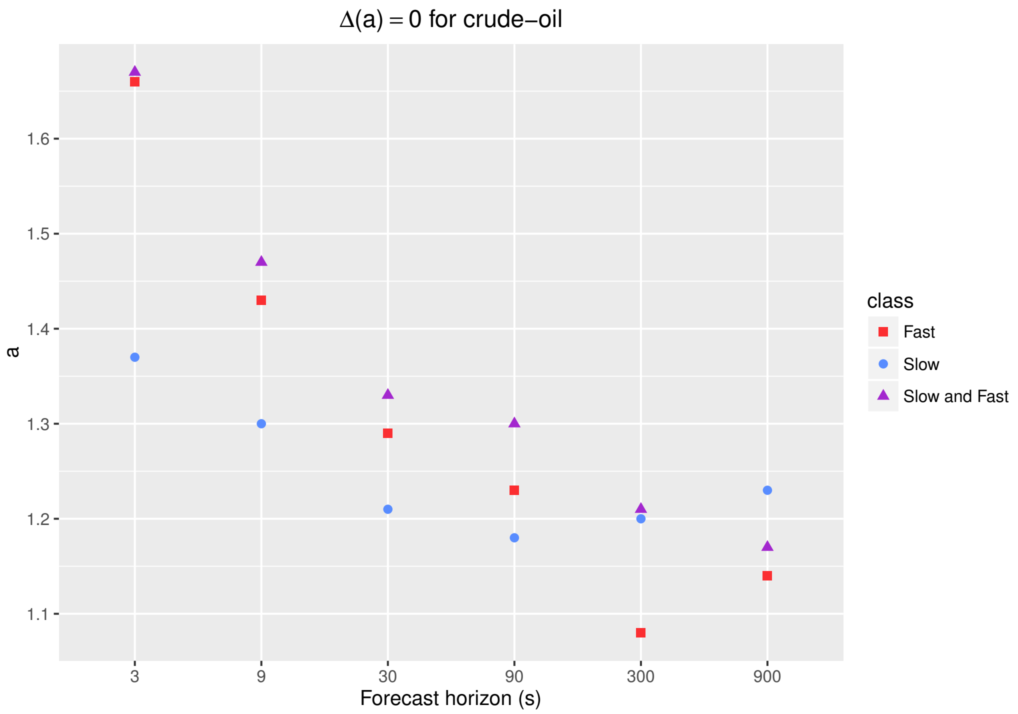


Figure 6.15: The a_{crit}-values for the three models. The contract is crude-oil.

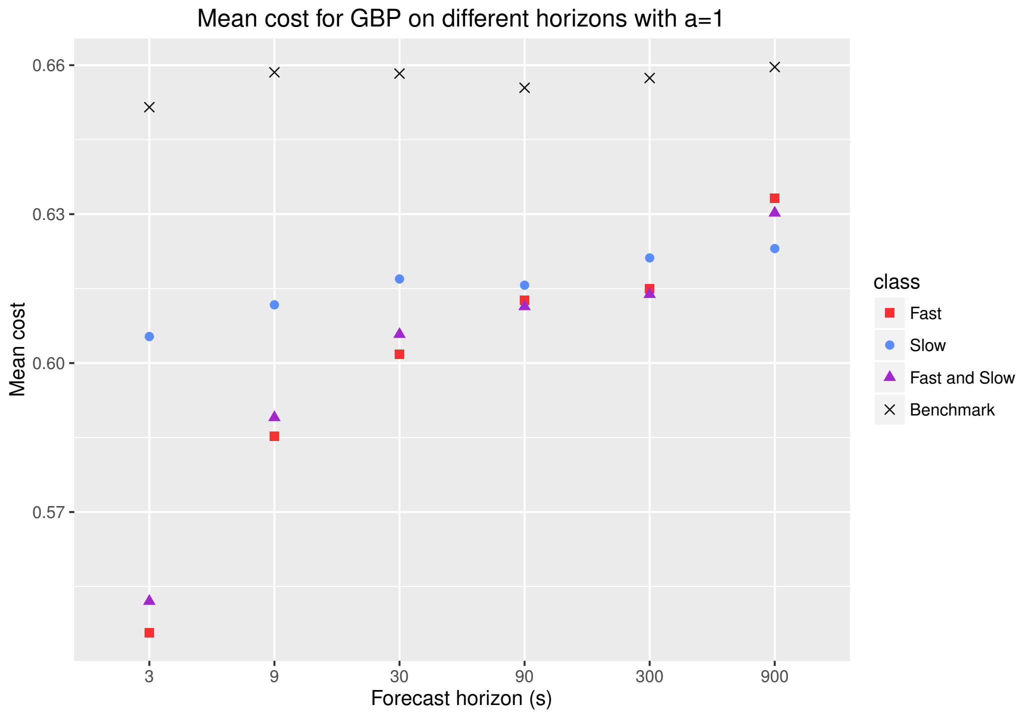


Figure 6.16: The mean cost for the four models on different forecast horizons with $a = 1$. The contract is GBP.

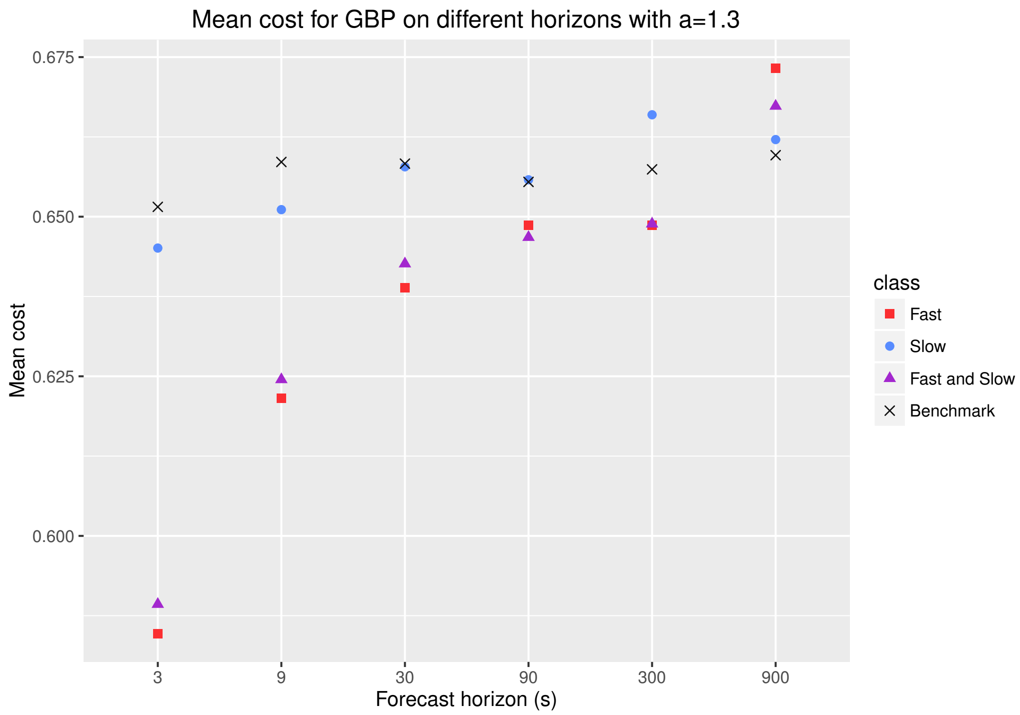


Figure 6.17: The mean cost for the four models on different forecast horizons with $a = 1.3$. The contract is GBP.

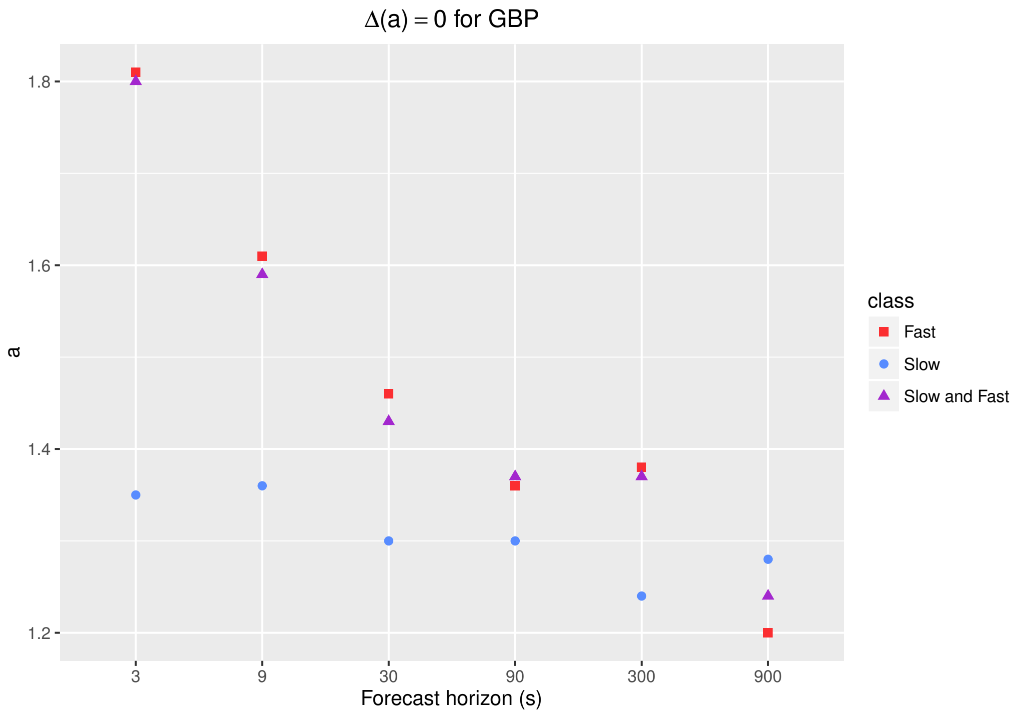


Figure 6.18: The a_{crit}-value for the three models. The contract is GBP.

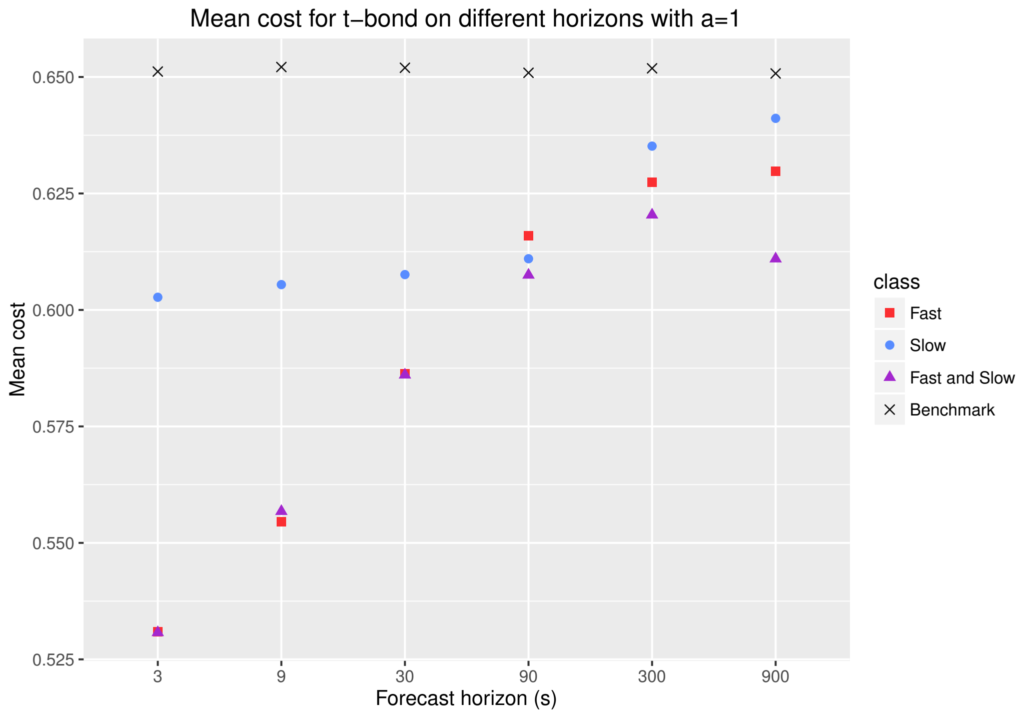


Figure 6.19: The mean cost for the four models on different forecast horizons with $a = 1$. The contract is t-bond.

