Norwegian School of Economics
Bergen, Spring 2022

# Portfolio Modelling: Man vs Machine

*An empirical study of machine learning efficiency and variable selection in portfolio modelling on the Oslo Stock Exchange*

**Nicholas de Flon & Sander Nikolai Berge Pettersen**

**Supervisor: Carsten Gero Bienz**

Master thesis, Economics and Business Administration

Major: Financial Economics

NORWEGIAN SCHOOL OF ECONOMICS

# Acknowledgements

This thesis was written in the spring of 2022, and marks the final work in our master of science degree in Economics and Business Administration from NHH. A degree we are nothing short of proud of.

We would first like to thank our supervisor, Associate Professor Carsten Gero Bienz. His feedback, advice and different perspectives has been invaluable in the completion of this thesis. Further, we would like to thank Kellis Akselsen from SNF and research scholar Simon Flatebø Selle for expediant service in the requests for accounting data.

We are grateful to family and friends for their feedback, support and interest throughout this spring.

This thesis has been very interesting to write and work on. We have learned a great deal about machine learning and have had the chance to further our coding abilities, often accompanied with great frustration. So finally, we would like to thank each other for the mutual support and motivation throughout the entirety of this 5-year degree.

Norwegian School of Economics

Bergen, May 2022

Nicholas de Flon

Sander Nikolai Berge Pettersen

# Abstract

We investigated and compared the performance of machine learning methods in the context of empirical asset pricing. We used seven different algorithms and 83 firm characteristics, comparing the models' monthly predictive accuracy and variable importance on Norwegian stock and accounting data. Additionally, we investigated the models' ability to generate excess returns in monthly-rebalanced, long-short and long-only portfolios.

We found that the XGBoost algorithm has the highest prediction accuracy of 53.16%, and that it more heavily weights momentum variables. Furthermore, we found excess risk-adjusted returns when constructing portfolios free of market frictions. A long-only portfolio with predictions from the XGBoost model outperformed the index, on average, by around 0.5% each month in the out-of-sample period. When accounting for market frictions an institutional investor might encounter, the returns are diminished to the point of significantly underperforming. When presenting a strategy that a retail investor could implement, we found excess returns. The XGBoost model's net returns outperformed the index by 0.16% and 0.67% over the period, after excluding the largest 25% and 50% firms, respectively. Upon investigating the explanation for this possible market inefficiency, we found that the returns are largely driven by highly illiquid stocks. We suggest that these returns likely are unattainable because of the high degree of illiquidity, and therefore could be impossible to arbitrage away in the way we would expect the market to do when it discovers an inefficiency. We call this phenomenon "rainbow-returns", as they are likely only observable and unattainable.

Our findings support the efficient market hypothesis, in that one cannot beat the market using public available information, and adds to existing literature in the emerging field of empirical asset pricing through machine learning.

***Keywords*** – Machine Learning, Finance, Classification, Variable Importance, Portfolio Modelling

# Contents

# List of Figures

# List of Tables

# 1 Introduction

The fundamental goal in the world of empirical asset pricing, is to understand the behaviour of risk premiums (Gu et al., 2020). With this as its most cardinal objective, academics have provided theories that seeks to give explanations to the variations in returns. Because, if you understand the behaviour of something, you can make predictions. And successful predictions in the stock market can be very lucrative.

This thesis seeks to predict the Norwegian stock market with publicly available information, through the implementation of machine learning algorithms. With the aim to identify the best performing algorithm, the most important predictors and portfolio modelling capabilities. We model 7 different machine learning classification algorithms on 83 firm characteristics, and find that a boosted decision tree model has the highest prediction accuracy. Aggregated importance for all the models showed that firm characteristics based on risk measures, had been the most weighted. However, when giving the higher performing more weight, the momentum factors are more important. The portfolios generated by the algorithms show excess returns before considering market frictions, but not after. The main reason being liquidity constraints in the stocks that drive the highest returns. Our findings support the efficient market hypothesis, in that we are unable to beat the market with public information.

Following the introduction of the capital assets pricing model in the 1960s, several factors have been introduced to better explain the driving forces behind returns. In 1993, Fama and French found that value stocks tend to outperform growth stocks and that small-cap stocks tend to outperform large-cap stocks. With these findings they created an expansion on the CAPM, known as the Fama-French three-factor model, that includes size and book-to-market risk factors. The Fama-French model has since been expanded to a five-factor model. Meanwhile, hundreds of other firm characteristics have been shown to have some influence on asset pricing. Green et al. (2017) take up John Cochrane's challenge to identify the firm characteristics, from what was referred to as a "zoo of factors" by Feng et al (2017), that provide independent information about U.S. stock returns. They implement 94 characteristics in regression analyses to investigate their ability to explain the variability in returns. These 94 characteristics have since been implemented in other studies, with not only regression as their research methods. In recent times, machine learning has grown in its implementation in both academic research and in the financial industry. Gu et al. (2020) performed a study of the U.S. stock market, using several machine learning algorithms and these 94 firm characteristics. With the objective to provide benchmarks of predictive accuracy of machine learning methods in measuring risk premiums, as well as synthesize the empirical asset pricing literature with the field of machine learning. Since then, Leippold et al. (2021) performed the same study in the Chinese stock market. While they both find improvements in $R^2$ compared to previous literature, the 94 firm characteristics' individual impact differed vastly between the U.S. and the Chinese market. This is interesting because their markets are comprised very differently. The U.S. market is predominantly comprised by institutional investors, while the Chinese market is dominated by retail investors (Leippold et al., 2021). The difference in the importance of the characteristics indicates that not all markets are made equal, but might be products of the psychology of its investors, economy size, degree of regulation, sector compositions and other factors. This raises the question: What matters to the Norwegian investors in the Norwegian market? A market which consists mainly of institutional investors, with retail accounts owning a mere 4.4% of assets (AksjeNorge, 2022), are a relatively small economy compared to the U.S. and China, and have a large industry sector.

In this study we take inspiration from the frameworks of Gu et al. (2020) and Leippold et al. (2021), but employ instead a classification framework rather than regression. We investigate seven different machine learning algorithms, from tree based models to neural networks, to find the best performing one. Furthermore, we analyse the variable importance of 83 firm characteristics in each model to investigate what matters most to market players in the Norwegian market. Finally, we evaluate the portfolios made from the model predictions on the basis of finding a viable investment strategy.

As an investor, you seek to obtain the highest return for the lowest amount of risk. Investors gather information and make predictions to the best of their ability, to maximize their returns. In fact, there is an entire industry built around asset management to gain the highest possible returns. There are hedge funds, mutual funds, 60/40 funds etc, competing against each other for the capital of investors. The investors have the choice of investing in an active portfolio, meaning a portfolio comprising of individually selected stocks, or a passive portfolio, meaning an index portfolio. The reason one would choose active over passive is because of the expectation of higher returns. Meanwhile, paradoxically, the efficient market hypothesis, one of the most fundamental financial theories, suggests that individual investors are unable to outperform the market, because the markets are efficient. There is also research supporting this, Ackermann et al. (1999) found that hedge funds and mutual funds were unable to outperform the market index. Leippold et al. (2021) found that their machine learning portfolios generated returns above that of their index after costs, by a significant margin. Suggesting inefficiencies in the Chinese markets. The research in this thesis is interesting, because by investigating the predictive power of the machine learning models, we fundamentally test the efficient market hypothesis. In that we are trying to beat the market using publicly available information, which according to the hypothesis should already be priced in.

Our findings show that a boosted decision tree model has the highest prediction accuracy of the seven algorithms, with a monthly accuracy of 53.16%. The other models have accuracies between 51.59% and 52.32%. Suggesting that the models find some predictive ability in the variables used. Variable importance varied from each model, but the highest combined importance between all models is characteristics involving risk measures. The second most important group of characteristics is momentum factors, ranking highly in the decision-tree models. The third group of characteristics involve valuation ratios and fundamentals, while the fourth important group involve liquidity factors.

We modelled monthly portfolios, where 6 out of 7 of the long-only portfolios outperformed the index. However, when performing robustness checks for institutional investors, we find that the excess returns disappear and the models significantly underperform. Interestingly, a retail portfolio strategy, consisting of the 75% smallest stocks, showed excess returns. Upon investigating the reason for this market inefficiency, we find, however, that these returns are mainly driven by highly illiquid stocks. Furthermore, the liquidity in these stocks are so low, such that the price equilibrium is so fragile that one would likely move the price sufficiently to significantly reduce returns, whenever entering or exiting positions. We argue that there isn't enough liquidity in these markets to arbitrage the returns away. We call this phenomenon "rainbow-returns", as they are likely only observable and never attainable.

In chapter 2 we give a walkthrough of the relevant literature in explaining returns. We present our research questions in chapter 3. Next, in chapter 4 we detail the methodology used in our research. Furthermore, we explain the data gathering process and treatment of the data in chapter 5, along with a descriptive analysis of the final sample. In chapter 6 we present the results, and do several robustness checks. We conclude the thesis in chapter 7 and give suggestions to further research.

# 2 Theoretical Background

In the following chapter we will describe the theories motivating this thesis. We will first introduce the efficient market hypothesis and it's paradox. Further, we describe the capital asset pricing model and expand further with the Fama-French 5-Factor Model. Lastly, we give an account of some of the latest research in empirical asset pricing.

## 2.1 Efficient Market Hypothesis

The efficient market hypothesis defines an efficient market as a market in which prices always fully reflect all available information Fama (1970). There are several market conditions that hinder efficient adjustment of prices given information, therefore making efficient markets rather unreasonable in practice. Hence, Fama has divided and tested market efficiency in a weak, semi-strong and strong form. The weak form efficiency states that all past available information about a stock is reflected in the price, and therefore deem technical analysis and machine learning unable to outperform the market. The semi-strong form takes this further and implies that prices will reflect all new public information, thereby saying fundamental investment strategies are unable to outperform the market. Lastly, in the strong form prices will fully reflect not only all public available information and all past information, but also all non-public information. In such a market, as all information is accounted for, there is no possibility for outperforming the market.

### 2.1.1 Grossman-Stiglitz Paradox

Creating a machine learning investment strategy would be based on beliefs that there are market inefficiencies that deem all versions of the efficient market hypothesis flawed. One theory that support such beliefs is the Grossman-Stiglitz Paradox, stating that it is impossible for competitive markets to always be in equilibrium. The theory is that if markets always include all information, then nobody would go through with the costly activity of gathering this information, as there is no expectations of returns from doing so. Grossman and Stiglitz therefore propose a model where "prices reflect the information of informed individuals but only partially, so that those who expend resources to obtain information do receive compensation" (Grossman and Stiglitz, 1980). If we assume that the model is true, we open the possibility to achieve abnormal returns. Through machine learning, the mispricing from uninformed investors could be exploited to achieve returns.

## 2.2 Capital Asset Pricing Model

The Capital asset pricing model (CAPM) was developed in the early 1960s by William Sharpe (1964), Jack Treynor (1962), John Lintner (1965) and Jon Mossin (1966) (Perold, 2004). The model was developed to create a framework for answering how the risk of an investment should affect its expected return. The equation for calculating the expected return given its systematic risk is:

$$ER_i = R_f + \beta_i(ER_m - R_f)$$

However, this only shows the expected return of the stock given the benchmark used to calculate beta. To consider the difference between the realized return and the expected return given the model, we need to include the alpha Jensen (1968). The equation will then be:

$$R_{it} - R_{ft} = \alpha_i + \beta_i(R_{mt} - R_{ft}) + \epsilon_{it}$$

Where $R_{it}$ is the realized return of the investment i at time t and $R_{ft}$ is the risk-free rate at the time t. $\alpha_i$ represents the abnormal return for the investment i. $\beta_i$ represents the beta of the investment and measures the systematic risk relative to the market index at time t, $R_{mt}$. $\epsilon_{it}$ is the error term for the firm at time t.