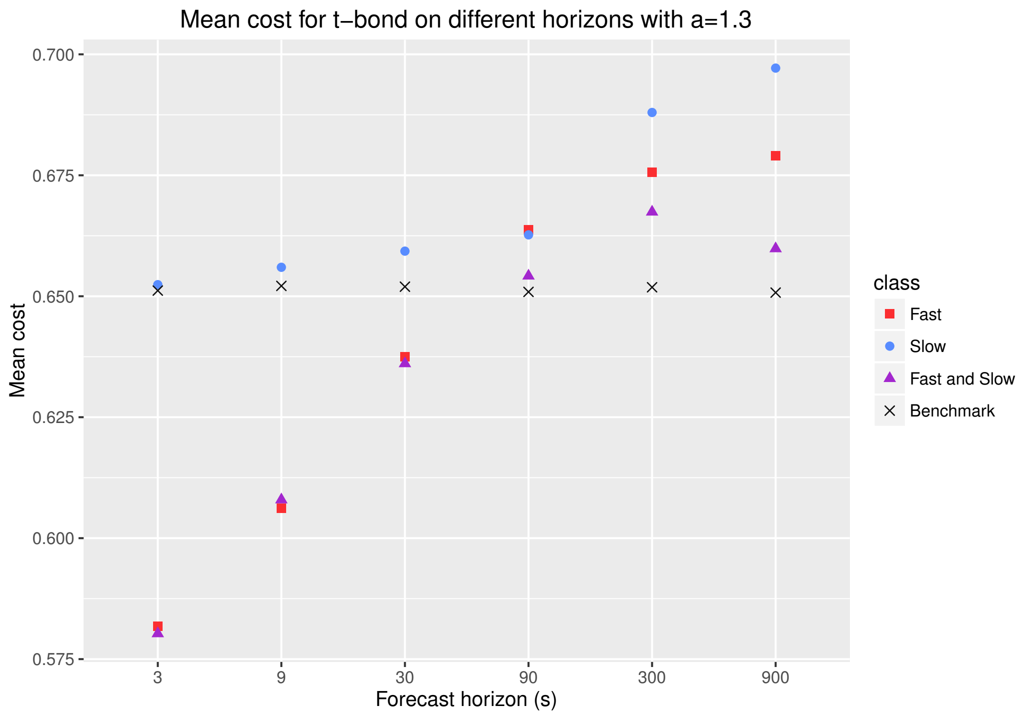


Figure 6.20: The mean cost for the four models on different forecast horizons with $a = 1.3$. The contract is t-bond.

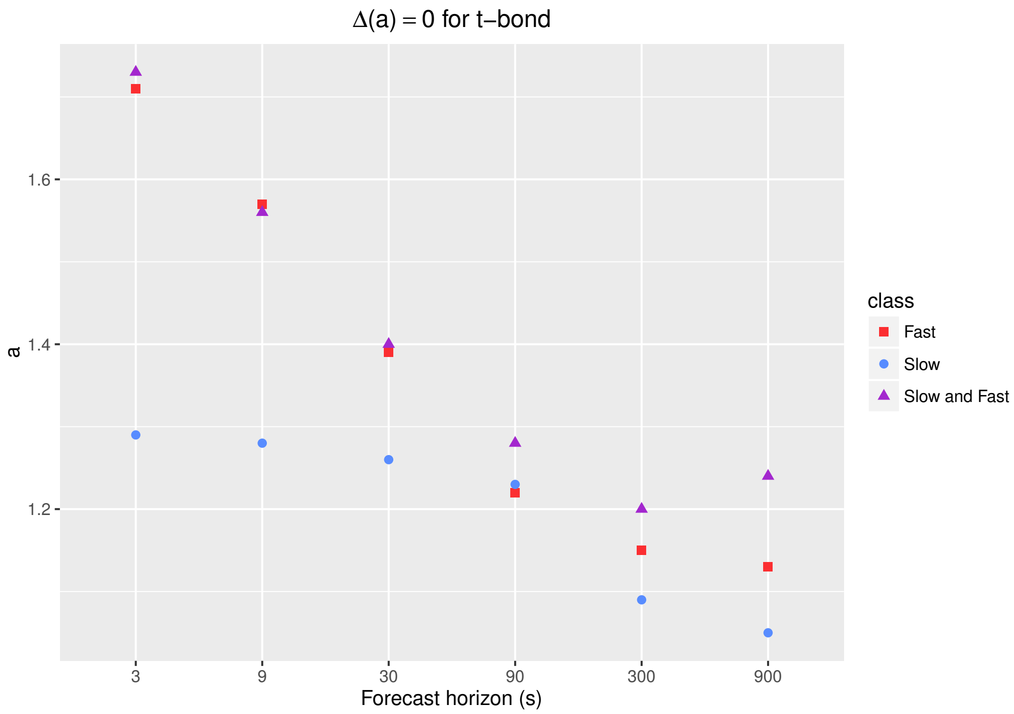


Figure 6.21: The a_{crit}-values for the three models. The contract is t-bond.

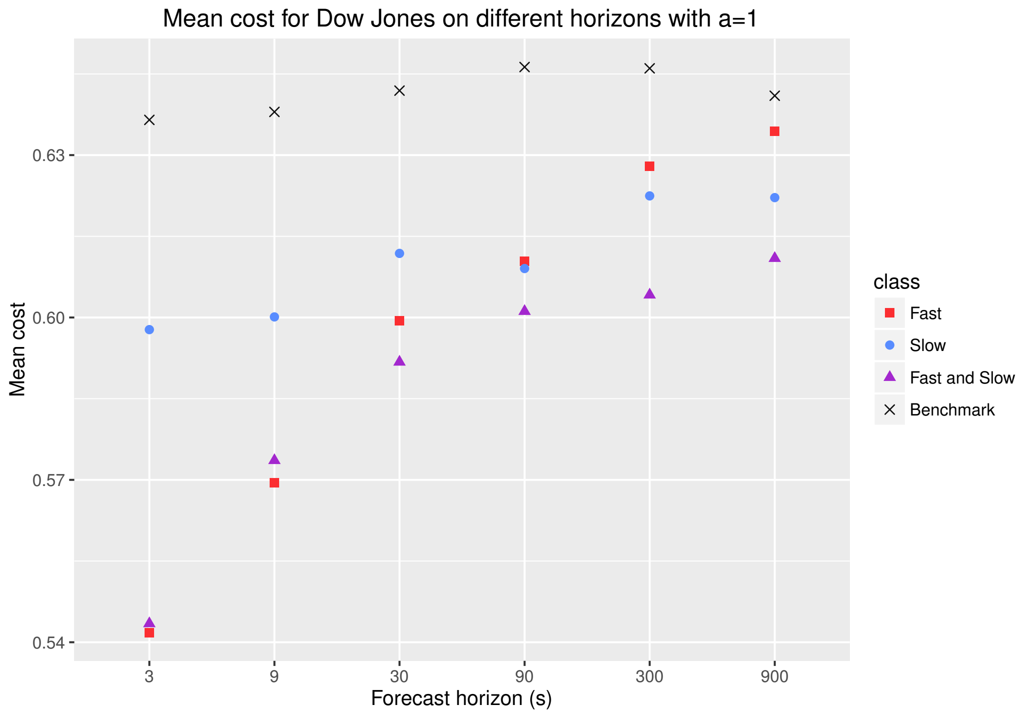


Figure 6.22: The mean cost for the four models on different forecast horizons with $a = 1$. The contract is Dow Jones.

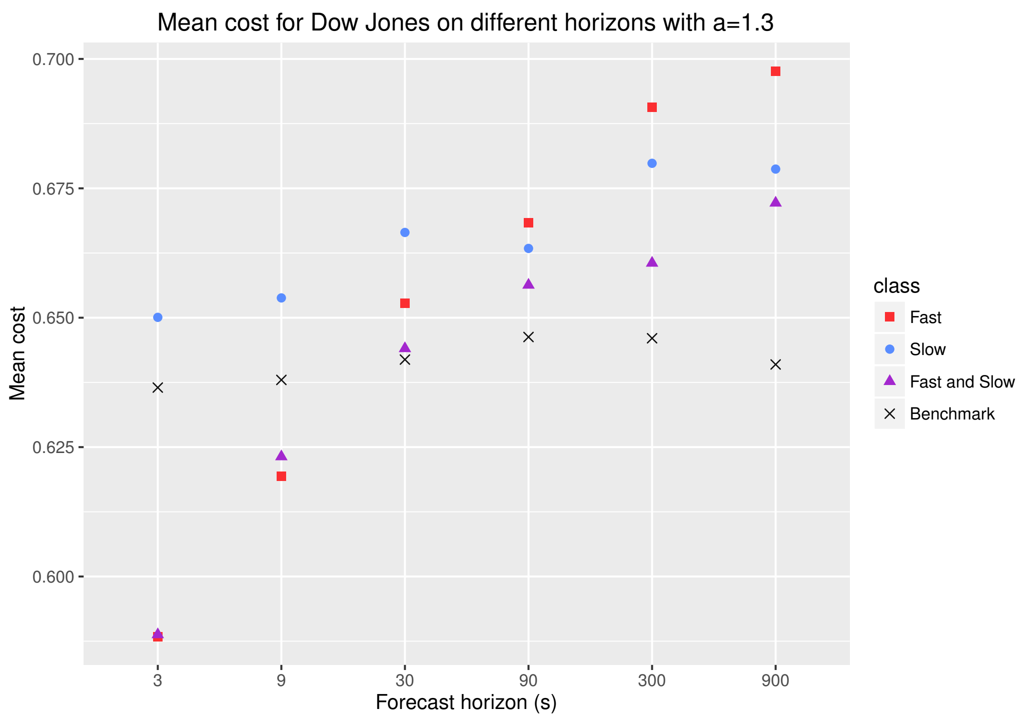


Figure 6.23: The mean cost for the four models on different forecast horizons with $a = 1.3$. The contract is Dow Jones.

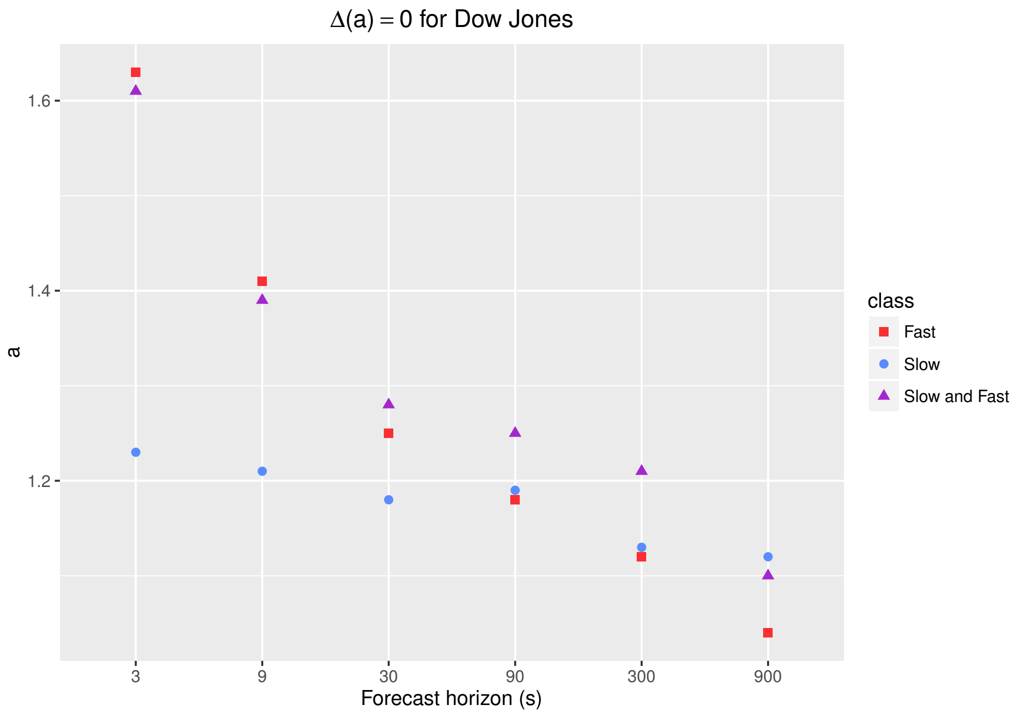


Figure 6.24: The a_{crit}-values for the three models. The contract is Dow Jones.