## 2.3 Fama-French 5-Factor Model

In 1992 Fama and French concluded that size and book-to-market equity proxy for common risk factors in stock returns (Fama and French, 1993). With this knowledge they created the Fama-French three-factor model that expands on CAPM by adding the size risk and value risk factors to the model. This expansion to the CAPM means that the model seeks to explain stock returns through two additional factors known as: small minus big

(SMB), which considers the outperformance of small-cap stocks relative to large-cap stocks, and high minus low (HML), which considers the outperformance of stocks with a higher book-to-market value relative to those with a low book-to-market value. Mathematical representation of the Fama French three-factor model (Fama and French, 2015):

$$R_{it} - R_{ft} = \alpha + \beta_i(R_{mt} - R_{ft}) + s_i SMB_t + h_i HML_t + \epsilon_{it}$$

After the introduction of the three-factor model researchers were eager to include other factors to the model. One of those researchers were Mark Carhart. He found that funds with higher return last year would have higher-than-average expected return for next year, but not in the years after. In 1997 he proposed a momentum factor to be added to the three-factor model from Fama and French. The introduction improved the model's ability to explain stock returns as the factor were distinct from the Fama-French factors. Mathematical representation of the Carhart 4-factor model (Carhart, 1997):

$$R_{it} - R_{ft} = \alpha + \beta(R_{mt} - R_{ft}) + s_i SMT_t + h_i HMLt + m_i MOM_t + \epsilon_{it}$$

Where $MOM$ takes into account the one-year momentum in stocks.

Fama and French adapted their model to include five factors in 2014 (Fama and French, 2015). The new factors introduced were robust minus weak (RMW), which considers the outperformance of companies with a more robust operating profitability relative to those with weaker, and conservative minus aggressive (CMA), which considers the outperformance of companies with lower investments relative to those with higher. Mathematical representation of the Fama French 5-factor model plus momentum (Fama and French, 2015):

$$R_{it} - R_{ft} = \alpha + \beta_i(R_{mt} - R_{ft}) + s_i SMB_t + h_i HML_t + r_i RMW_t + c_i CMA_t + m_i MOM_t + \epsilon_{it}$$

With the introduction of these factors an investing strategy revolving around them have also risen to prominence. This is what is known as factor investing, which also has a slightly altered version known as smart beta investing. The strategies revolve around choosing companies based on quantifiable factors, like those explained above, and receive the associated risk premium.

## 2.4 Machine Learning

Further research on factors that drive returns have been done using machine learning algorithms. Gu et al. (2020) uses the list of characteristics from Green et al. (2017) and go through an extensive list of machine learning methods, with the purpose of measuring equity risk premiums in the US market. Through their research they demonstrate the possibility for gains to investors by the implementation of these machine learning models. Through the extensive list of firm characteristics and macroeconomic predictors, used to train the algorithms, they also propose what factors are important in the US market. Their findings therefore add to that of the Fama-French five-factor model, helping us better understand what drives returns in the US market.

Inspired by their results, similar research has been undertaken on the Chinese stock market by Leippold et al. (2021). Interestingly, the factors that are important in predicting the market vary greatly from those in the US market. The most critical factors in the US market revolves around price trends and momentum, while these factors only play a minor role in the Chinese market. Even though the characteristics between the markets vary, they also found that machine learning methods could be successfully applied to the Chinese market.

# 3 Research Questions

Limited research has been done on empirical asset pricing through machine learning on the Norwegian stock market. This kind of research has mainly been done on U.S. and Chinese markets previously. This thesis aims to contribute to existing literature on the field and provide additional benchmarks in the Norwegian markets, upon which improvements can be made. Based on this, this thesis explores the following research question:

> *RQ: Using machine learning algorithms, to what degree can we predict next month's return on the Oslo Stock Exchange, and which algorithm performs best?*

The disparity in variable importance between the U.S. (Gu et al., 2020) and Chinese markets (Leippold et al., 2021), raises the question of what is important in the Norwegian markets. We, therefore, investigate the research question:

> *RQ: Which variables are the most important when predicting Oslo Stock Exchange?*

The variable importance of the models can also give investors a hint towards what information is the most valuable to gather and use in their analysis when making their own predictions on the Oslo Stock Exchange.

Finally, we investigate the returns investors can achieve with portfolio modelling through machine learning, and if these are significantly larger than the market portfolio.

> *RQ: Is it possible to obtain an alpha through machine learning portfolio modelling?*

The three research questions presented are interesting, because, fundamentally, they test the efficient market hypothesis. As all new information, according to the theory, is instantly priced into the market, machine learning modelling, as a method, should not yield results. On the other hand, if there is alpha to obtain, investors can enjoy the benefits of machine learning modelling.

# 4 Methodology

In this chapter we will explain what machine learning (ML) is and how we have used it as a method in our study. We will first give a general overview of machine learning, then we will elaborate deeper on each ML-method that was used in this study. Lastly, we will show how the performance of the ML-models are measured and evaluated.

## 4.1 Machine Learning

Machine learning is a process of building statistical models with the purpose of predicting an output variable, from a set of input variables. The input variables are typically denoted using the variable symbol $X$, with a subscript to distinguish them. The input variables can go by different names, such as predictors, independent variables, features or sometimes just variables. The output variable is often called the response or dependent variable and is typically denoted using the symbol $Y$ (James et al., 2013).

In classical econometrics, a linear model explaining the relationship between $Y$ and $X$, in generalized form, is written:

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \epsilon,$$

where $\beta_i$ is a coefficient explaining the relationship between $Y$ and $X_i$, and $\epsilon$ is an error term, which is assumed to be normally distributed with a mean of 0, and uncorrelated with the independent variables. This is called a regression problem if the response,$Y$, is a continuous variable. However, if $Y$ is binary, then we are more interested in what the probability of $Y$ equals to 1 is, given our independent variables, $X_i$. This is called a classification problem, and can be expressed like:

$$Logit(P(Y = 1|X_1, \ldots, X_p)) = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \epsilon$$

In classical econometrics we are often interested in statistical and causal inference, while in machine learning the interest is more directed towards accurate predictions (James et al., 2013). A machine learning model can be written in the general form like:

$$Y = f(X_1, X_2, \ldots, X_p) + \epsilon,$$

where $f$ is an unknown function of the independent variables, and e is an error term with the same assumptions as in the linear model. $f$ is the function to be estimated to be able to predict Y. In machine learning, we can say the different algorithms represent $f$.

### 4.1.1 Supervised and Unsupervised Learning

In the machine learning process, we distinguish between two sub-categories of machine learning called supervised and unsupervised learning. The distinction between the two is simple. In the learning process, supervised learning models are given data that is clearly labelled and told what the correct outcomes for the data are, hence "supervised". While in unsupervised learning, the model is only given unlabelled data with no outcome. The model analyzes and finds pattern without the need for human assistance. The goal in unsupervised learning is often different from that in supervised learning, with other problems than regression or classification, for example clustering. Where the model's objective is to sort the data into categories, or "clusters".

The goal in unsupervised learning is often different from that in supervised learning. Unsupervised learning causes other problems than regression or classification, for example clustering, where the model's objective is to sort the data into categories, or "clusters".

### 4.1.2 Classification vs Regression

In the supervised learning-category of machine learning, we can further distinguish between two kinds of models. Regression and classification models. As mentioned previously, a regression model is used when the objective is to predict a numeric, continuous variable. Like sales given advertisement spending, or house prices given its location and number of bedrooms. While a classification model is used when the objective is to determine whether an outcome belongs in a certain class, a binary classification model tries to predict the data into two distinct classes. For example, 0 or 1, male or female, positive or negative etc.

Previous studies have predicted stock returns using machine learning regression models (Gu et al. (2020), Leippold et al. (2021)). We mainly base our study on the works of Gu et al. (2020), but with a few twists of our own. The difficulty of predicting stock returns as a continuous variable is high, as all random, daily noise can affect stock prices. This noise is hard to measure and include in a regression model. In classical econometrics, even finding a variable that has significance for a stock price, with no backdoor, is notoriously difficult. Gu et al. (2020) report $R^2$ of around 1% for all models, meaning they can explain around 1% of all variation in a given stock's price movement. Instead of predicting a stock's price or exact return like Gu et al. (2020) and Leippold et al. (2021), we attempt to predict a stock price's direction. In this study we are predicting if a stock's return next month will be positive or negative, i.e., a classification problem.

## 4.2 Models used

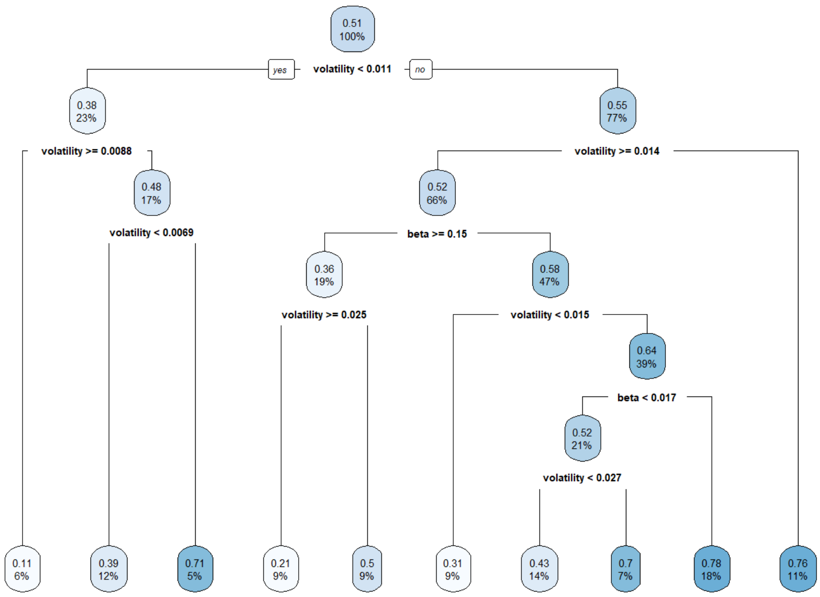
There are many kinds of machine learning algorithms, and some algorithms can even handle both regression and classification tasks. However, as not all algorithms handles both types, the goal of one's study will have implications of which algorithms are available. We will in the following section describe the machine learning algorithms used in this study.

### 4.2.1 Decision Trees

Decision trees are a supervised learning method, used for classification and regression problems. To predict the value of a variable, the model makes simple decision rules created from the data. To do so, it creates a decision tree where each node represents a decision with different outcomes. The outcomes from each node will either branch out to a new node or stop at a leaf node, which is what the final output is known as. Since we are using a classification model, the leaf node will either end in a 0 or a 1, in our example showing whether we will have positive returns or not.

Figure 4.1 offers a visualization of a decision tree using our sample data. In order to keep the visualization simple, the decision tree is created by using only the two characteristics, beta and volatility. In the figure we can see that, depending on the value of both volatility and beta, the decision tree results in ten output nodes. The output shows the average value of our classifier, that is positive return, and percentage of the data that is categorized by that output node. The higher the value in the output node, the higher amount of stocks with positive return are represented by those characteristics.

**Figure 4.1:** Simple decision tree example
Figure 4.1 shows a visualization of a decision tree using our sample data. For simplicity the decision tree is created by using only the characteristics beta and volatility.




### 4.2.2 Boosted Decision Trees and Extreme Gradient Boosting

With gradient boosting we take our decision tree model and chain multiple versions of the decision trees together. When doing so, it will increase accuracy by using a loss function. This loss function is high when the outcome and the prediction disagree and 0 if they are in perfect agreement. Every tree will then boost the attribute that lead to misclassification, therefore reducing the loss compared to the previous tree. We then end up with multiple trees, that build on top of each other and improve on the previous tree. Extreme gradient boosting (XGBoost) is an extension to gradient boosted trees, improving both calculation speed and performance. Here the second partial derivatives of the loss function are used, which improves information about the direction of gradients and the minimizing of the loss function. XGBoost has also added more advanced regularization, two common types are lasso and ridge, that helps with reducing the potential of overfitting. This is added through multiplying the sum of features weights with the regularization term in the loss function. XGBoost also takes advantage of parallelization during the construction of each tree, by training the branches in the tree separately.

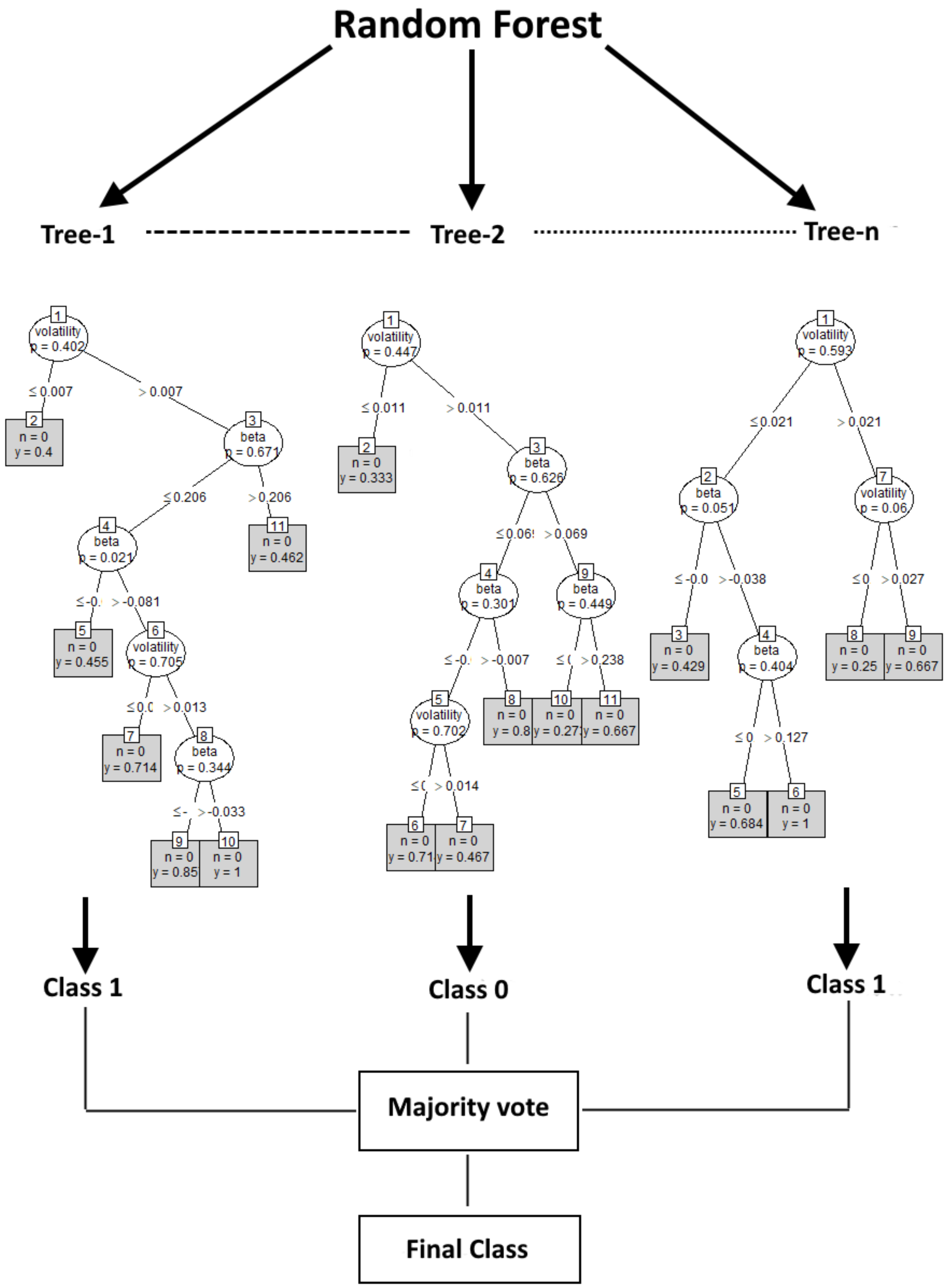
### 4.2.3 Random Forest

Similarly to XGBoost, random forest is also a supervised machine learning algorithm that builds on decision trees. Random forest consists of many decision trees which operates in an ensemble, therefore creating what can be seen as a forest. It does so while also using feature randomness, by randomly selecting inputs or combinations of inputs at each node to grow each tree (Breiman, 2001). This is done to create a low correlation between the trees and therefore creating better results from the group of trees, similar to what is done in investments when combining assets with lower correlation, like stocks and bonds. Depending on the task at hand, random forest will use either, for regression, an average of all the individual decision trees or, for classification, what is called a majority vote, where each tree casts a unit vote for the most popular class (Breiman, 2001).

Figure 4.2 illustrates a simple random forest model using the characteristics volatility and beta from our sample data. We have an $n$ number of trees that all end up with a classification result, which in our case is either 1 or 0. The results are then combined into one by taking the majority vote. Through the majority voting we get the final classification for the combination of trees.

**Figure 4.2:** Random Forest illustration

Figure 4.2 illustrates a simple random forest model using the characteristics volatility and beta from our sample data. We have an $n$ number of trees that all end up with a classification result, which in our case is either 1 or 0. The results are then combined into one by taking the majority vote. Through the majority voting we get the final classification for the combination of trees.



### 4.2.4 Support Vector Machine

A support vector machine (SVM) is a supervised machine learning algorithm mostly used for classification problems. To do so, the model finds what is known as an "optimal hyperplane" to the classification problem. The hyperplane is a decision boundary that separates the two groups depending on their attributes. For a dataset with several input features, like ours, the dimension of the hyperplane will be one less than that of its ambient space. Meaning that a 3-dimensional dataset would have 2-dimensional hyperplanes. For training the model finds the maximum marginal hyperplane, which is done through maximizing the distance between the closest pairs of datapoints for the classes (Evgeniou and Pontil, 1999).

#### 4.2.4.1 The Kernel Trick

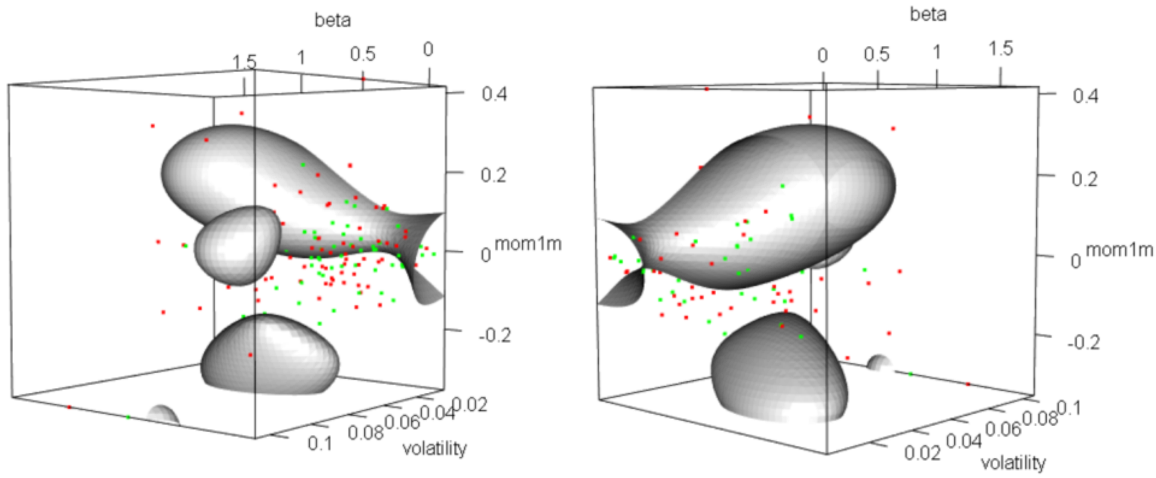
To get the data into higher dimensions, SVM implements a method often referred to as "The Kernel Trick". When using the kernel trick, the model also relies on "soft marginal classifiers", which can be seen as errors on the training data (data points not correctly separated by the hyperplane). The degree of tolerance, often represented by C, tells the model the grade of misclassification that is tolerated when creating the hyperplane, and is therefore an important hyper-parameter. Using a larger C will allow less misclassification, as it will penalize the misclassifications more. The advantage of using the kernel trick is that the method enables us to construct algorithms in dot product spaces (Hofmann et al., 2008). This is because data is represented through a similarity measure between the original data observations instead of transforming them. If we assume a dataset in which $x, z \epsilon X$ and a map $\phi :$ then the Kernel function can be defined as (Hofmann et al., 2008):

$$k(x, z) = \langle \phi(x), \phi(y) \rangle$$

Here $\phi$ maps the data into a dot product space, which is a function that corresponds to the lower dimension vector of the data.

The figure shows an illustration of SVM-hyperplanes, created using a sample of 150 datapoints from our dataset. To make it interpretable we have reduced the number of input features to only include beta, 1-month momentum and volatility. The green datapoints in the 3D model represents the value of input features which have yielded positive return, while the red represents the value of input features which have yielded negative return. The complexity of the data means that the model cannot differentiate the groups with the use of only one hyperplane. The hyperplanes of our data are illustrated by the grey "figures" in the 3D plot, where the hyperplanes will differentiate the datapoints into either positive return, which can be interpreted as "inside" the hyperplanes, and non-positive return, which is the datapoints outside the hyperplanes.

**Figure 4.3:** Hyperplane illustration
Figure 4.3 shows an illustration of SVM-hyperplanes, created using a sample of 150 datapoints of beta, 1-month momentum and volatility from our dataset. The green datapoints in the 3D model represents the value of input features which have yielded positive return, while the red represents the value of input features which have yielded negative return. The complexity of the data means that the model cannot differentiate the groups with the use of only one hyperplane. The hyperplanes of our data are illustrated by the grey "figures" in the 3D plot, where the hyperplanes will differentiate the datapoints into either positive return, which can be interpreted as "inside" the hyperplanes, and non-positive return, which is the datapoints outside the hyperplanes.




### 4.2.5 Logistic Regression

Logistic regression can be seen as an extension of linear regression and is used to model the probabilities of different outcomes for classification problems. In this paper the task will be to predict whether a positive return (1) is achieved for the next period or not (0). Instead of using a straight line to solve the problem, like linear regression does, logistic regression uses a logistic function to limit the output between 1 and 0. The logistic function, also called sigmoid function, according to Monroe (2017):

$$\sigma(z) = \frac{1}{(1 + e^{-z})}$$

Here $\sigma(z)$ will return a probability between 0 and 1 for any value of $z$.

Logistic regression uses maximum likelihood estimation to determine the values for the parameters in the model. It does so by checking what parameters will lead to the highest probability of observing the data. The maximum likelihood estimation is then the value of the parameters that maximize the likelihood function.

### 4.2.6 Neural Networks

Neural networks are a machine learning method inspired by how neurons work in the human brain. The network is built using artificial neurons, known as nodes, that are put into node layers (Schmidhuber (2015)). These layers include an input layer, one or several hidden layers, and an output layer that are connected through the nodes. Each node has an associated weight and threshold that decides, depending on the value relative to the threshold, whether the data will be sent through to the next layer in the network or not. To achieve threshold values that maximize the chance of correctly identifying what data to send to the next layer, the model needs training data to learn and improve from. When training, the mean squared error cost function is used, which measures the average squared difference between the actual observation in the data and the predicted values.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2$$

Where $N$ is the average of all results, $Y$ is the true observation and $\hat{Y}$ is the predicted value. Through the cost function, the model adjusts its weights and seek to reach the local minimum. To effectively find the local minimum gradient descent is used with each repetition of training are bringing it closer.

## 4.3 Performance Measures

In the training phase of a machine learning modelling process, we must first tell the model how to evaluate its own learning. The model needs to know when it is making progress and when it's not. The model is given a performance metric, that it either tries to maximize or minimize. When the model later makes predictions, we also use these metrics to determine how solid these predictions are. Choosing the correct metrics to evaluate a model is important, not only for objectively showing a model's performance, but also so that the algorithms learns, and predicts, in the intended way, in order to maximize results. There are various ways to evaluate the performance of a model, and their uses are determined by what kind of model was implemented and the objective of the study. Since this study implements classification models, we will focus on performance measures for these.