As one can see from figures 6.13, 6.16, 6.19 and 6.22, the fast-acting features were substantially better at the 3 seconds forecast horizon than the slow-acting, across contracts. However, as one increases the forecast horizon the gap between the two models decrease. At the 900 seconds forecast horizon, the slow-acting features had a lower mean cost than the fast-acting ones, except for the t-bond contact. There, the slow-acting features did not become better than the slow-acting features for longer forecast horizons, but it got better at the 90 seconds forecast horizon.

The models with slow and fast-acting features are roughly the same as the best models, across forecast horizons. However, it seems to be slightly worse at the 3 and 900 seconds forecast horizon compared to the best models. This seems to be the case for all contacts except for the t-bond contract where the model with slow and fast-acting features are roughly the same as the best model until the 90 seconds forecast horizon. Then, the mixed group became better than both the fast acting features, but also the slow-acting features.

All of the models are better than the benchmark, across forecast horizons and contracts.

Now, if one examine the figures 6.14, 6.17, 6.20 and 6.23, they resemble the previous ones. This is also the case for the a_{crit}-figures, 6.15, 6.18, 6.21 and 6.24.

6.3.2 Multivariate model

First of all, random forest was trained, using data from all four contracts. Afterwards, the mean cost for the multivariate model was compared to average mean cost for the four contracts.

In figure 6.25, the difference between the mean cost of the multivariate model and the average mean cost for the univariate models are displayed, given different forecast horizons and feature setups with $a = 1$. A negative number indicates that the multivariate models has a lower cost (i.e. performs better) and vice versa.

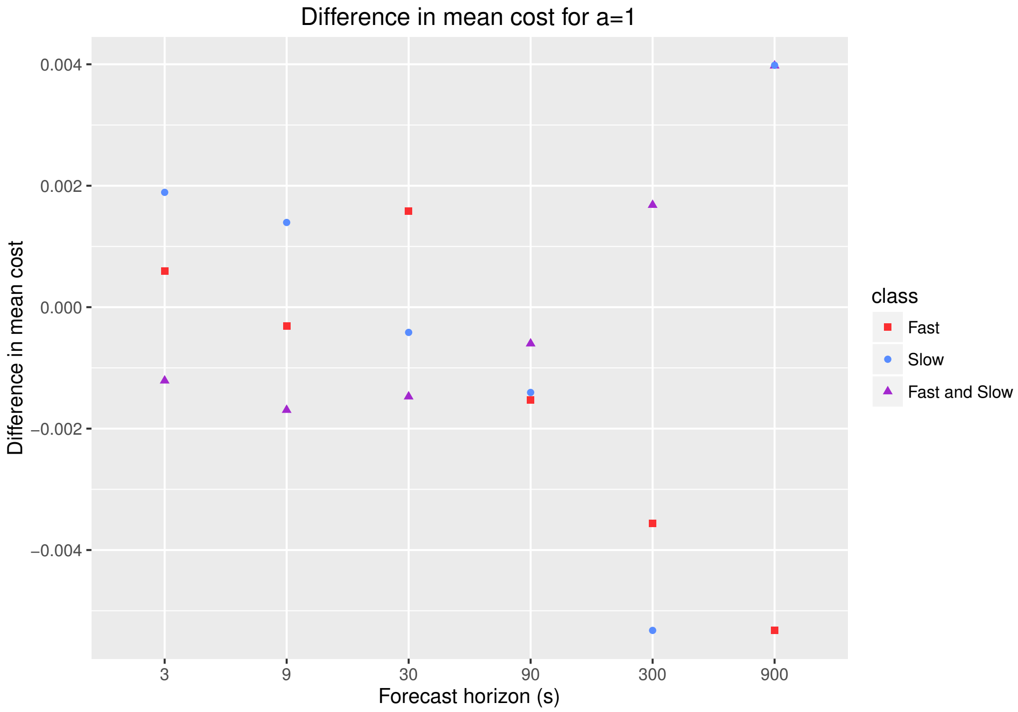


Figure 6.25: Difference in mean cost between the multivariate random forest and the average of the four contracts with $a = 1$.

As one can see in figure 6.25, the difference seems to oscillate between 0. It is slightly skewed towards the negative side. However, this is not the case for the forecast horizons 3 and 900 seconds. The same analysis was done with $a = 1.3$:

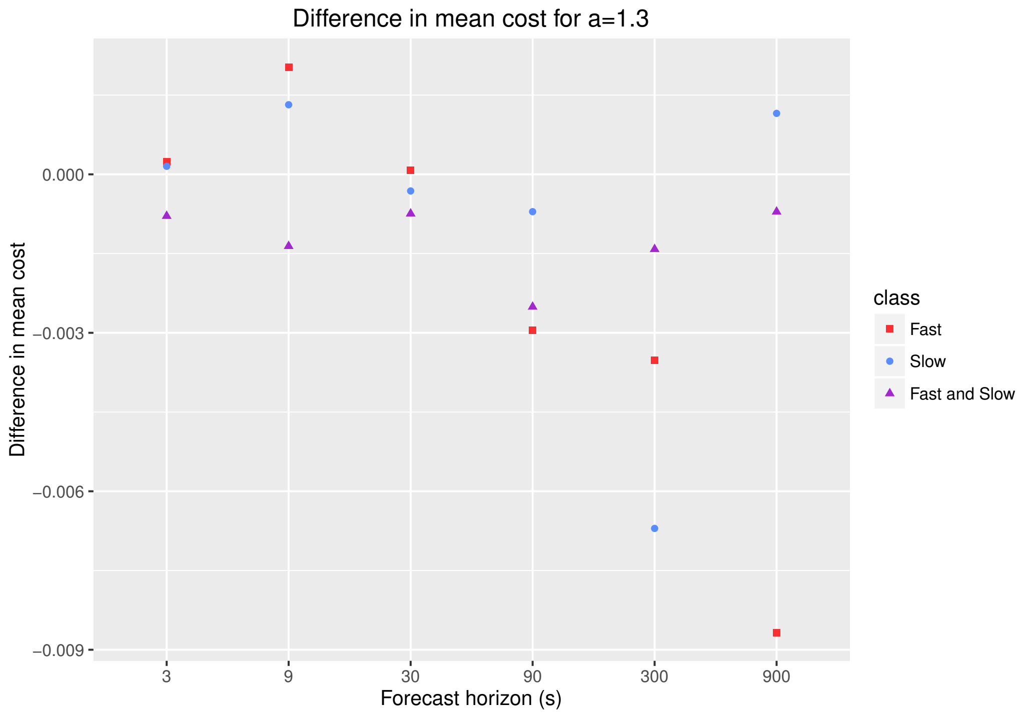


Figure 6.26: Difference in mean cost between the multivariate random forest and the average of the four contracts with $a = 1.3$.

Figure 6.26 resembles figure 6.25 with the excepts that it seems to be more skewed towards the negative side.

Chapter 7

Discussion

In this chapter, the results obtained in chapter 6 are examined. The chapter begins with a discussion about the group importance results from section 6.2. Then it continues with the results from section 6.3 regarding the mean cost.

7.1 Group importance

In section 6.2.1 the group importance for the four contracts at the six forecast horizons was reviewed. The main objective for the investigation was to examine if there is an difference in group importance between the fast- and slow-acting features. As one can see from the figures in section 6.2, the fast-acting features had a higher group importance value at the 3 seconds forecast horizon, across contracts.

However, when the forecast horizon was increased, the fast-acting features' group importance drops quite substantially. This is the case for slow-acting features as well (except for the GBP contract), however they did not drop as drastically. For the GBP contract the group importance for the slow-acting features seem to be relatively stable. The slow-acting features becomes better at the 90/300 seconds forecast horizon.

An interesting observation is that as one increases the forecast horizon, the group importance for both groups dropped. One explanation for this could be that it becomes difficult to predict price movements at longer forecast horizons. One could speculate that it is due to larger profit margins and therefore becomes more "noisy" since more market participants operates there.

Now, if one examines the individual feature importances, the order flow imbalance feature was the best feature at the 3 seconds forecast horizon, across contracts. In the original paper, where the order flow imbalance feature is described [16], it is showed that it had a good explanatory power on the price changes, however the finding in section 6.2 suggests that it has a good predictive power for shorter forecast horizons. When the forecast horizon was increased, the ofi importance decreased, drastically.

The volatility indicators performed well, across forecast horizons and contracts. Both the fast- and slow-acting atr and chv features were among the best at the 3 seconds horizon, across contracts. An interesting observation is that the faster version of the volatility indicators outperforms the slower versions at 3 and 9 seconds forecast horizons. However, this dynamic changes around the horizons 30 and 90 seconds. For 300 and 900 seconds forecast horizons the slower versions performed better. Furthermore, the slow and fast volatility indicators were amongst the best at 900 seconds forecast horizons.

Most technical indicators' ranks were centred around the middle, i.e. they were not amongst the best or worse. For the GBP contract' technical indicators there seems to a hierarchy between the fast-acting and the slow-acting, i.e. the faster indicators performed better on short forecast horizons while the slower performed better on the longer horizons. However, most contracts do not have this structure.

Some of the limit order book derived features performed relatively poorly across forecast horizons. However, the volume imbalance feature (at the first level) performed well on the GBP contract and the t-bond contract. Moreover, one notices that the vrv and vs features became better at the longer forecast horizons, which is a bit counter-intuitive. One explanation for this phenomenon is that these features operate on deeper levels of the limit order book and therefore has a longer time dependency, which can be useful on longer forecast horizons.

If one examines the results from the multivariate model they resemble the univariate ones. The fast-acting features perform substantially better at the shorter forecast horizons. At the 90 seconds forecast horizon the group importances are roughly the same. However, the difference between the fast- and slow-acting group importance is smaller at the 900 seconds forecast horizon for the multivariate model than for the univariate ones.

The individual feature importances for the multivariate model also matches the univariate results. One difference is that some limit order book derived features had a higher rank for the lower forecast horizons than they did for the univariate models.

The results from section A.1 and 6.2.1 regarding group importance for the mid price definition and the volume weighted one for the Dow Jones contract are rather similar. The results from the mid price definition can be found in Appendix, A.1. When comparing the individual feature importance between the two definitions, one notes that some of the volume based limit order book derived features performed better for the mid price definition than on the volume weighted price definition, for example the vi (on the first level) and the vrv features.

One explanation for this could be that the volume is already incorporated into the volume weighted price definition. Therefore, a feature that is based on volume, for example the volume imbalance feature, will perform worse on the weighted volume price definition since some of it predictive power has already been extracted in the price definition.

In essence, the fast-acting features were substantially better than the slow-acting features in terms of group importance on shorter forecast horizons. For the longer forecast horizons, the slower features were slightly better.

7.2 Mean cost

In section 6.3.1 the mean cost results are presented. To begin with, one notices that the mean cost for $a = 1$ at the 3 seconds horizon is dominated by the fast-acting features, quite substantially. Also, the mixed group with both fast- and slow-acting features perform well there. However, it is interesting that the fast-acting features seem to outperform the mixed feature group slightly (except for the t-bond contract where they are roughly equal). This suggests that adding the slower features actually hurts the prediction ability for the 3 seconds forecast horizon.

The fast and mixed group are dominating until the 90/300 seconds forecast horizon, where they start to become roughly equal to the slow-acting features. At 900 seconds forecast horizon the slow-acting features were better than the fast-acting ones, across contracts, except for the t-bond contract, where the slow-acting features become better at 90 seconds forecast horizon. However, later it shifts back and the fast-acting features becomes better at 300 and 900 forecast horizon, which was unexpected.

It seems to be a trend that the mean cost is increasing with forecast horizons across contracts for all three groups. This is aligned with the findings in section 6.2, i.e. it becomes more difficult to predict the outcomes on longer horizons. However, all three methods outperformed the benchmark across forecast horizons and contracts.