### 4.3.1 Confusion Matrix

One of the most elementary and intuitive visualizations of a model's performance is the confusion matrix. Figure 4.4 shows a matrix with 4 categories. These are true positive (TP), false positive (FP), false negative (FN) and true negative (TN). We will use our own study as an example. Our model attempts to predict whether a stock's next month return will be positive or negative. If the model predicts a stock's next month return to be positive, and the actual value is positive, then that prediction will be considered a true positive. If the model predicts positive returns, but the actual returns are negative, the prediction is classified as a false positive. Inversely for true negative and false negative. That means the matrix shows how many predictions were correct and how many were incorrect. Obviously, we want to minimize the false positives and false negatives to maximize the model's correct predictions. We won't use confusion matrices as a performance measure in this study, but the concept of true and false positives is important for the measure we do use.

**Figure 4.4:** Confusion Matrix

| | | **Predicted** | |
|---|---|---|---|
| | | **Positive** | **Negative** |
| **Actual** | **Positive** | True Positive | False Negative |
| | **Negative** | False Positive | True Negative |

### 4.3.2 Area Under Curve

A metric and visual employed in this study is the area-under-curve (AUC), visualized on "Receiver operating characteristic curve", also known as ROC-curve. This metric shows how greater the model's predictions are compared to a random guess. There are two other metrics that are needed to display this.

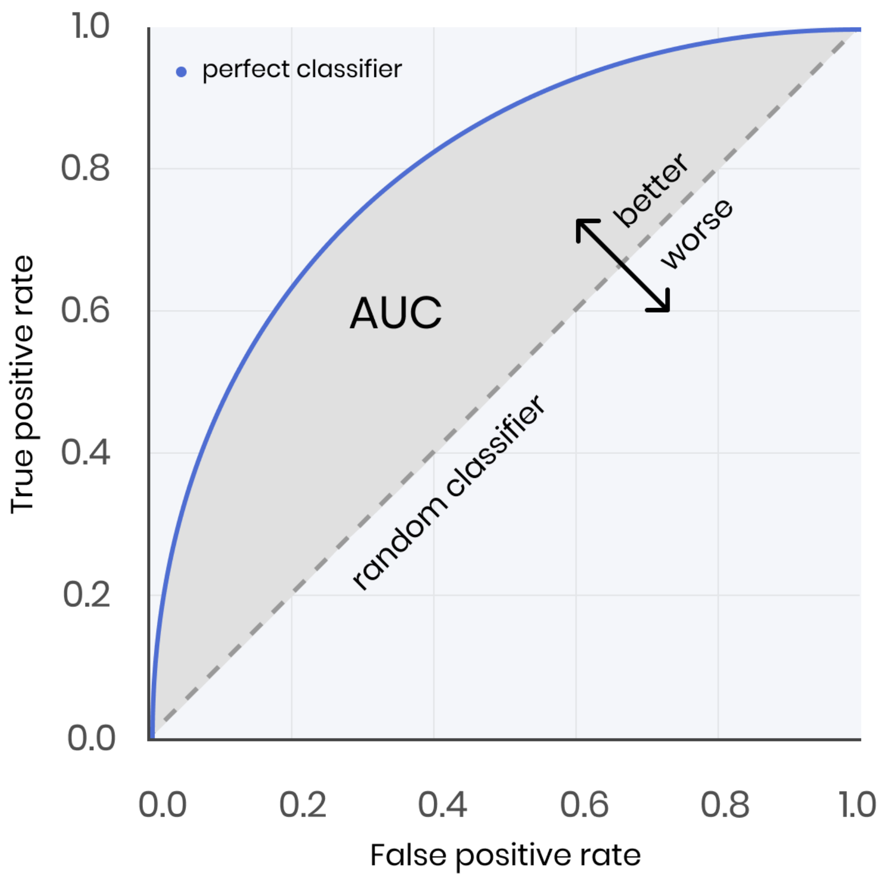
- True positive rate (TPR), also known as "recall", is the ratio of which the model predicts a positive outcome when the actual outcome is positive (James et al., 2013):

$$TPR = \frac{TP}{TP + FN}$$

- False positive rate (FPR) is the ratio of which the model predicts a positive outcome when the actual outcome is negative (James et al., 2013):

$$FPR = \frac{FP}{TN + FP}$$

The ROC-curve is made by plotting the true positive rate against the false positive rate, with a predetermined discrimination threshold. The threshold represents the TPR and FPR for random guessing between two outcomes, which is 0.5.




The AUC indicates how much better the predictions are than a random guess. The further away the curve is to the discrimination threshold, the better the model performs.
To make it easier to compare the models' AUC, we can simply compare the area under the curve's numeric value.

### 4.3.3 Accuracy

Accuracy is another method to evaluate a model's performance and compare models' predictive abilities to each other. Accuracy is a numeric measure of how many predictions were correct, out of the total number of predictions the model made (James et al., 2013).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

An accuracy of 100% means the model predicts correct 100% of the time, while 0% means every prediction was wrong. Depending on the difficulty of the prediction task a model has, achieving an accuracy of 100% is realistically not possible. The task of predicting the stock market is extraordinary difficult, as the random daily variance that happens globally can influence stocks and market sentiment. However, Gu et al. (2020) showed that a small increase in predictive ability can have large gains in portfolio performances. Indicating that the challenge might be worth undertaking.

## 4.4 Variable Importance

Variable importance, also known as feature importance, shows how much the model relies on the different variables to make its prediction. The more the model relies on a variable, the higher its importance score will be.

To check the importance of each variable in neural networks, permutation variable importance is used. The idea behind this method is simple. If the variable is important to the model, then shuffling this variable, while keeping all other constant, should increase the model's error. The higher the increase in model error, the higher the importance of that variable is. If it doesn't increase the error, then the model didn't rely on that variable for its prediction. This method is repeated for every variable in the model, so that the end-results are comparable importance scores for each variable. For XGBoost and Random Forest, what is known as "gain" is used. Here, the variable importance is calculated by taking each variables contribution to each tree in the model. Having a higher contribution on more trees will therefore give a variable a higher influence on the model's predictions. For logistic regression, the absolute value of the standardized coefficients is used. The larger the coefficients, the more importance that variable has on the model's prediction. For SVM, a 1-dimensional sensitivity analysis is used. The sensitivity analysis looks at how a change in our variable, while keeping all other variables constant, affects our target variable. The higher the effect on our target variable, the more important the variable is in making our predictions.

# 5 Data

In this chapter we will share the data sources and how the data was cleaned and wrangled. Furthermore, we will present the variables constructed and used in the models, and lastly, we will describe firm characteristics in the final dataset.

## 5.1 Data Sources

Though we are studying the Norwegian stock market, we have chosen to limit the study to stocks on the Oslo stock exchange (OSE). Meaning we are excluding stocks listed on the Euronext Growth Oslo and Euronext Expand Oslo indices. The data we collected are of two types and from two different sources. We gathered daily stock data and yearly accounting data. The stock data were gathered from NHH's own database, Børsprosjektet, which they in turn received directly from the exchange up until Euronext's takeover. The stock data span from January, 2000 up until the takeover on November 27th, 2020, and contain columns such as closing price, share turnover, turnover in Norwegian kroner, value-weighted average price and shares outstanding, of all the publicly listed firms on the exchange. The accounting data were gathered from SNF, a research company within the NHH-community. This data contain the yearly income statement and the balance sheet, and various variables generated by SNF, of all companies in Norway, both listed and unlisted.

## 5.2 Data Wrangling

The data, in their raw forms, contain all the firms on the Oslo stock exchange and all the concerns in Norway. However, as not all the concerns are listed on the exchange, an index was made of all the firms in the stock data. This index was used to only keep the firms that were present in both the stock data and accounting. In addition, a criterion was set that the firms had to have at least 4 years worth of observations. There were two reasons for setting a criterion. The first reason is because several of the variables were constructed with rolling windows of up to 4 years, secondly, to save both computing power and time when constructing the variables later.

We split the stock data by date into two datasets. One dataset for training and one for testing. After this split, the stock data and accounting data were merged according to each firm's organization number and year. The training dataset contained data from year 2000 to 2015, and the testing dataset from the year 2012 to 2020. The overlap in the datasets from 2012 to 2015 is to give the testing dataset the 4-year rolling window for variable construction. After the variables were created, missing values were removed, such that the rolling window, or calculation period, of 4 years was removed. Meaning that the final training data were from year 2004 to 2015, and the testing data were from year 2016 to 2020.

Before the data were used in the models, all variables with exception of factors, logged variables and variables with a point score, were winsorized at the 2% level. In the neural network models, all the variables, expect factor variables, were normalized in addition to being winsorized.

### 5.2.1 Missing data in accounting data

While reviewing the yearly accounting data, we found that some firms had years of missing data. This posed an issue, as several of the variables we intended to construct show period-to-period change. To reduce the amount of data that had to be removed, we calculated the length of the periods on both sides of the "gap" and kept the data with the largest length. If the lengths were equal, the most recent length would be kept. This was applied to both the training and testing set. In addition, the length of the period that was kept had to be of 4 years or longer, in order to meet the criteria mentioned above.

The firms that have foreign headquarters do not have complete accounts in our data. The data we had for those firms only included business done in Norway. This is likely because only the subsidiaries that do business in Norway has reporting duty, and not the concern as a whole. The firms were excluded, as the full accounts are not available.

## 5.3 Variables

This study implements 83 independent variables and 1 dependent variable. The predictors one chooses when attempting to undertake the tall task of predicting the stock market, are essential to the quality of the outcome. Good data and relevant variables are needed. All the independent variables we have included in our study are previously studied, and have shown significance in explaining stock returns. Some are more often used than others, such as beta and momentum. Their frequency of change is either monthly or yearly. This is because it is either based on the stock data, accounting data or both. Implementation of quarterly accounting data would have been preferable, to include more variance in the variables containing an accounting element. They are a good sample of what we can call technical and fundamental indicators. An overview of the variables, with a brief description, is given in table 3.1 below. Further explanations of how the variables are calculated are given in table A1 in the appendix. The variables' authors and corresponding paper, can be found in table A1.1 in the appendix.

**Table 5.1:** Variables
Table containing the predictors and the outcome, with description and frequency. "Variable name" shows the name of the variable in the dataset. "Description" describes the variable. "Frequency" is an indicator for how often the variable changes.

| Variable name | Description | Frequency |
|---|---|---|
| positive_return | Indicator for positive or negative return in the month | Monthly |
| absacc | Absolute accruals | Yearly |
| acc | Working capital accruals | Yearly |
| agr | Asset growth | Yearly |
| beta | Beta | Monthly |
| betasq | Beta squared | Monthly |
| bm | Book-to-market | Monthly |
| bm_ia | Industry-adjusted book to market | Monthly |
| cash | Cash holdings | Yearly |
| cashdebt | Cash flow to debt | Yearly |
| cashspr | Cash productivity | Monthly |
| cfp | Cash flow to price ratio | Monthly |
| cfp_ia | Industry-adjusted cash flow to price ratio | Monthly |
| chato | Change in asset turnover | Yearly |
| chato_ia | Industry-adjusted change in asset turnover | Yearly |
| chinv | Change in inventory | Yearly |
| chmom | Change in 6-month momentum | Monthly |
| chpm | Change in profit margin | Yearly |
| chpm_ia | Industry-adjusted change in profit margin | Yearly |

| | | |
|---|---|---|
| chsho | Change in shares outstanding | Monthly |
| chtx | Change in tax expense | Yearly |
| cinvest | Corporate investment | Yearly |
| currat | Current ratio | Yearly |
| depr | Depreciation | Yearly |
| divi | Dividend initiation | Yearly |
| divo | Dividend omission | Yearly |
| dolvol | NOK trading volume | Monthly |
| dy | Dividend to price | Monthly |
| egr | Growth in common shareholder equity | Yearly |
| gma | Gross profitability | Yearly |
| grCAPX | Growth in capital expenditures | Yearly |
| herf | Industry sales concentration | Yearly |
| idiovol | Idiosyncratic return volatility | Monthly |
| ill | Illiquidity | Monthly |
| invest | Capital expenditures and inventory | Yearly |
| lev | Leverage | Yearly |
| lgr | Growth in long-term debt | Yearly |
| ma_price_short | Moving average of 10 days | Monthly |
| ma_price_int | Moving average of 50 days | Monthly |
| ma_price_long | Moving average of 200 days | Monthly |
| maxret | Maximum daily return in the given month | Monthly |
| mom1m | 1-month momentum | Monthly |
| mom6m | 6-month momentum | Monthly |
| mom12m | 12-month momentum | Monthly |
| mom36m | 36-month momentum | Monthly |
| ms | Financial statement score | Yearly |
| mve | Market capitalization | Monthly |
| mve_ia | Industry-adjusted market capitalization | Monthly |
| nincr | Number of earnings increases | Yearly |
| operprof | Operating profitability | Yearly |
| pchcapx_ia | % Industry adjusted % change in capital expenditures | Yearly |