Then, the penalty was increased for predicting an incorrect price direction, i.e. a was set to 1.3. As one can see from the figures in section 6.3, the dynamics are fairly similar to the results when $a = 1$, i.e. the groups that had the smallest mean cost when $a = 1$ also had it when $a = 1.3$.

The three random forest models were all better than the benchmark at 3 seconds forecast horizon. The fast-acting features and the mixed group performed substantially better. The benchmark becomes better at 30 seconds forecast horizon for all contracts expect GBP, where the benchmark gets better at 900 seconds forecast horizon. At 900 seconds forecast horizon the benchmark was better than all the random forest models, across contracts.

If one inspects the a_{crit} figures they follow the theme from the mean cost calculations. For the 3 seconds forecast horizons the fast and mix group dominated. The allowed a_{crit} was around 1.6 to 1.8. For the slow-acting group it was substantially lower with a value of around 1.2 to 1.4. When the forecast horizon was increased the allowed a_{crit} dropped. It is the most significant for the fast-acting features and the mixed group. The slow group seems to be stable until the 90 second forecast horizon, then it also dropped. This was also observed in the mean cost results for both $a = 1$ and $a = 1.3$.

At the 900 second forecast horizon the slower features become better across all contracts except for the t-bond contract. However, the mixed group is better than both of them which indicates that a combination is necessary to achieve the highest a_{crit}.

Moreover, the results from the multivariate analysis in section 6.3.2 suggest that the difference between the multivariate model and the univariate ones do not differ greatly, in terms of mean cost. Although, there is a difference for the 900 seconds forecast horizon for both $a = 1$ and $a = 1.3$, i.e. the univariate models are better for this forecast horizon in terms of mean cost. This is supported by the results from the group importance for the multivariate model.

In essence, the three investigations with $a = 1$, $a = 1.3$ and to find a_{crit} showed similar outcomes. The fast-acting features were substantially better than the slow-acting features for lower forecast horizons. However, this changes for longer forecast horizons where the slow-acting features become better.

Chapter 8

Conclusions and Future Work

8.1 Conclusions

One of the objective questions described in section 1.1 was to investigate if the group importances for fast- and slow-acting features differ, given different forecast horizons. From section 6.2 one can see that this seems to be the case, i.e. the faster features group importance are better at shorter forecast horizons, however this dynamic changes as the forecast horizon increases. But if one inspects the individual feature importance, one notices that certain technical indicators are better than others, regardless of the length parameter. However, there seems to be a hierarchy within certain technical indicators, i.e. the fast-acting indicator are better at shorter forecast horizons and vise versa. One example of is the Chaikin volatility indicator.

Furthermore, some of the limit order book derived features performed poorly, across forecast horizons. However, this could be explained by the weighted price definition that was used.

The second objective question described in section 1.1 was to investigate if the mean cost differs for the fast- and slow-acting features, given different forecast horizons. From section 6.3 one can see that this is true for most contracts, i.e. the faster features has a lower cost on shorter forecast horizon. This changes as one increases the forecast horizon.

However, the findings are not conclusive since the t-bond contract did not show this behaviour at the 900 seconds forecast horizon. The fast-acting features were better for shorter forecast horizons, across contracts, quite substantially.

Moreover, the results from the multivariate analysis do not differ greatly from the univariate analysis, neither in terms of group importance or mean cost. The largest difference is for the 900 seconds forecast horizon.

In conclusion, the fast-acting features performed better than the slow-acting features at short forecast horizons. After that, the slow-acting features became better. So, as a rule of thumb, fast-acting features should be chosen for shorter forecast horizons and slow-acting features for longer forecast horizons, according to the results in section 6.2 and 6.3. However, it would be a good idea to investigate which type of feature performs best given a forecast horizons, since the importance for the fast-acting features varied extensively. The same holds for the slow-acting features.

8.2 Future work

This thesis lays the groundwork for group/feature importance for short-term price prediction. Now, a natural question is whether it is possible to apply this work in practice. One application could be for executing orders. Assuming that one has a "base" algorithm that dictates when an order should be executed, one could predict whether the price will increase/decrease in the near future. This can aid the decision whether an order should be executed now or at a later time in order to decrease execution cost.

Furthermore, it would be interesting to see if there is an "optimal" length for the technical indicators given a forecast horizon. The length parameters were somewhat arbitrarily chosen in this thesis, that is, it is not necessarily true that one can simply transfer the length parameters from day trading to high frequency trading. In some sense, the length choice for short-term prediction should reflect the same information flow as it does when applied to daily series. However, this information flow can be difficult to measure or even define.

Appendix A

Additional group importance analysis

A.1 Mid price

The mid price analysis was done for one contract, Dow Jones. The reason for choosing this contract is that it has roughly equal class distribution for the 3 seconds forecast horizon. For the other contracts, the class frequency is skewed at shorter forecast horizons. For example, the t-bond contract has a frequency of around 70% for the neutral class, at the 3 seconds forecast horizon.

This is problematic since it is unreasonable to expect that height of an accuracy for the up and down classes. Therefore, if a feature were to predict increases/decreases in prices, these would yield low/negative feature importance, which is undesirable. The same hold for the group importance.

The mid price is defined by:

$$p_t = \frac{p^a_{t,1} + p^b_{t,1}}{2} \tag{A.1.1}$$

In order to achieve equal class distribution on longer forecast horizons, the quantile approach described in section 5.1.2 was used. Furthermore, the price and certain features were normalized by the Yang-Zhang volatility, as in section 6.1.

Therefore, the group importance for the fast- and slow-acting features were computed for the mid price definition. Moreover, their means and standard deviations were estimated, displayed in figure A.1:

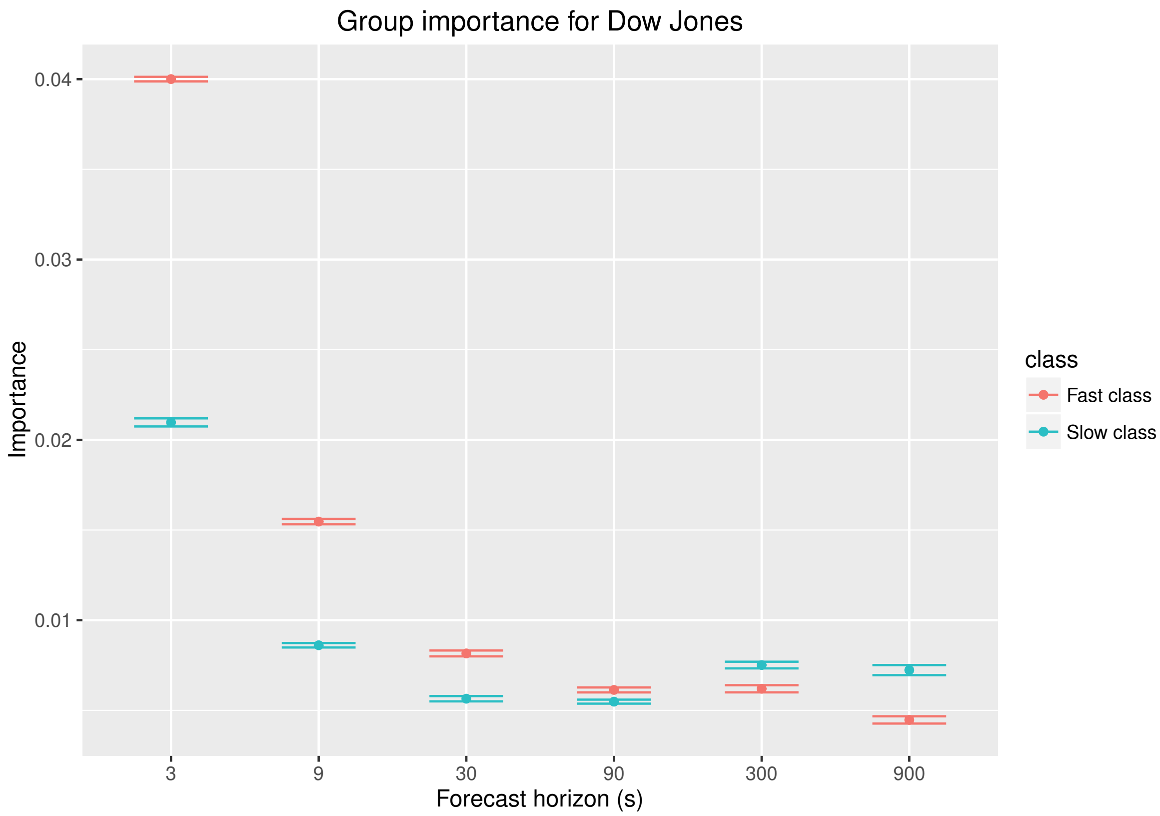


Figure A.1: Group importance for the fast and slow acting features on different forecast horizons with the mid price definition. The contract is Dow Jones.

As one can see from figure A.1, the fast-acting features were clearly better for the shorter forecast horizons. The difference between the two groups shrinks as one increases the forecast horizon. At the 300 and 900 seconds forecast horizon the slow-acting features had a higher group importance than the fast-acting features. The results are fairly similar to the volume weighted price definition for Dow Jones, found in section 6.2.1.

The individual feature importance were computed. Furthermore, the features were ranked, i.e. the feature with the highest feature importance got rank 1, etc. This was displayed in figure A.2 for different forecast horizons:

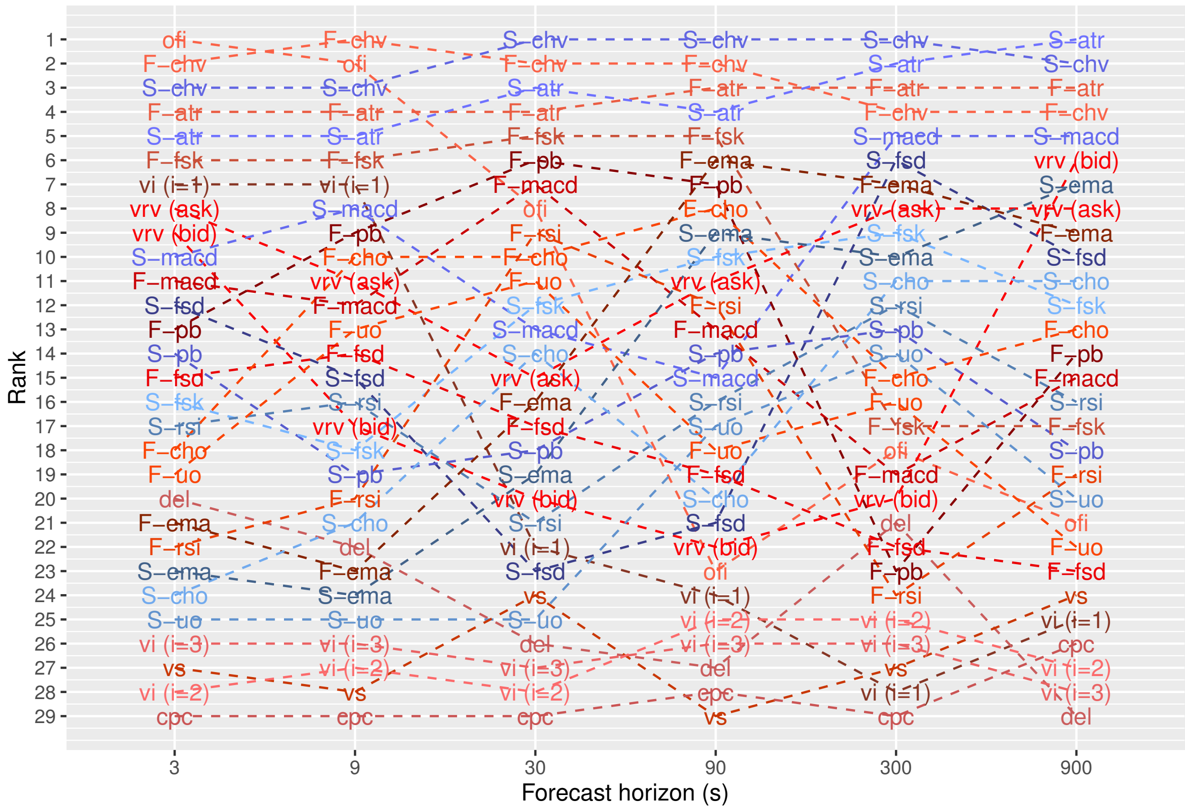


Figure A.2: The ranking of individual features on different forecast horizons with mid price definition. The contract is Dow Jones.