| | | |
|---|---|---|
| pchcurrat | % change in current ratio | Yearly |
| pchdepr | % change in depreciation | Yearly |
| pchgm_pchsale | % change in gross margin - % change in sales | Yearly |
| pchquick | % change in quick ratio | Yearly |
| pchsale_pchinvt | % change in sales - % change in inventory | Yearly |
| pchsale_pchrect | % change in sales - % change in A/R | Yearly |
| pchsaleinv | % change sales-to-inventory | Yearly |
| pctacc | Percent accruals | Yearly |
| ps | Financial statements score | Yearly |
| quick | Quick ratio | Yearly |
| rd | R&D increase | Yearly |
| rd_mve | R&D to market capitalization | Monthly |
| rd_sale | R&D to sales | Yearly |
| realestate | Real estate holdings | Yearly |
| roaq | Return on assets | Yearly |
| roavol | Earnings volatility | Yearly |
| roeq | Return on equity | Yearly |
| roic | Return on invested capital | Monthly |
| rsup | Revenue surprise | Monthly |
| salecash | Sales to cash | Yearly |
| saleinv | Sales to inventory | Yearly |
| salerev | Sales to receivables | Yearly |
| sgr | Sales growth | Yearly |
| sp | Sales to price | Yearly |
| std_nokvol | Volatility of liquidity (NOK turnover) | Monthly |
| std_turn | Volatility of liquidity (share turnover) | Monthly |
| stdacc | Accrual volatility | Yearly |
| stdcf | Cash flow volatility | Yearly |
| tang | Debt capacity/firm tangibility | Yearly |
| tb | Tax income to book income | Yearly |
| turn | Share turnover | Monthly |
| volatility | Return volatility | Monthly |

| | | |
|---|---|---|
| zerotrade | Number of zero-trading-days | Monthly |

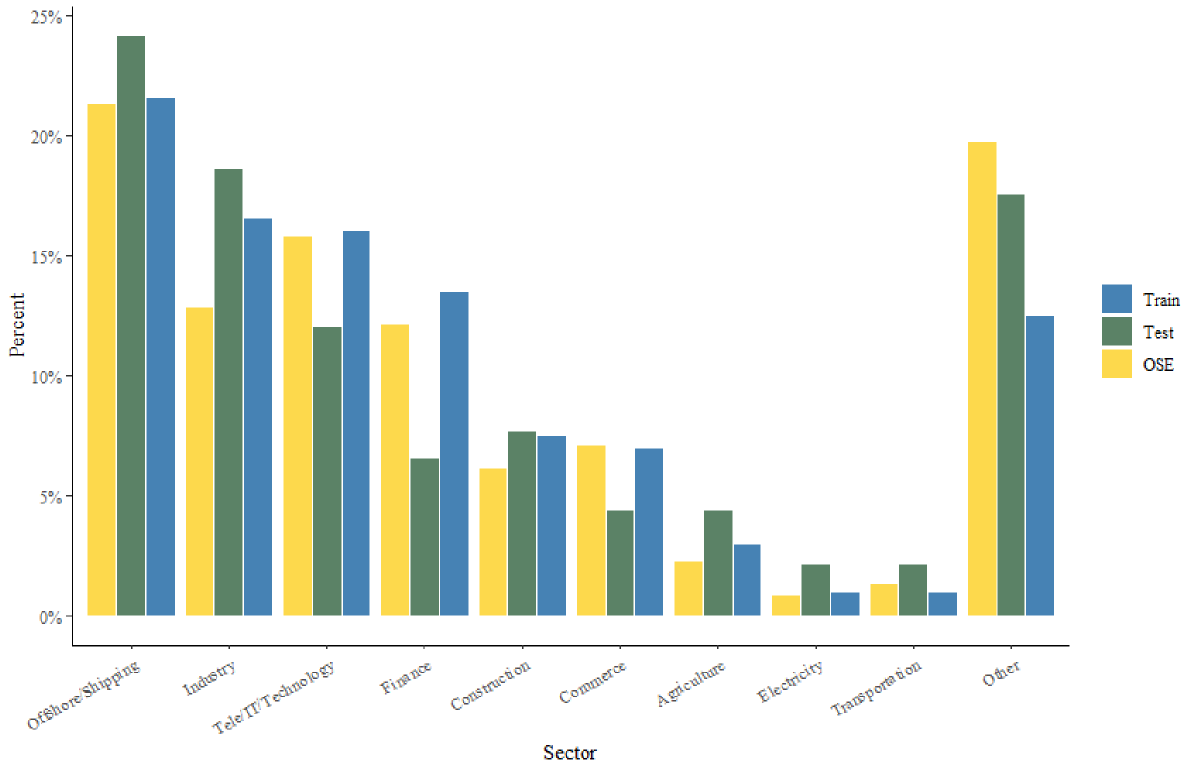
## 5.4 Descriptive Analysis of sample

In this section we will give a descriptive analysis of the training and testing samples' sector distribution, the number of firms, their sizes and returns. Also, steps taken for handling unclear sectors.

### 5.4.1 Sector Distribution

Figure 5.1 shows the distribution of the different sectors each firm is categorized in. The categorization comes from the accounting data, where the firm itself has self-categorized to a certain sector.

**Figure 5.1:** Distribution of sectors
In figure 3.1 we show the distribution of firms in each sector within our samples, compared to the entire Oslo Stock Exchange. Note that the distribution is the number of firms within each sector, not the distribution of market value.



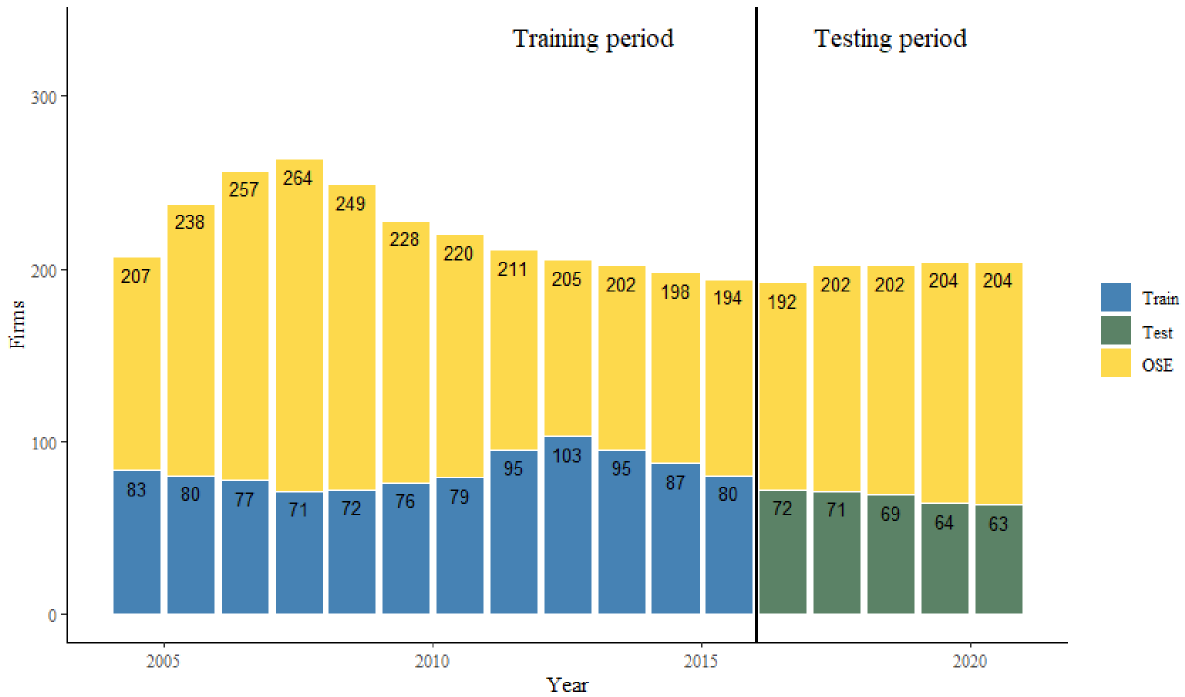

We compare the distributions of the training and test samples to each other, additionally to the Oslo stock exchange (OSE). The objective is that samples are as similar to each other, and the real population, as possible. From the figure we can see that both the training set and testing set are deviating somewhat from the true population. Offshore/shipping is, to no surprise, the sector with the most firms on the Oslo stock exchange, followed by Telecom/IT/Technology and Industry. The biggest discrepancy is between the training and test sample in the finance sector, where the occurrences of finance firms in the test sample are about half that of the training sample. In addition, both samples are somewhat over-represented in the industry sector. However, the samples follow the true distribution closely for the most, so we argue that the samples lose little to no ability to be representative.

The other category is large and there are big variations in the type of firms within this sector. The category contains hotel chains, geo-service providers, biotech companies, manufacturers within the renewable energy space and other sectors. This is not optimal, as all firms in each sector will be industry-adjusted for that the sector other. Meaning that firms in the other category will be industry-adjusted for many other sectors as well. This skews the variables and render them incorrect. Because of this, the category is discarded and only the remaining categories will be used in the models.

### 5.4.2 Number of firms

In the final dataset there are total 177 unique firms. In the training period there are a total of 174 unique firms, while in the testing period there are 75. In figure 5.2 we show how many individual stocks are in our samples, compared to the total number of stocks on the OSE. We split the samples by the training period and testing period, which are designated by the colors blue and green, respectively. OSE is indicated by yellow.

**Figure 5.2:** Number of firms by year
Figure 3.2 shows how many firms are in the dataset each year compared to how many firms are listed on OSE, after removing firms in the sector category "Other".



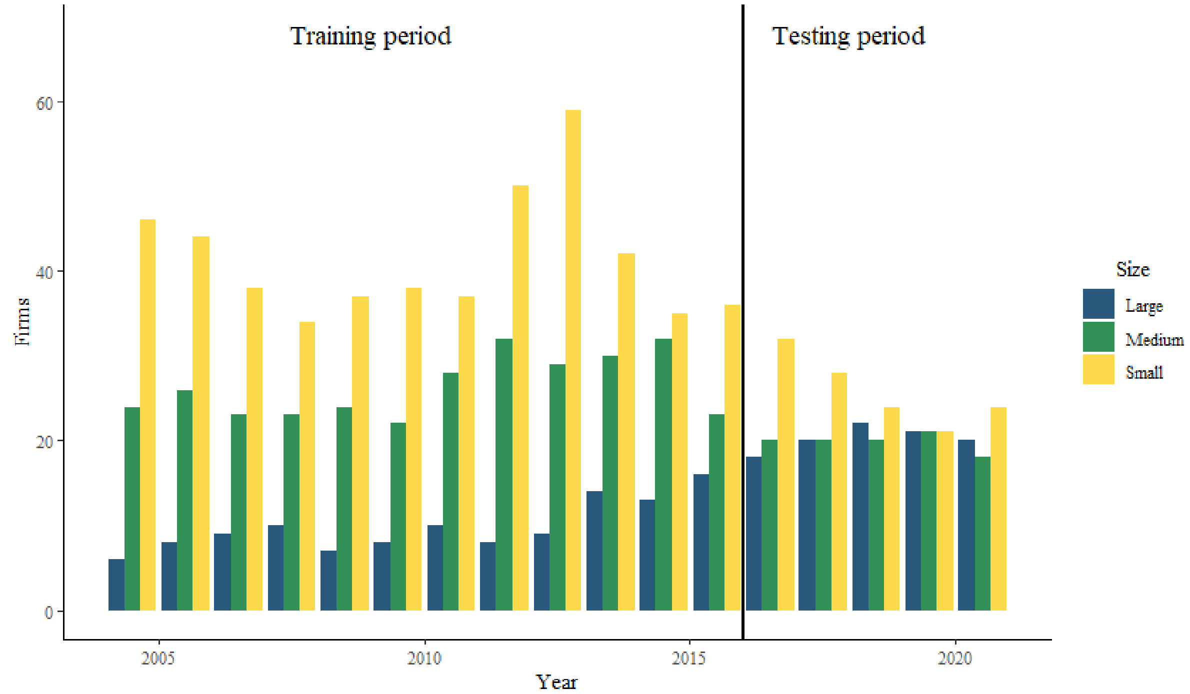

As mentioned, we have excluded firms that have headquarters in a foreign country because of the missing accounting data. The reasons for the yearly changes in the number of firms are multiple. Firms can go off, or be taken off, the exchange as a result of a buyout or bankruptcy. Firms can go on the exchange through an IPO. The accounting data tends to be more complete in later years, rather than the early. This means the "gaps" mentioned previously tends to be early in the data. Consequently, the "side of the gap" that is longer is often after the gap. This can be why there is a decline in 2007 and the years before. Furthermore, firms in our dataset must meet the 4 years of observations-criteria. Meaning that even though a firm has been on the exchange since 2005 or 2018, it will not be included in our dataset until 2009 or, in the latter case, not at all.