As one can see from figure A.2, the ofi feature was the best for the shortest forecast horizon (3 seconds) and the volatility indicators also performed well. As one increases the forecast horizon the ofi feature importance dropped while the volatility indicators still performed well.

The list of individual feature importance values for the Dow Jones contract with mid price definition can be found in Appendix, section B.3, table B.6.

In general, the results from this analysis are fairly similar to the volume weighted price results for the Dow Jones contract.

Appendix B

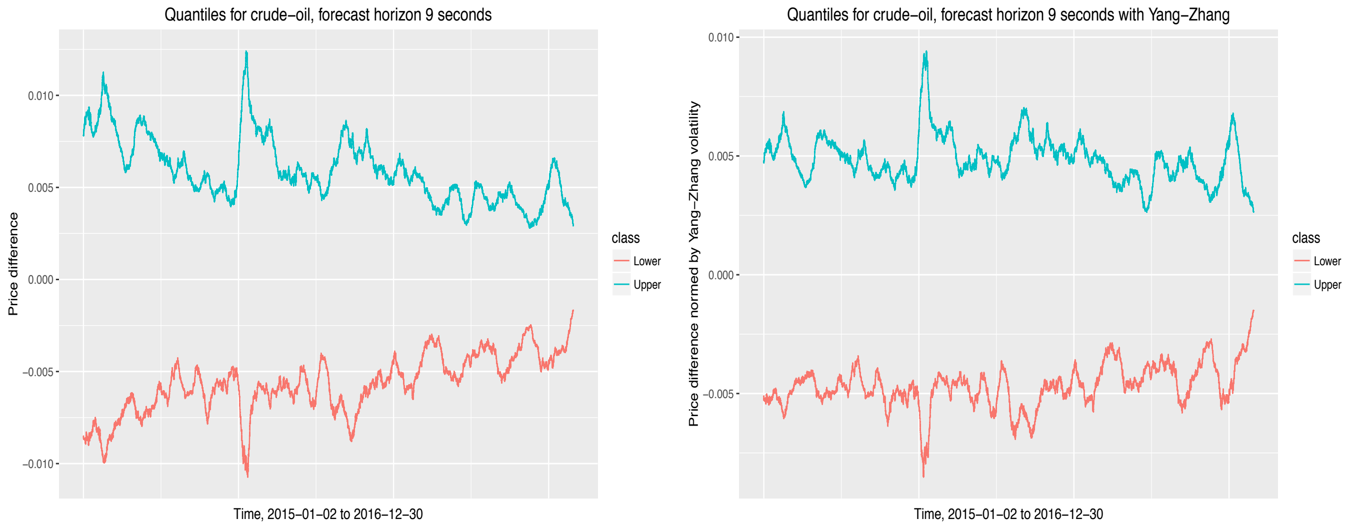
Figures and tables

B.1 Quantiles

In this section the quantiles figures for the crude oil contract are presented:

(a) (b)

Figure B.1: Quantiles for the crude oil contract with 3 seconds forecast horizon on different horizons, unnormalized/normalized with Yang-Zhang volatility.

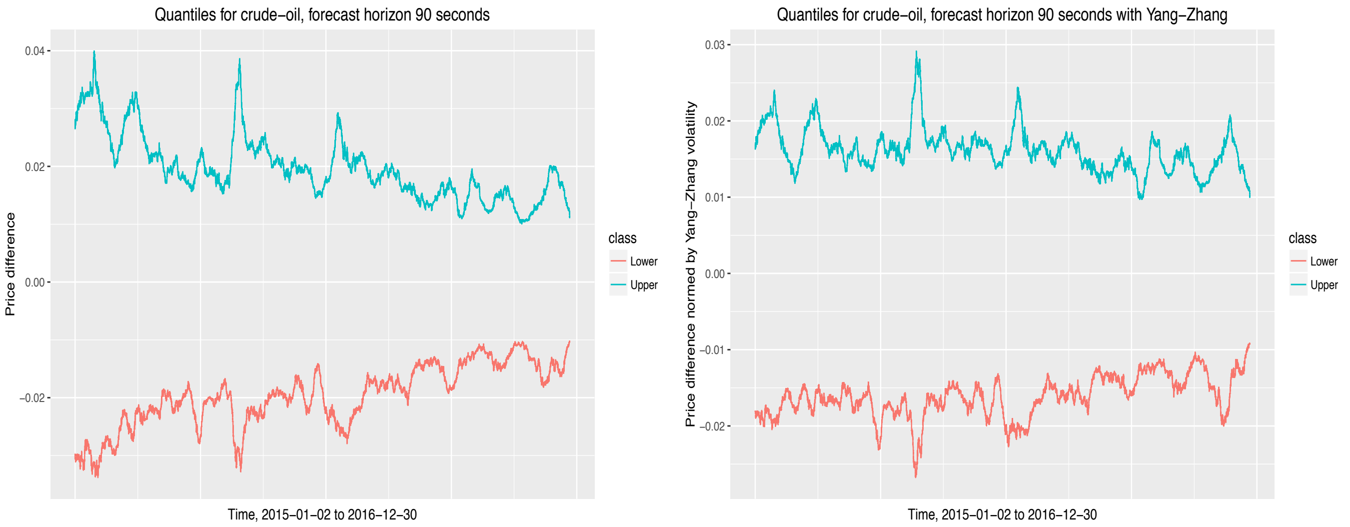


(a) (b)

Figure B.2: Quantiles for the crude oil contract with 9 seconds forecast horizon on different horizons, unnormalized/normalized with Yang-Zhang volatility.

(a) (b)

Figure B.3: Quantiles for the crude oil contract with 30 seconds forecast horizon on different horizons, unnormalized/normalized with Yang-Zhang volatility.

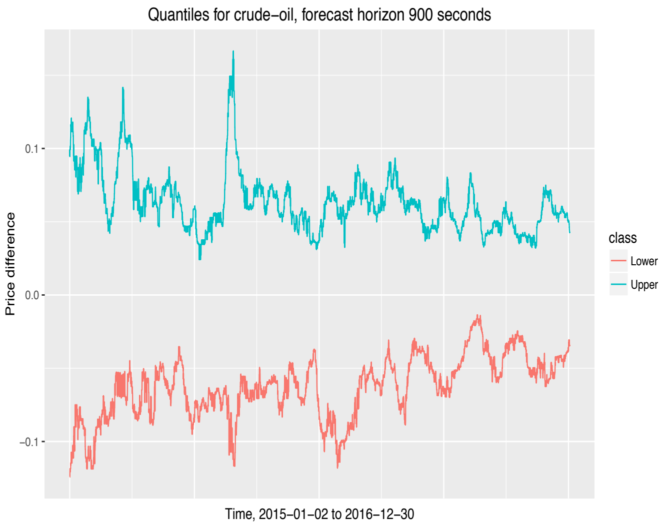


(a) (b)

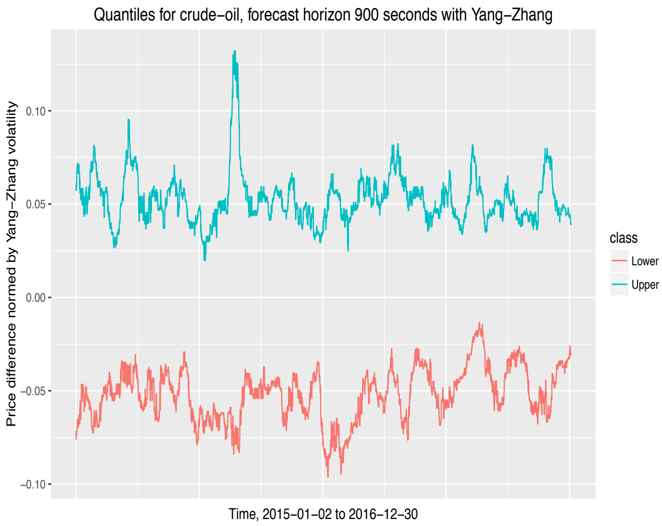
Figure B.4: Quantiles for the crude oil contract with 90 seconds forecast horizon on different horizons, unnormalized/normalized with Yang-Zhang volatility.

(a) (b)

Figure B.5: Quantiles for the crude oil contract with 300 seconds forecast horizon on different horizons, unnormalized/normalized with Yang-Zhang volatility.



(a)



(b)

Figure B.6: Quantiles for the crude oil contract with 900 seconds forecast horizon on different horizons, unnormalized/normalized with Yang-Zhang volatility.

B.2 Histogram example

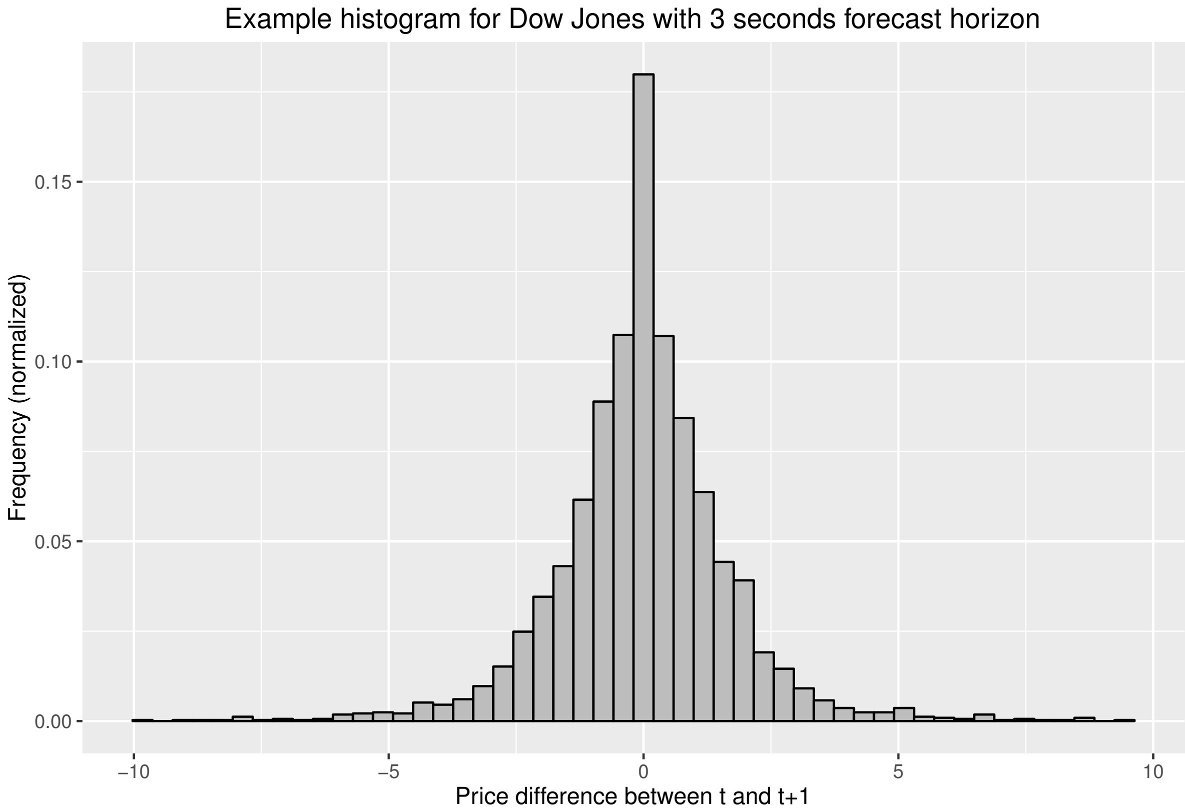


Figure B.7: Example of a histogram for price differences between $t+1$ and t for the 3 seconds forecast horizon. The contract is Dow Jones. Furthermore, the histogram is constructed using two weeks of data, 2015-01-02 till 2015-01-14.

B.3 Feature importance tables

In this section the individual feature importance for all contracts and forecast horizons are presented.