### 5.4.3 Firm sizes

Figure 5.3 shows the distribution of the firms' market value in the sample. In finance, it is common to divide sizes into three categories. Small, medium and large. The criteria for what makes a firm size small or large are not set and can vary between markets and case use. We have made the following definitions of the different sizes:

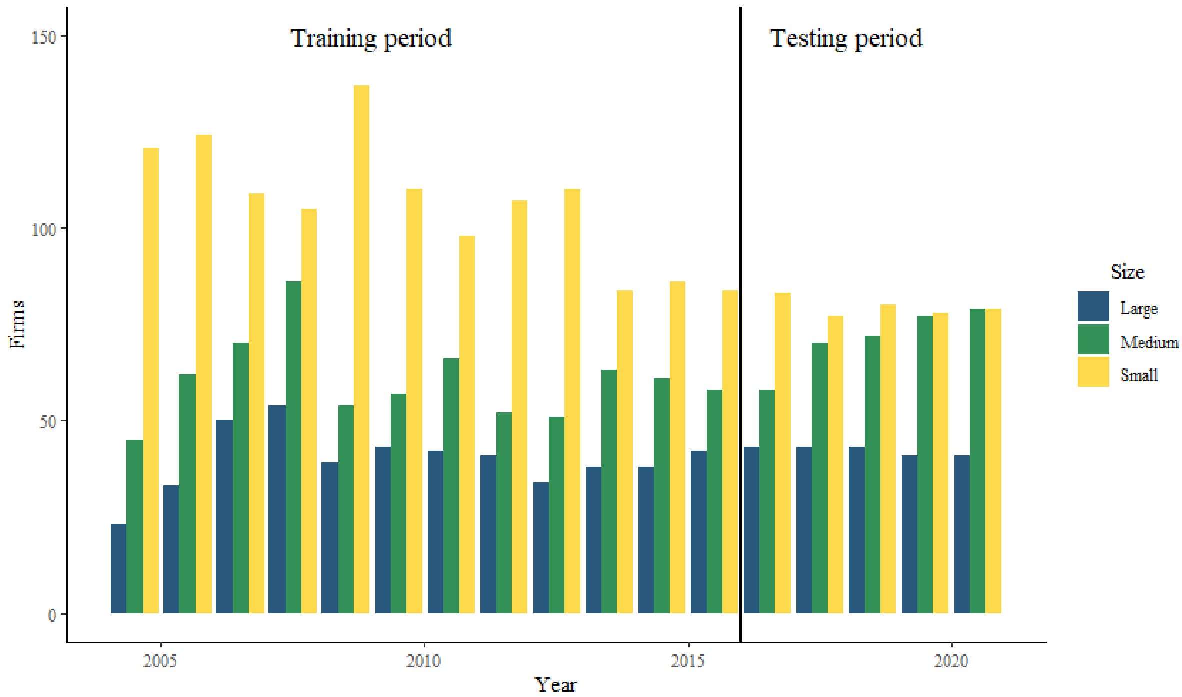
- Small: Market capitalization of less than 1.5 billion NOK.
- Medium: Market capitalization between 1.5 billion and 10 billion NOK.
- Large: Market capitalization of more than 10 billion NOK.

**Figure 5.3:** Firm size distribution in the sample
Figure 5.3 shows the distribution of the firms' market value in the sample, based on our definition of small, medium and large market cap.




Naturally, there are more small firms than large and medium firms, but the share of small firms is trending downwards throughout the sample and OSE. This could partly be because of inflation. Small and medium size firms are somewhat underrepresented in our testing sample, compared to the OSE. This is likely because of our 4-years of observations criteria. New listings are, most often, likely to be a small or medium sized firm, and these will not have had enough time to generate data and meet the criteria. consequently, they are excluded. This could introduce a survivorship-bias, as we are excluding all firms that may have gone bankrupt in 4 years or less. However, considering the small size of the Norwegian market and the shortness of the time frame, we consider the potential bias to be small and negligible.

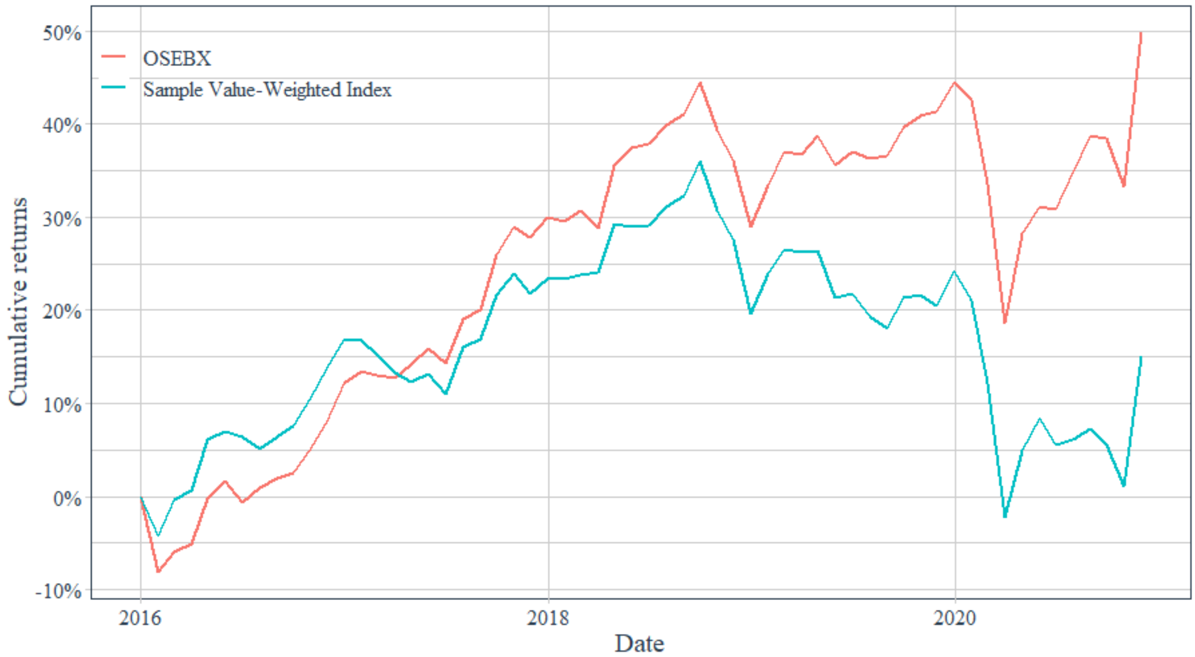
**Figure 5.4:** Firm size distribution on the Oslo Stock Exchange
Figure 3.4 shows the firm size distribution for the whole Oslo Stock Exchange using the same size definitions as in figure 3.3.




### 5.4.4 Returns

In figure 5.5 we show a comparison of the cumulative returns of taking a passive position in OSEBX versus a value-weighted index created using the firms in our sample. As seen in the figure, the OSEBX index outperforms our sample index with around 35% over the period. We find no other rational reasoning for OSEBX's outperformance other than the fact that it is a random creation from the data handling, due to data limitations. However, this givea our models something of an uphill-battle in the portfolio modelling section. While it is no small task to try to outperform the market, the fact that the stocks they have to choose from has generally underperform to begin with, makes the task much harder.

**Figure 5.5:** Sample Value-Weighted Index vs OSEBX
Figure 5.5 shows a comparison of the cumulative returns of taking a passive position in OSEBX versus a value-weighted index created using the firms in our sample. OSEBX is represented by red, while our sample value-weighted index is represented by light blue.



# 6 Results and Discussion

In this chapter we present and discuss the results. First, we present the prediction accuracies of the models and variables importances. Further, we show the returns generated by the machine learning model predictions and discuss possible viable investment strategies.

## 6.1 Accuracy

As discussed earlier, the ROC-curve can be used to show how the model predictions compare to a random guess. A random guess, seen as the dotted line in the figures in figure 6.1, has an area under the curve of 0.5. All our models have an AUC above 0.5, albeit only slightly, meaning that they perform better than the random guess at classifying whether the return will be positive or not.

The model that yields the best performance in our case is XGBoost, with an AUC of 0.5342. In most other studies an AUC of that value would be considered very poor. However, in our study it shows that there is a possibility of predicting the market, better than that of a random guess, based on the characteristics used. Of our other models, Random Forest and SVM perform best, with an AUC of just below 0.53 and just above 0.52, respectively. Followed by Neural network with three hidden layers with an AUC of 0.5140. Lastly, logistic regression and Neural network with a single and two hidden layers perform the worst, all with an AUC of under 0.51.