	3 seconds		9 seconds		30 seconds		90 seconds		300 seconds		900 seconds	
Rank	Names	Mean	Names	Mean	Names	Mean	Names	Mean	Names	Mean	Names	Mean
1	ofi	0.01512	F-chv	0.00745	F-chv	0.00468	S-chv	0.00465	S-chv	0.00335	S-chv	0.00274
2	F-chv	0.01054	S-chv	0.00587	S-chv	0.00451	S-atr	0.00318	S-atr	0.00306	S-atr	0.00202
3	F-atr	0.00624	F-atr	0.00509	F-atr	0.00316	F-chv	0.00313	F-atr	0.00160	F-atr	0.00166
4	S-chv	0.00602	ofi	0.00493	S-atr	0.00297	F-atr	0.00261	F-chv	0.00131	F-chv	0.00139
5	S-atr	0.00375	S-atr	0.00394	F-fsk	0.00139	F-fsk	0.00107	S-macd	0.00106	vs	0.00111
6	S-macd	0.00211	S-macd	0.00163	ofi	0.00128	S-macd	0.00097	F-macd	0.00062	S-pb	0.00072
7	S-pb	0.00199	F-fsk	0.00136	F-pb	0.00120	S-rsi	0.00094	S-rsi	0.00062	S-macd	0.00063
8	F-fsk	0.00180	F-pb	0.00124	F-macd	0.00104	F-macd	0.00093	S-pb	0.00058	F-pb	0.00047
9	S-rsi	0.00174	S-pb	0.00114	F-uo	0.00096	F-pb	0.00087	S-fsk	0.00057	S-uo	0.00045
10	S-fsd	0.00169	S-fsd	0.00111	F-cho	0.00091	F-rsi	0.00086	F-cho	0.00044	F-cho	0.00045
11	F-macd	0.00168	F-macd	0.00108	S-macd	0.00088	S-fsk	0.00083	S-fsd	0.00040	F-macd	0.00045
12	S-fsk	0.00161	S-rsi	0.00100	S-fsk	0.00083	F-cho	0.00076	F-ema	0.00040	S-rsi	0.00044
13	F-pb	0.00130	S-fsk	0.00095	S-pb	0.00077	S-cho	0.00073	F-rsi	0.00038	F-uo	0.00043
14	vrv (bid)	0.00127	F-cho	0.00090	F-rsi	0.00077	S-pb	0.00072	F-pb	0.00036	S-cho	0.00043
15	F-fsd	0.00115	F-rsi	0.00083	S-rsi	0.00068	F-fsd	0.00069	S-uo	0.00035	S-fsk	0.00038
16	S-cho	0.00105	S-cho	0.00081	F-fsd	0.00067	ofi	0.00062	vrv (ask)	0.00035	F-fsk	0.00037
17	vrv (ask)	0.00100	F-uo	0.00080	S-uo	0.00058	S-uo	0.00057	S-cho	0.00034	vrv (ask)	0.00037
18	F-uo	0.00094	F-fsd	0.00069	S-cho	0.00057	S-fsd	0.00056	S-ema	0.00034	S-fsd	0.00029
19	F-cho	0.00093	F-ema	0.00067	S-fsd	0.00053	F-uo	0.00055	F-fsk	0.00030	F-ema	0.00026
20	S-uo	0.00091	vrv (bid)	0.00063	F-ema	0.00042	F-ema	0.00041	vrv (bid)	0.00026	vrv (bid)	0.00021
21	F-rsi	0.00089	S-ema	0.00059	S-ema	0.00039	S-ema	0.00036	F-uo	0.00025	F-fsd	0.00020
22	F-ema	0.00083	S-uo	0.00059	vrv (ask)	0.00031	vrv (bid)	0.00028	F-fsd	0.00019	S-ema	0.00020
23	vi (i=1)	0.00074	vrv (ask)	0.00043	vi (i=1)	0.00023	vrv (ask)	0.00027	ofi	0.00017	F-rsi	0.00018
24	S-ema	0.00069	vi (i=1)	0.00029	vi (i=2)	0.00018	vi (i=1)	0.00019	vi (i=3)	0.00011	ofi	0.00010
25	vs	0.00044	vs	0.00022	vrv (bid)	0.00016	del	0.00012	cpc	0.00007	vi (i=3)	0.00009
26	cpc	0.00035	vi (i=2)	0.00018	vs	0.00014	vs	0.00011	vi (i=1)	0.00005	vi (i=2)	0.00004
27	del	0.00032	del	0.00017	del	0.00012	cpc	0.00007	vi (i=2)	0.00004	vi (i=1)	0.00004
28	vi (i=2)	0.00028	cpc	0.00017	vi (i=3)	0.00011	vi (i=2)	0.00006	del	0.00002	cpc	0.00002
29	vi (i=3)	0.00017	vi (i=3)	0.00008	cpc	0.00010	vi (i=3)	0.00003	vs	0.00001	del	-0.00002

Table B.1: The individual feature importance for the crude oil contract on different forecast horizons. The S-index indicates the slower version while the F-index indicates the faster version.

	3 seconds		9 seconds		30 seconds		90 seconds		300 seconds		900 seconds	
Rank	Names	Mean	Names	Mean	Names	Mean	Names	Mean	Names	Mean	Names	Mean
1	ofi	0.01501	F-chv	0.00917	S-chv	0.00688	S-chv	0.00606	S-chv	0.00476	S-chv	0.00402
2	F-chv	0.01057	ofi	0.00767	F-chv	0.00670	F-chv	0.00510	S-atr	0.00439	S-atr	0.00368
3	F-atr	0.00636	S-chv	0.00609	F-atr	0.00510	S-atr	0.00436	F-atr	0.00295	F-atr	0.00145
4	vi (i=1)	0.00552	F-atr	0.00572	S-atr	0.00475	F-atr	0.00430	F-chv	0.00270	vrv (ask)	0.00103
5	S-chv	0.00529	S-atr	0.00419	ofi	0.00263	F-macd	0.00130	S-pb	0.00112	F-chv	0.00094
6	S-atr	0.00377	vi (i=1)	0.00222	vrv (ask)	0.00139	S-macd	0.00099	F-macd	0.00111	S-fsk	0.00087
7	F-fsk	0.00284	vrv (bid)	0.00178	F-macd	0.00138	vrv (bid)	0.00095	S-macd	0.00106	S-pb	0.00080
8	vrv (bid)	0.00234	vrv (ask)	0.00158	vrv (bid)	0.00124	S-cho	0.00091	S-rsi	0.00095	S-macd	0.00071
9	vrv (ask)	0.00228	F-fsk	0.00157	F-cho	0.00120	F-cho	0.00090	S-fsk	0.00085	vrv (bid)	0.00070
10	F-pb	0.00213	F-macd	0.00155	S-macd	0.00106	ofi	0.00081	S-cho	0.00080	S-cho	0.00065
11	F-fsd	0.00164	F-pb	0.00142	S-cho	0.00101	F-fsk	0.00079	F-cho	0.00078	F-macd	0.00062
12	F-uo	0.00153	F-cho	0.00118	vi (i=1)	0.00100	F-pb	0.00078	ofi	0.00073	F-fsk	0.00059
13	F-macd	0.00149	S-fsk	0.00105	F-fsk	0.00090	S-fsk	0.00077	vrv (bid)	0.00062	vs	0.00057
14	F-rsi	0.00142	S-pb	0.00101	F-pb	0.00090	F-rsi	0.00073	F-fsk	0.00062	S-rsi	0.00055
15	F-cho	0.00142	F-rsi	0.00100	S-pb	0.00087	S-pb	0.00072	F-pb	0.00061	F-cho	0.00046
16	S-fsk	0.00129	F-fsd	0.00099	S-fsk	0.00078	vrv (ask)	0.00070	vrv (ask)	0.00054	F-rsi	0.00046
17	S-pb	0.00118	S-cho	0.00099	F-rsi	0.00070	F-fsd	0.00067	S-fsd	0.00054	F-pb	0.00043
18	del	0.00101	S-macd	0.00091	F-fsd	0.00062	S-rsi	0.00064	F-rsi	0.00051	S-uo	0.00042
19	S-macd	0.00095	F-uo	0.00088	S-rsi	0.00061	S-fsd	0.00054	F-uo	0.00050	S-fsd	0.00039
20	S-rsi	0.00090	S-rsi	0.00082	S-fsd	0.00059	vi (i=1)	0.00050	F-fsd	0.00045	F-fsd	0.00033
21	S-cho	0.00089	del	0.00073	F-uo	0.00055	F-uo	0.00047	S-uo	0.00040	ofi	0.00033
22	S-fsd	0.00088	S-fsd	0.00073	del	0.00045	S-uo	0.00037	F-ema	0.00033	F-uo	0.00025
23	S-uo	0.00075	vs	0.00060	S-uo	0.00044	F-ema	0.00033	S-ema	0.00030	F-ema	0.00024
24	F-ema	0.00042	S-uo	0.00058	vs	0.00040	S-ema	0.00031	del	0.00026	S-ema	0.00021
25	S-ema	0.00038	F-ema	0.00041	F-ema	0.00036	del	0.00030	vs	0.00026	vi (i=3)	0.00011
26	vi (i=2)	0.00029	S-ema	0.00038	S-ema	0.00033	vs	0.00026	vi (i=1)	0.00024	cpc	0.00007
27	vi (i=3)	0.00021	vi (i=3)	0.00026	vi (i=3)	0.00017	vi (i=3)	0.00026	cpc	0.00004	del	0.00000
28	cpc	0.00019	vi (i=2)	0.00024	vi (i=2)	0.00015	vi (i=2)	0.00014	vi (i=2)	0.00003	vi (i=1)	-0.00003
29	vs	0.00019	cpc	0.00015	cpc	0.00010	cpc	0.00005	vi (i=3)	-0.00000	vi (i=2)	-0.00012

Table B.2: The individual feature importance for the GPB contract on different forecast horizons. The S-index indicates the slower version while the F-index indicates the faster version.