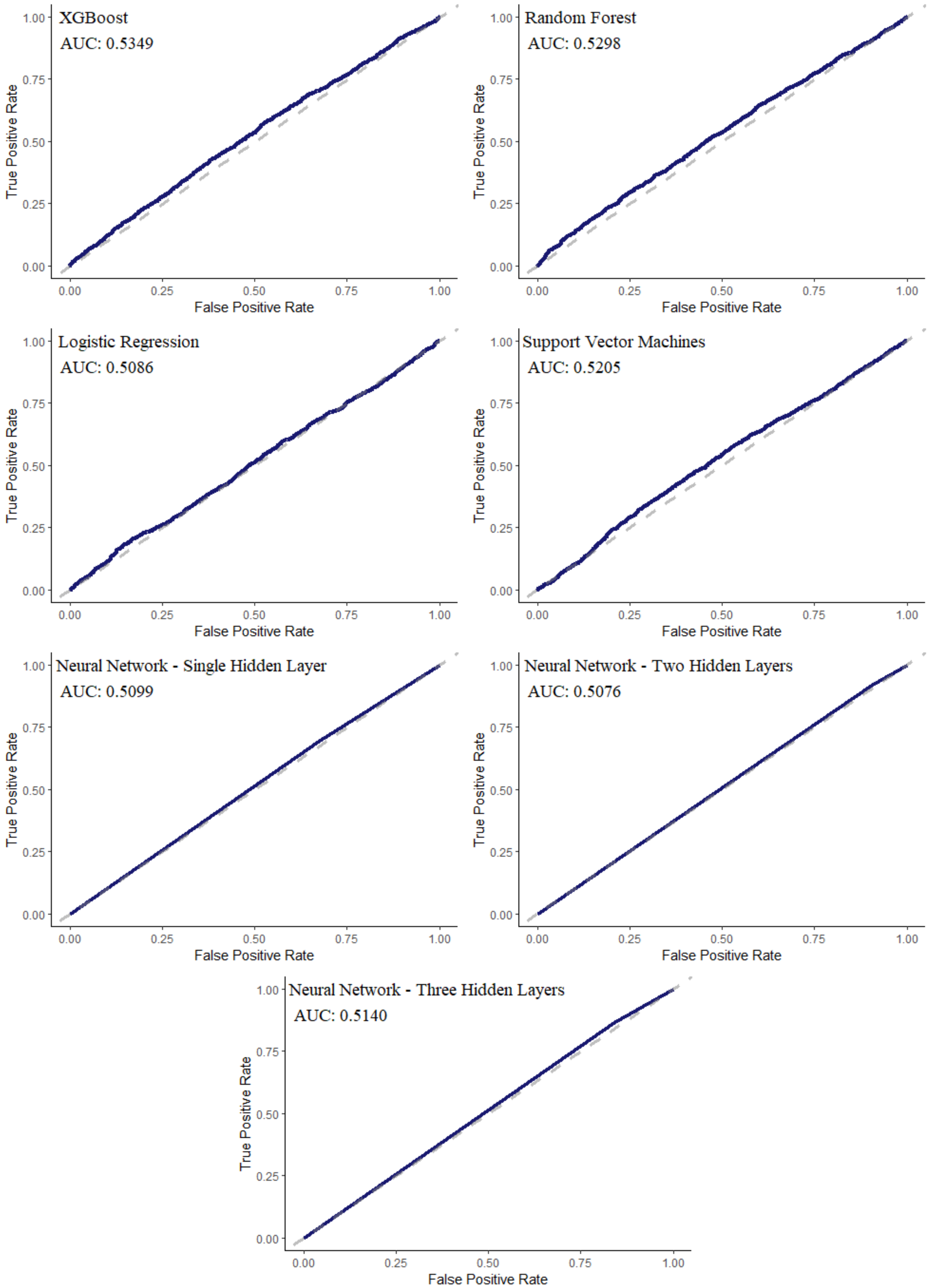
Looking at the model's accuracy we see similar results as from the area under the curve, with XGBoost as the best performer. The difference is that in terms of accuracy, both NN2 and NN3 perform similarly to that of RF and SVM. Logistic regression and NN1 is the worst performing models in terms of both accuracy and AUC-curve.

**Table 6.1:** Summary of performance measures
The table below shows a summary of the most important performance measures for all the models.

| | **XGB** | RF | SVM | LOG | NN1 | NN2 | NN3 |
|---|---|---|---|---|---|---|---|
| Accuracy | **0.5316** | 0.5219 | 0.5217 | 0.5105 | 0.5159 | 0.5209 | 0.5232 |
| Area Under Curve | **0.5349** | 0.5298 | 0.5205 | 0.5086 | 0.5099 | 0.5076 | 0.5140 |

**Figure 6.1:** ROC-Curve

The figure below shows the ROC-curve for each model. The curve for a given model is illustrated with a blue line



## 6.2 Variable Importance

In this study We group our stock characteristics into four categories. The first category is price trends/momentum and involves moving average over the short (ma price short), intermediate (ma price int) and long (ma price long) term, the 1month (mom1m), 6months (mom6m) and 12months (mom12m) momentum, the change in 6months momentum (chmom) and the maximum daily return (maxret). Our second category involves risk measures, which are represented by volatility of returns (volatility), idiosyncratic volatility of returns (idiovol), beta (beta) and squared beta (betasq). The third category includes valuation ratios and fundamentals. The characteristics included in this category is industry adjusted market value (mve_ia), the logarithm of market value (logmve), the return on assets (roaq) and the recent increase in earnings (nincr). The last category, liquidity, contains trading volume (dolvol), zero trading days (zerotrade), illiquidity (ill), turnover (turn) and the standard deviation of turnover (std_turn). The categories are similarly constructed to those in earlier studies in the U.S. (Gu et al., 2020) and Chinese stock market (Leippold et al., 2021).

From the figure 6.2, which shows the 20 most important factors for each model, we can see a distinct difference between most important factors in XGBoost and RF, compared to the other models. Their most important factors almost exclusively involve momentum factors and risk measures, whereas the other models have a higher importance on fundamentals and valuation ratios. Industry adjusted market value is ranked most important in logistic regression, SVM, neural network with 2 hidden layers and neural network with three hidden layers. For the models using decision trees, XGBoost and RF, moving average over a short time span is the most important.

**Figure 6.2:** Model Variable Importance
Figure 6.2 shows the twenty most important characteristics for each model, ordered from most important at the top, to least important at the bottom.

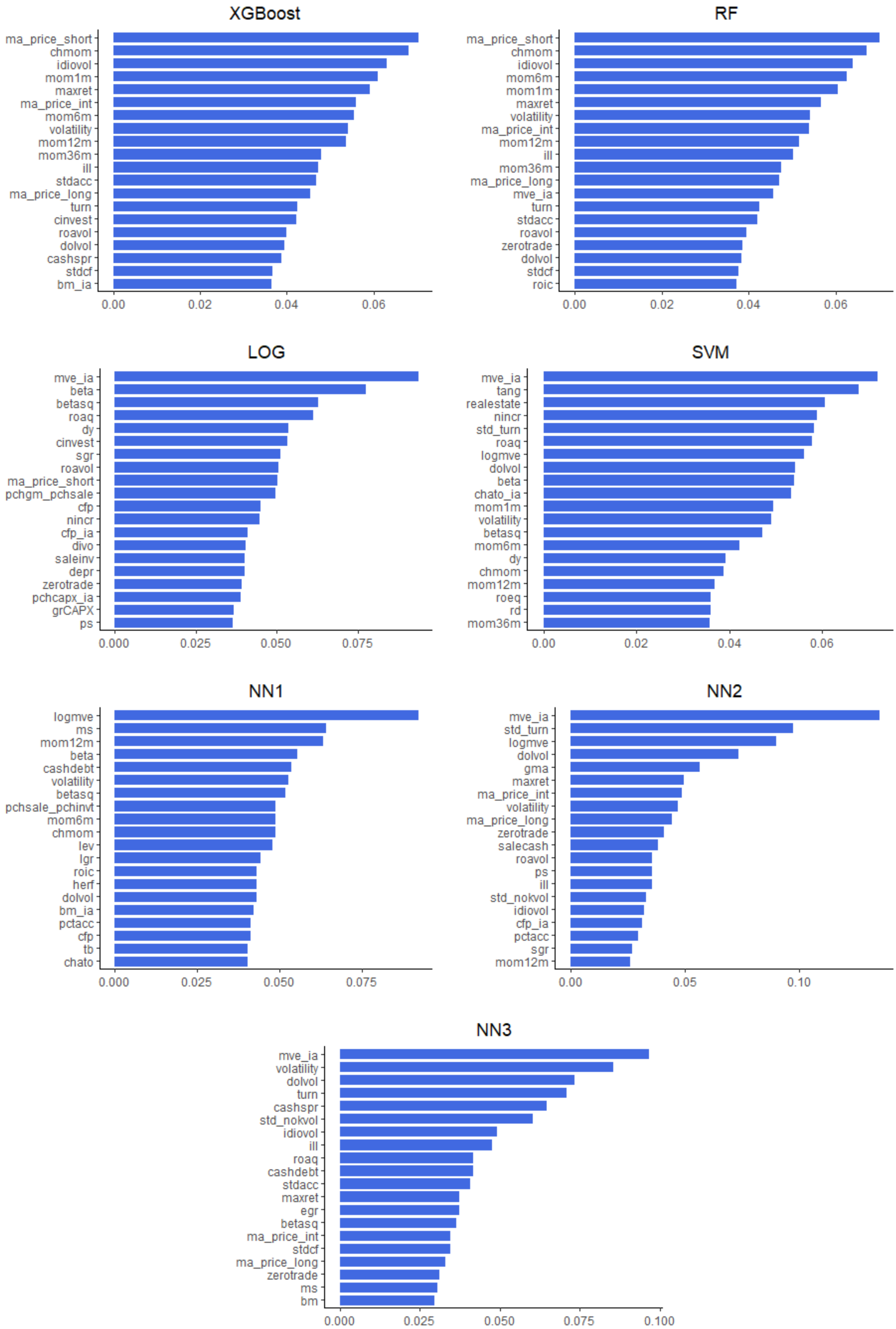
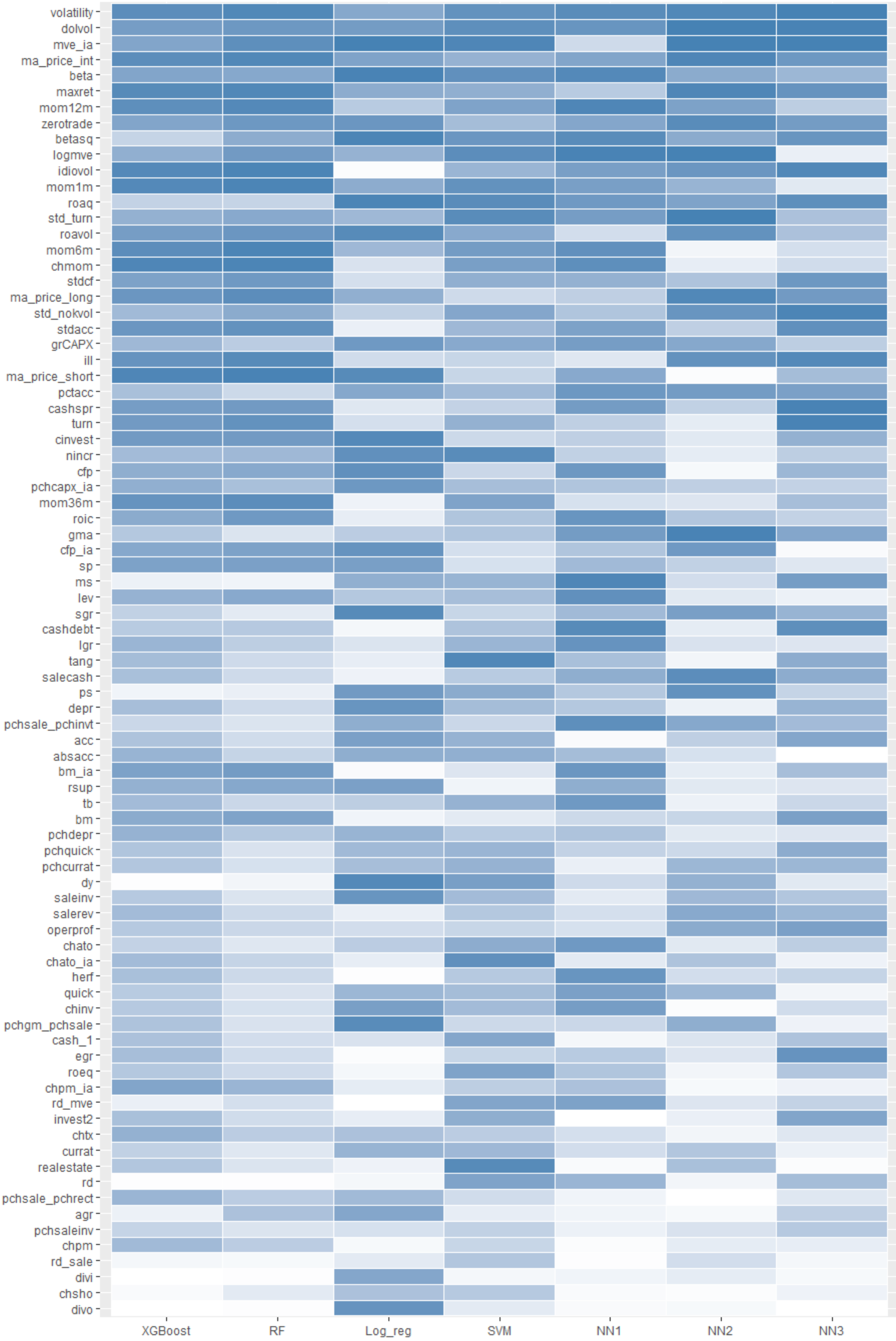


**Figure 6.3:** Variable Importance Comparison
Figure 6.3 shows all stock characteristics ranked by their combined importance. The order is given from creating a score for each model and using the sum of scores, with the most important variables at the top and least important at the bottom. The importance given to each characteristic, by a specific model, is represented by the color gradient, where we go from most important with dark blue to least important with white.