	3 seconds		9 seconds		30 seconds		90 seconds		300 seconds		900 seconds	
Rank	Names	Mean	Names	Mean	Names	Mean	Names	Mean	Names	Mean	Names	Mean
1	ofi	0.02043	ofi	0.01234	F-chv	0.00642	S-chv	0.00570	S-chv	0.00317	S-chv	0.00197
2	F-chv	0.01150	F-chv	0.01021	S-chv	0.00622	S-atr	0.00454	S-atr	0.00293	S-atr	0.00170
3	S-chv	0.00612	S-chv	0.00591	ofi	0.00419	F-chv	0.00289	F-chv	0.00117	F-chv	0.00087
4	F-atr	0.00580	F-atr	0.00503	F-atr	0.00412	F-atr	0.00221	F-atr	0.00085	vrv (bid)	0.00076
5	vi (i=1)	0.00423	S-atr	0.00374	S-atr	0.00379	S-macd	0.00091	vrv (bid)	0.00069	F-atr	0.00074
6	S-atr	0.00395	vi (i=1)	0.00317	vi (i=1)	0.00123	F-macd	0.00082	vrv (ask)	0.00053	S-macd	0.00052
7	F-fsk	0.00284	F-fsk	0.00202	F-macd	0.00111	F-fsk	0.00065	ofi	0.00049	F-macd	0.00040
8	F-fsd	0.00237	F-fsd	0.00176	vrv (ask)	0.00110	ofi	0.00062	F-macd	0.00048	S-rsi	0.00032
9	vrv (ask)	0.00216	vrv (bid)	0.00166	F-fsk	0.00108	vrv (bid)	0.00061	F-fsk	0.00038	vi (i=1)	0.00029
10	vrv (bid)	0.00195	vrv (ask)	0.00162	vrv (bid)	0.00108	S-fsk	0.00061	F-cho	0.00033	F-pb	0.00023
11	S-pb	0.00189	S-fsk	0.00152	S-fsk	0.00100	S-cho	0.00058	S-macd	0.00032	S-pb	0.00023
12	S-fsk	0.00188	F-macd	0.00148	S-macd	0.00097	F-pb	0.00057	del	0.00031	F-cho	0.00020
13	F-macd	0.00184	S-pb	0.00141	S-pb	0.00096	vrv (ask)	0.00055	F-uo	0.00031	vrv (ask)	0.00019
14	F-pb	0.00176	F-cho	0.00130	F-fsd	0.00090	S-pb	0.00055	vi (i=1)	0.00030	S-cho	0.00018
15	F-uo	0.00175	F-uo	0.00129	F-cho	0.00088	F-fsd	0.00054	S-rsi	0.00028	F-uo	0.00017
16	S-fsd	0.00168	S-rsi	0.00126	F-uo	0.00085	S-fsd	0.00050	S-fsk	0.00028	S-fsk	0.00017
17	S-rsi	0.00166	F-pb	0.00124	S-rsi	0.00081	F-cho	0.00049	vi (i=2)	0.00026	vi (i=2)	0.00016
18	F-cho	0.00158	S-fsd	0.00119	S-cho	0.00074	F-uo	0.00044	F-pb	0.00022	ofi	0.00013
19	S-macd	0.00146	S-macd	0.00114	F-pb	0.00073	S-rsi	0.00041	S-pb	0.00022	F-rsi	0.00009
20	F-rsi	0.00139	F-rsi	0.00106	S-fsd	0.00073	F-rsi	0.00040	S-cho	0.00021	F-fsk	0.00008
21	S-uo	0.00130	S-uo	0.00105	S-uo	0.00071	S-uo	0.00034	F-fsd	0.00021	S-fsd	0.00008
22	S-cho	0.00121	S-cho	0.00100	F-rsi	0.00069	F-ema	0.00029	S-uo	0.00021	del	0.00004
23	del	0.00084	del	0.00067	F-ema	0.00038	S-ema	0.00026	F-rsi	0.00020	vi (i=3)	-0.00003
24	F-ema	0.00068	F-ema	0.00053	del	0.00037	vi (i=1)	0.00021	S-fsd	0.00018	vs	-0.00004
25	S-ema	0.00063	S-ema	0.00049	S-ema	0.00035	del	0.00015	S-ema	0.00015	F-ema	-0.00004
26	vs	0.00039	vs	0.00033	vs	0.00021	vs	0.00011	F-ema	0.00015	cpc	-0.00005
27	vi (i=2)	0.00028	vi (i=2)	0.00031	vi (i=2)	0.00018	vi (i=3)	0.00009	cpc	-0.00006	S-ema	-0.00007
28	vi (i=3)	0.00024	vi (i=3)	0.00024	vi (i=3)	0.00012	vi (i=2)	0.00003	vs	-0.00007	S-uo	-0.00010
29	cpc	-0.00001	cpc	0.00000	cpc	-0.00000	cpc	-0.00001	vi (i=3)	-0.00012	F-fsd	-0.00011

Table B.3: The individual feature importance for the t-bond contract on different forecast horizons. The S-index indicates the slower version while the F-index indicates the faster version.

	3 seconds		9 seconds		30 seconds		90 seconds		300 seconds		900 seconds	
Rank	Names	Mean	Names	Mean	Names	Mean	Names	Mean	Names	Mean	Names	Mean
1	ofi	0.01343	F-chv	0.00710	S-chv	0.00497	S-chv	0.00456	S-chv	0.00464	S-atr	0.00335
2	F-chv	0.00912	S-chv	0.00570	F-chv	0.00467	F-chv	0.00369	S-atr	0.00390	S-chv	0.00301
3	S-chv	0.00592	ofi	0.00511	S-atr	0.00363	S-atr	0.00334	F-atr	0.00272	F-atr	0.00198
4	F-atr	0.00581	F-atr	0.00500	F-atr	0.00318	F-atr	0.00279	F-chv	0.00253	F-chv	0.00139
5	S-atr	0.00439	S-atr	0.00395	F-fsk	0.00148	F-ema	0.00100	S-macd	0.00111	S-macd	0.00106
6	F-fsk	0.00243	F-fsk	0.00189	F-macd	0.00127	S-macd	0.00092	F-ema	0.00086	S-ema	0.00077
7	S-macd	0.00225	S-macd	0.00167	F-pb	0.00126	S-ema	0.00088	S-ema	0.00080	F-ema	0.00074
8	S-pb	0.00207	F-cho	0.00154	ofi	0.00123	F-macd	0.00080	vrv (ask)	0.00071	vrv (ask)	0.00073
9	S-fsd	0.00183	F-pb	0.00135	S-macd	0.00112	F-fsk	0.00079	F-cho	0.00069	S-cho	0.00067
10	F-cho	0.00178	F-macd	0.00116	F-rsi	0.00096	F-cho	0.00076	S-fsk	0.00069	vrv (bid)	0.00063
11	S-rsi	0.00175	S-rsi	0.00113	F-cho	0.00093	F-pb	0.00075	S-rsi	0.00069	S-fsk	0.00054
12	S-fsk	0.00169	S-pb	0.00112	F-ema	0.00091	S-fsk	0.00075	S-fsd	0.00068	S-fsd	0.00053
13	F-pb	0.00162	S-fsd	0.00112	S-ema	0.00085	vrv (ask)	0.00073	S-cho	0.00059	F-macd	0.00052
14	F-macd	0.00158	S-fsk	0.00107	S-fsk	0.00084	S-rsi	0.00068	S-pb	0.00058	F-pb	0.00050
15	F-fsd	0.00147	F-fsd	0.00103	F-fsd	0.00083	S-pb	0.00068	F-macd	0.00055	S-rsi	0.00041
16	S-cho	0.00127	F-ema	0.00099	vrv (ask)	0.00080	F-rsi	0.00064	F-uo	0.00049	S-uo	0.00038
17	F-uo	0.00122	S-cho	0.00095	F-uo	0.00076	F-fsd	0.00059	vrv (bid)	0.00047	F-fsk	0.00036
18	F-rsi	0.00115	S-ema	0.00094	S-pb	0.00071	F-uo	0.00055	F-fsk	0.00044	F-rsi	0.00036
19	vrv (ask)	0.00107	F-rsi	0.00094	S-rsi	0.00065	S-cho	0.00055	F-fsd	0.00040	F-fsd	0.00033
20	F-ema	0.00106	F-uo	0.00090	S-cho	0.00059	S-fsd	0.00053	S-uo	0.00034	F-uo	0.00028
21	S-uo	0.00102	vrv (bid)	0.00085	vrv (bid)	0.00053	S-uo	0.00048	ofi	0.00032	F-cho	0.00023
22	vrv (bid)	0.00102	vrv (ask)	0.00083	S-uo	0.00051	vrv (bid)	0.00045	F-pb	0.00032	S-pb	0.00022
23	S-ema	0.00093	S-uo	0.00077	S-fsd	0.00050	ofi	0.00037	del	0.00032	vi (i=1)	0.00019
24	vi (i=3)	0.00055	del	0.00037	del	0.00023	vi (i=3)	0.00015	F-rsi	0.00028	ofi	0.00018
25	del	0.00050	vs	0.00034	vi (i=3)	0.00016	vi (i=1)	0.00014	vi (i=2)	0.00022	vs	0.00009
26	vi (i=1)	0.00048	vi (i=3)	0.00031	vi (i=1)	0.00012	vi (i=2)	0.00013	vi (i=3)	0.00017	del	0.00006
27	vs	0.00047	vi (i=1)	0.00030	vi (i=2)	0.00012	del	0.00013	vi (i=1)	0.00010	cpc	0.00003
28	vi (i=2)	0.00032	vi (i=2)	0.00018	vs	0.00010	vs	0.00010	vs	0.00005	vi (i=3)	-0.00003
29	cpc	0.00001	cpc	0.00001	cpc	-0.00000	cpc	-0.00000	cpc	-0.00003	vi (i=2)	-0.00004

Table B.4: The individual feature importance for the Dow Jones contract on different forecast horizons. The S-index indicates the slower version while the F-index indicates the faster version.

	3 seconds		9 seconds		30 seconds		90 seconds		300 seconds		900 seconds	
1	ofi	0.01849	F-chv	0.00949	F-chv	0.00644	S-chv	0.00562	S-chv	0.00378	S-chv	0.00244
2	F-chv	0.01128	ofi	0.00877	S-chv	0.00599	S-atr	0.00489	S-atr	0.00372	S-atr	0.00216
3	F-atr	0.00733	S-chv	0.00640	F-atr	0.00500	F-chv	0.00461	F-atr	0.00259	F-atr	0.00181
4	S-chv	0.00729	F-atr	0.00630	S-atr	0.00484	F-atr	0.00375	F-chv	0.00239	F-chv	0.00146
5	S-atr	0.00568	S-atr	0.00521	ofi	0.00301	S-macd	0.00125	S-macd	0.00129	F-macd	0.00097
6	F-fsk	0.00325	F-fsk	0.00214	F-macd	0.00150	F-macd	0.00119	vrv (ask)	0.00101	S-macd	0.00077
7	vi (i=1)	0.00268	vrv (bid)	0.00173	F-fsk	0.00145	F-fsk	0.00092	F-macd	0.00085	vrv (ask)	0.00073
8	vrv (ask)	0.00258	F-macd	0.00160	F-cho	0.00125	F-cho	0.00092	vrv (bid)	0.00081	vrv (bid)	0.00070
9	vrv (bid)	0.00254	vrv (ask)	0.00153	F-pb	0.00122	vrv (bid)	0.00091	S-rsi	0.00075	S-pb	0.00061
10	S-macd	0.00204	S-macd	0.00152	vrv (ask)	0.00121	vrv (ask)	0.00089	S-fsk	0.00072	vs	0.00058
11	F-fsd	0.00198	F-cho	0.00149	S-macd	0.00119	S-cho	0.00087	S-pb	0.00069	S-fsk	0.00053
12	F-macd	0.00193	F-pb	0.00138	vrv (bid)	0.00117	ofi	0.00085	S-cho	0.00062	F-cho	0.00043
13	S-pb	0.00193	vi (i=1)	0.00123	S-fsk	0.00086	F-pb	0.00085	ofi	0.00057	F-pb	0.00040
14	F-pb	0.00192	F-fsd	0.00120	F-uo	0.00086	S-fsk	0.00080	F-cho	0.00055	S-fsd	0.00036
15	S-fsd	0.00170	S-pb	0.00118	S-cho	0.00084	F-rsi	0.00070	S-fsd	0.00049	F-uo	0.00035
16	F-cho	0.00169	S-fsd	0.00113	F-fsd	0.00083	S-pb	0.00069	vs	0.00049	S-cho	0.00035
17	S-rsi	0.00169	S-fsk	0.00113	F-rsi	0.00083	S-rsi	0.00069	F-fsk	0.00047	F-fsk	0.00035
18	S-fsk	0.00167	S-rsi	0.00106	S-pb	0.00077	F-fsd	0.00066	F-ema	0.00046	S-rsi	0.00030
19	F-uo	0.00150	F-uo	0.00104	S-rsi	0.00063	vs	0.00055	F-uo	0.00040	S-uo	0.00029
20	F-rsi	0.00135	S-cho	0.00102	F-ema	0.00058	F-ema	0.00055	F-pb	0.00040	F-ema	0.00026
21	S-cho	0.00128	F-rsi	0.00100	S-ema	0.00057	F-uo	0.00054	S-ema	0.00036	S-ema	0.00024
22	S-uo	0.00103	S-uo	0.00075	S-fsd	0.00056	S-fsd	0.00053	F-rsi	0.00030	F-fsd	0.00021
23	vs	0.00088	F-ema	0.00071	S-uo	0.00055	S-ema	0.00049	F-fsd	0.00028	F-rsi	0.00020
24	F-ema	0.00079	S-ema	0.00062	vs	0.00055	S-uo	0.00048	S-uo	0.00028	ofi	0.00019
25	del	0.00073	vs	0.00062	vi (i=1)	0.00050	vi (i=1)	0.00029	del	0.00022	vi (i=1)	0.00004
26	S-ema	0.00070	del	0.00049	del	0.00029	del	0.00019	vi (i=1)	0.00019	vi (i=3)	0.00000
27	vi (i=2)	0.00036	vi (i=3)	0.00025	vi (i=2)	0.00017	vi (i=2)	0.00011	vi (i=2)	0.00016	cpc	-0.00000
28	vi (i=3)	0.00027	vi (i=2)	0.00023	vi (i=3)	0.00014	vi (i=3)	0.00011	vi (i=3)	0.00007	del	-0.00003
29	cpc	0.00011	cpc	0.00010	cpc	0.00007	cpc	0.00003	cpc	0.00001	vi (i=2)	-0.00005

Table B.5: The individual feature importance for the multivariate model on different forecast horizons. The S-index indicates the slower version while the F-index indicates the faster version.

	3 seconds		9 seconds		30 seconds		90 seconds		300 seconds		900 seconds	
1	ofi	0.01479	F-chv	0.00514	S-chv	0.00288	S-chv	0.00264	S-chv	0.00405	S-atr	0.00357
2	F-chv	0.01325	ofi	0.00444	F-chv	0.00265	F-chv	0.00216	S-atr	0.00308	S-chv	0.00345
3	S-chv	0.00967	S-chv	0.00393	S-atr	0.00201	F-atr	0.00201	F-atr	0.00238	F-atr	0.00159
4	F-atr	0.00841	F-atr	0.00356	F-atr	0.00199	S-atr	0.00178	F-chv	0.00183	F-chv	0.00118
5	S-atr	0.00747	S-atr	0.00283	F-fsk	0.00149	F-fsk	0.00079	S-macd	0.00095	S-macd	0.00111
6	F-fsk	0.00547	F-fsk	0.00217	F-pb	0.00110	F-ema	0.00071	S-fsd	0.00068	vrv (bid)	0.00083
7	vi (i=1)	0.00424	vi (i=1)	0.00170	F-macd	0.00071	F-pb	0.00064	F-ema	0.00067	S-ema	0.00076
8	vrv (ask)	0.00374	S-macd	0.00132	ofi	0.00069	F-cho	0.00063	vrv (ask)	0.00065	vrv (ask)	0.00061
9	vrv (bid)	0.00353	F-pb	0.00125	F-rsi	0.00066	S-ema	0.00058	S-fsk	0.00062	F-ema	0.00060
10	S-macd	0.00276	F-cho	0.00116	F-cho	0.00062	S-fsk	0.00057	S-ema	0.00060	S-fsd	0.00050
11	F-macd	0.00256	vrv (ask)	0.00096	F-uo	0.00056	vrv (ask)	0.00053	S-cho	0.00056	S-cho	0.00046
12	S-fsd	0.00250	F-macd	0.00094	S-fsk	0.00056	F-rsi	0.00052	S-rsi	0.00049	S-fsk	0.00045
13	F-pb	0.00247	F-uo	0.00092	S-macd	0.00055	F-macd	0.00047	S-pb	0.00042	F-cho	0.00043
14	S-pb	0.00246	F-fsd	0.00089	S-cho	0.00050	S-pb	0.00046	S-uo	0.00041	F-pb	0.00040
15	F-fsd	0.00242	S-fsd	0.00082	vrv (ask)	0.00048	S-macd	0.00043	F-cho	0.00040	F-macd	0.00037
16	S-fsk	0.00227	S-rsi	0.00081	F-ema	0.00045	S-rsi	0.00043	F-uo	0.00039	S-rsi	0.00036
17	S-rsi	0.00225	vrv (bid)	0.00080	F-fsd	0.00045	S-uo	0.00041	F-fsk	0.00038	F-fsk	0.00029
18	F-cho	0.00191	S-fsk	0.00077	S-pb	0.00045	F-uo	0.00041	ofi	0.00031	S-pb	0.00029
19	F-uo	0.00171	S-pb	0.00075	S-ema	0.00043	F-fsd	0.00038	F-macd	0.00030	F-rsi	0.00029
20	del	0.00159	F-rsi	0.00071	vrv (bid)	0.00041	S-cho	0.00034	vrv (bid)	0.00027	S-uo	0.00026
21	F-ema	0.00145	S-cho	0.00057	S-rsi	0.00035	S-fsd	0.00032	del	0.00023	ofi	0.00025
22	F-rsi	0.00145	del	0.00053	vi (i=1)	0.00028	vrv (bid)	0.00027	F-fsd	0.00023	F-uo	0.00020
23	S-ema	0.00135	F-ema	0.00051	S-fsd	0.00028	ofi	0.00027	F-pb	0.00021	F-fsd	0.00017
24	S-cho	0.00131	S-ema	0.00044	vs	0.00027	vi (i=1)	0.00027	F-rsi	0.00019	vs	0.00017
25	S-uo	0.00120	S-uo	0.00043	S-uo	0.00023	vi (i=2)	0.00016	vi (i=2)	0.00016	vi (i=1)	0.00014
26	vi (i=3)	0.00119	vi (i=3)	0.00030	del	0.00016	vi (i=3)	0.00012	vi (i=3)	0.00007	cpc	0.00003
27	vs	0.00107	vi (i=2)	0.00028	vi (i=3)	0.00014	del	0.00012	vs	0.00006	vi (i=2)	-0.00010
28	vi (i=2)	0.00081	vs	0.00022	vi (i=2)	0.00007	cpc	-0.00001	vi (i=1)	0.00000	vi (i=3)	-0.00012
29	cpc	0.00001	cpc	0.00002	cpc	0.00000	vs	-0.00003	cpc	-0.00001	del	-0.00012

Table B.6: The individual feature importance for the Dow Jones contract on different forecast horizons with the mid price definition. The S-index indicates the slower version while the F-index indicates the faster version.

Bibliography

[1] Abergel, Frédéric et al. *Limit Order Books*. Cambridge University Press, 2016.

[2] Achelis, Steven. *Technical Analysis from A to Z*. 2:th. McGraw Hill Professional, 2000.

[3] Aldridge, Irene and Krawciw, Steven. *Real-Time Risk: What Investors Should Know About FinTech, High-Frequency Trading, and Flash Crashes*. 1:th. Wiley, 2017.

[4] Appel, Gerald. *Technical Analysis: Power Tools for Active Investors*. 1:th. Financial Times/Prentice Hall, 2005.

[5] Baranauskas, Josè, Oshiro, Thais, and Perez, Pedro. "How Many Trees in a Random Forest?" In: *Machine Learning and Data Mining in Pattern Recognition*. Lecture Notes in Computer Science. Springer, 2012.

[6] Biggs, Norman. *Discrete Mathematics*. OUP Oxford, 2002.

[7] Bollinger, John. *Bollinger on Bollinger Bands*. 1:th. McGraw-Hill Education, 2001.

[8] Booth, Ash. "Automated Algorithmic Trading: Machine Learning and Agent-based Modelling in Complex Adaptive Financial Markets". PhD thesis. University of Southampton, Apr. 2016.

[9] Breiman, Leo. *Classification and Regression Trees*. Chapman and Hall, 1984.

[10] Brockwell, Peter and Davis, Richard. *Time Series: Theory and Methods*. 2:th. Springer, 2009.

[11] Caruana, Rich and Niculescu-Mizil, Alexandru. *An Empirical Comparison of Supervised Learning Algorithms*. Department of Computer Science, Cornell University. 2006.

[12] Chen, Hao et al. "A double-layer neural network framework for high frequency forecasting". In: *ACM Transactions on Management Information Systems* 7.4 (2017).

[13] Chickering, Max, Meek, Chris, and Rounthwaite, Robert. "Efficient Determination of Dynamic Split Points in a Decision Tree". In: *In Proceedings of the 2001 IEEE International Conference on Data Mining*. IEEE, 2001.

[14] Chou, Ray, Chou, Hengchih, and Liu, Nathan. *Range Volatility Models and Their Applications in Finance*. Springer US, 2010.

[15] CME-G. *CME Group to Close Most Open Outcry Futures Trading in Chicago and New York*. Feb. 2015.

[16] Cont, Rama, Kukanov, Arseniy, and Stoikov, Sasha. "The Price Impact of Order Book Events". In: *Journal of financial econmetrics* 12.1 (2011).

[17] Creamer, Germán. "Model calibration and automated trading agent for Euro futures". In: *Quantitative Finance* 12.4 (2012).

[18] Dietterich, T.G. “Four current directions”. In: *Machine learning research* 14 (1997).

[19] Elkan, Charles. “The foundations of cost-sensitive learning”. In: *Proceedings of the 17th international joint conference on Artificial intelligence*. Morgan Kaufmann Publishers Inc, 2001.

[20] Essen, Brian Van, Macaraeg, Chris, and Livermore, Lawrence. “Accelerating a Random Forest Classifier: Multi-Core, GP-GPU, or FPGA?” In: *Field-Programmable Custom Computing Machines (FCCM), 2012 IEEE 20th Annual International Symposium* (July 2012).

[21] Folger, Jean and Leibfarth, Lee. *Make Money Trading: How to Build a Winning Trading Business*. Marketplace Books, 2007.

[22] Grabocka, Josif. “Invariant Features For Time-Series Classification”. PhD thesis. University of Hildesheim, Germany, 2015.

[23] Gregorutti, Baptiste, Michel, Bertrand, and Saint-Pierre, Philippe. “Correlation and variable importance in random forests”. In: *Statistics and Computing* 27 (2017), pp. 659–678.

[24] Gregorutti, Baptiste, Michel, Bertrand, and Saint-Pierre, Philippe. “Grouped variable importance with random forests and application to multiple functional data analysis”. In: *Computational Statistics and Data Analysis* 90 (2015), pp. 15–35.

[25] Hastie, Trevor, Tibshirani, Robert, and Friedman, Jerome. *The Elements of Statistical Learning*. 2:th. Springer, 2009.

[26] Hull, John. *Options, Futures and Other Derivatives*. 8:th. Pearson Education, Apr. 2011.

[27] Hyndman, Rob and Fan, Yanan. “Sample Quantiles in Statistical Packages”. In: *The American Statistician* 50 (1996).

[28] Janitza, Silke, Celik, Ender, and Boulesteix, Anne-Laure. “A computationally fast variable importance test for random forests for high-dimensional data”. In: *Advances in Data Analysis and Classification* 10 (2016).

[29] Kearns, Michael and Nevmyvaka, Yuriy. *Machine Learning for Market Microstructure and High Frequency Trading*. Department of Computer and Information Science, University of Pennsylvania.

[30] Kenney, John. *Mathematics of statistics*. 3:th. Van Nostrand, 1964.

[31] Kercheval, Alec and Zhang, Yuan. “Modeling high frequency limit order book dynamics with support vector machines”. In: *Quantitative Finance* 15.8 (2015).

[32] Lane, George. “Lane’s Stochastics”. In: *Commodities magazine* (June 1984), pp. 87–90.

[33] Lipton, Alexander, Pesavento, Umberto, and Sotiropoulos, Michael G. “Trade arrival dynamics and quote imbalance in a limit order book”. In: *ArXiv e-prints* (Dec. 2013). arXiv: `1312.0514 [q-fin.TR]`.

[34] Loh, Wei-Yin. “Classification and regression trees”. In: *Data Mining and Knowledge Discovery* 1 (2011).

[35] Maimon, Oded and Rokach, Lior. *Data Mining with Decision Trees: Theory and Applications*. 1:th. World Scientific Publishing Company, 2008.

[36] Murphy, John. *Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. 2:th. New York Institute of Finance, Dec. 1998.

[37] Paddrik, Mark et al. “Effects of limit order book information level on market stability metrics”. In: *Journal of Economic Interaction and Coordination* 12 (2015).

[38] Rechenthin, Michael and Street, Nick. “Using conditional probability to identify trends in intra-day high-frequency equity pricing”. In: *Physica A: Statistical Mechanics and its Applications* 392.14 (2013).

[39] Sona, Youngdoo, Nohb, Dong-jin, and Leea, Jaewook. “Forecasting trends of high-frequency KOSPI200 index data using learning classifiers”. In: *Expert Systems with Applications* 39.14 (2012).

[40] Tibshirani, Robert et al. *An Introduction to Statistical Learning*. 1:th. Springer-Verlag New York, 2013.

[41] Wilder, Welles. *New Concepts in Technical Trading Systems*. 1:th. Trend Research, 1978.

[42] Wright, Marvin N. and Ziegler, Andreas. *ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R*. 2015.

[43] Zheng, B., Moulines, E., and Abergel, F. “Price Jump Prediction in a Limit Order Book”. In: *Journal of Mathematical Finance* 3.12 (2013).

TRITA -MAT-E 2017:24
ISRN -KTH/MAT/E--17/24--SE