We find that the most important stock characteristics for prediction the Norwegian stock market is our risk measurements, with volatility scoring as the most important. This category is relevant in the predictions of all the models. When comparing for the second most important, we see a discrepancy depending on what models we value. If we value the models that perform better, which is XGBoost and RF, the price trends/momentum category is considered the second most important. But taking all models into account, you can easily argue for the category involving fundamentals and performance measures, as either the industry adjusted market value or the logarithm of market value is the most important in these other models. In our study, we have chosen to value the better performance higher, therefore considering price trends/momentum the second most important and fundamentals and valuation ratios the third most important. Volatility ranks the fourth of the importance categories, receiving a high importance in neural networks with two and three layers.

From the average rank comparison, in figure 6.2 below, we can see that our top 10 have an average importance rank of 18 in the American market and 31 in the Chinese market. This is an expected result, as the US and Norwegian market share more fundamental similarities. In both the US and Norwegian market, institutional investors play an important role, while the Chinese markets are primarily owned by retail investors. China also has tight regulations that contribute to different investor behaviour, e.g. the limited ability to short stocks.

Most of the variables have a similar importance to either the Chinese or American market, with the exceptions being the beta and beta squared that seems to have a higher importance in the Norwegian market. Maxret and logmve have an importance average between all markets of 8 and 6, respectively. Indicating that these characteristics are not only important but might also be less affected by differences between markets.

**Table 6.2:** VI comparison
Table 6.2 shows how the study's most important variables would rank in the earlier studies done on the American and Chinese market. e.g., we can see that volatility, the most important characteristic in our study, ranks 7th most important in the American market and 28th most important in the Chinese market. The "Average" column in the figure shows the average importance rank from all markets of that variable, which in the case of volatility is 12. To further compare the markets, we also have calculated the average importance rank of our top 10 variables in the other markets. Our intermediate moving average factors is not considered in the comparison, as it is not included in the other papers.

| Variables | Norway | USA | China | Average |
|---|---|---|---|---|
| volatility | 1 | 7 | 28 | 12 |
| dolvol | 2 | 8 | 27 | 12 |
| mve_ia | 3 | 50 | 74 | 42 |
| beta | 4 | 26 | 42 | 24 |
| maxret | 4 | 5 | 15 | 8 |
| mom12m | 5 | 3 | 39 | 16 |
| zerotrade | 6 | 30 | 2 | 13 |
| betasq | 7 | 27 | 48 | 27 |
| logmve | 8 | 2 | 9 | 6 |
| idiovol | 9 | 20 | 25 | 18 |
| Average Rank | | 18 | 31 | |

## 6.3 Portfolio Performance Analysis

In this section we perform a portfolio analysis of the models' predictions on the test set. We analyze one portfolio only taking long-positions, and one taking both long and short positions. For the long-only portfolios, positions are taken in the stocks that the model predicts have a higher than 50% chance of positive return for the next month. All the positions are given equal weight of 1/N, with N being the number of stocks predicted to have a positive return in the next month. In the long-short portfolio, the long positions are taken the same way as in the long-only portfolio, and the short positions are taken in the stocks that the model predicts have a higher than 50% chance of negative return in the next month. The positions are held for the entire given month. Then, new predictions are generated and both portfolios are rebalanced and repositioned based on the new stock picks. In addition, a zero-net investment strategy was implemented in the long-short portfolio. The intention for this was to keep leverage to a minimum to be able to meaningfully compare the two portfolio types. Furthermore, we compare the results to the monthly returns of the OSEBX in the test period, and a 1/N portfolio with all the stocks in the test sample. The latter portfolio shows the returns gained if we would take long positions in all the stocks in our test sample, rather than trying to pick individual winners.

### 6.3.1 Returns from Model predictions

A fundamental consideration in portfolio modelling is risk. Looking at returns without understanding the risk only provides half the picture. The standard deviation tells us the volatility of the return and gives us an indication of the associated risk. One way to evaluate returns in terms of risk-reward, is to consider the Sharpe ratio. We can risk-adjust the returns by using Sharpe ratio:

$$Annualized\ Sharpe\ Ratio = \frac{Avg.\ Return - Risk\text{-}Free}{Std.\ Deviation} \cdot \sqrt{12}$$

The Sharpe ratio tells us what the excess returns are, above that of the risk-free option, for every unit of risk associated with those returns. We have assumed a risk-free rate of 0.17%, which was the 1-month treasury bill rate as of March 31st, 2022.

In table 6.3 we report the performance of the portfolios on the test data, in the period January 2016 to November 2020, corresponding to 59 months. The table shows statistical summaries for the model portfolios and the benchmarks. "Total Return" shows the cumulative returns that the portfolio has accumulated throughout the period. "Avg. return" shows the average monthly returns the models generate. "Std. Dev." is the standard deviations of the monthly returns. "Sharpe ratio" is the Sharpe ratio. "Skewness" is skewness. "Kurtosis" is kurtosis. "Max DD" is maximum drawdown. "Max 1M Loss" is the largest monthly loss the portfolio has suffered in the period. "Avg. N" shows how many stocks, on average, are included in the portfolio per month.

All the long-only portfolios, with the exception for the logistic regression portfolio, outperform the OSEBX and the 1/N portfolios. The long-only portfolio constructed by the XGBoost model yields the highest return. This is consistent with it having the highest accuracy and AUC from table 6.1. It outperforms its benchmark by around 28% over the 5-year period, or around 0.5% each month. The logistic model yields the lowest return in both the long-only portfolio and the long/short portfolio. This is also the only model that has a negative return among the long-short portfolios.

**Table 6.3:** Gross returns summary
Descriptive summary table of the portfolio performances of the machine learning algorithms. The table reports the returns from the portfolios constructed from the model predictions on the test-sample. The test sample consist of 59 months of returns from January 2016 until November 2020. A long-only and long-short portfolio were constructed for each model. "Total Return" (%) is the total cumulative return achieved during the period. "Avg. Return" (%) shows the monthly average return. "Std. Dev." (%) is the standard deviation of the monthly returns. "Sharpe Ratio" is the annualized Sharpe ratio. "Skewness" is skewness. "Kurtosis" is kurtosis. "Max DD" (%) is the maximum drawdown the portfolio experiences. "Max 1M Loss" (%) is the largest monthly loss the portfolio suffers during the period. "Avg. N" shows the average number of stocks that are included in the portfolio each month.

| 2016 - 2020 | *Benchmarks* | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| *rf = 0.17%* | OSEBX | 1/N | XGB | RF | Log | SVM | NN1 | NN2 | NN3 |
| **Long-Only** | | | | | | | | | |
| Total Return | 49.83 | 47.43 | 77.47 | 65.86 | 35.99 | 59.00 | 53.18 | 62.15 | 62.51 |
| Avg. Return | 0.84 | 0.80 | 1.31 | 1.12 | 0.61 | 1.00 | 0.90 | 1.05 | 1.06 |
| Std. Dev. | 4.46 | 5.27 | 6.53 | 6.06 | 4.43 | 4.34 | 5.11 | 6.32 | 4.85 |
| Sharpe Ratio | 0.52 | 0.42 | 0.61 | 0.54 | 0.34 | 0.66 | 0.50 | 0.48 | 0.64 |
| Skewness | -0.28 | -0.54 | -0.45 | -0.19 | -1.55 | -1.41 | -0.69 | 0.01 | -0.92 |
| Kurtosis | 3.62 | 4.19 | 5.68 | 4.41 | 6.05 | 6.02 | 3.02 | 4.22 | 3.78 |
| Max DD | 24.63 | 33.13 | 38.25 | 33.35 | 32.76 | 25.96 | 28.16 | 29.61 | 26.66 |
| Max 1M Loss | -14.83 | -20.64 | -26.69 | -23.07 | -19.83 | -18.88 | -19.39 | -22.51 | -18.93 |
| Avg. N | | 66.79 | 29.05 | 27.51 | 24.54 | 24.59 | 21.10 | 5.78 | 14.27 |
| **Long-Short** | | | | | | | | | |
| Total Return | | | 55.39 | 34.75 | -19.20 | 19.21 | 7.79 | 17.75 | 26.18 |
| Avg. Return | | | 0.94 | 0.59 | -0.33 | 0.33 | 0.13 | 0.30 | 0.44 |
| Std. Dev. | | | 3.91 | 3.41 | 4.52 | 4.59 | 4.48 | 5.21 | 4.12 |
| Sharpe Ratio | | | 0.68 | 0.43 | -0.38 | 0.12 | -0.03 | 0.09 | 0.23 |
| Skewness | | | -0.04 | 0.32 | -2.83 | -3.53 | -3.38 | -0.58 | -1.62 |
| Kurtosis | | | 0.17 | -0.10 | 13.21 | 18.50 | 18.58 | 2.27 | 4.18 |
| Max DD | | | 22.78 | 20.68 | 29.17 | 29.70 | 27.95 | 25.15 | 23.13 |
| Max 1M Loss | | | -10.03 | -6.48 | -25.13 | -26.93 | -26.51 | -18.54 | -16.61 |
| Avg. N | | | 66.79 | 66.79 | 66.79 | 66.79 | 66.79 | 66.79 | 66.79 |

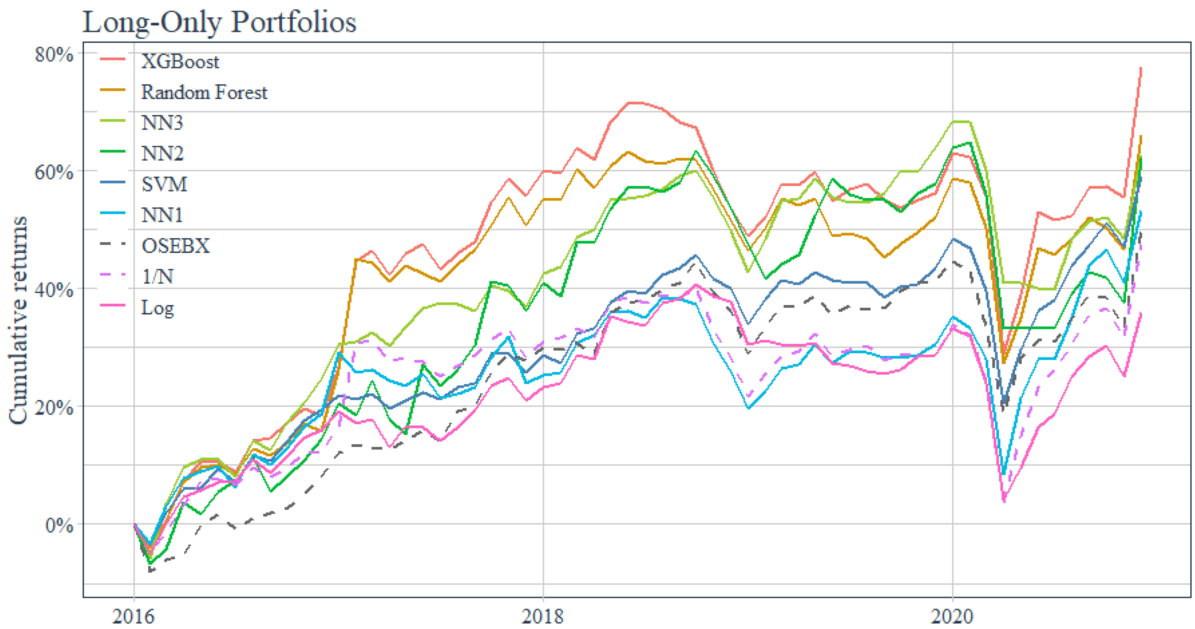
Interestingly, the long-short portfolios all underperform compared to their long-only counterparts. This means that the short part of the portfolios have negative returns. So, while it appears that the models are can winning stocks, to some degree, they are less able to do so for losing stocks. This is inconsistent with the findings of Leippold et al. (2021), where all the long-short portfolios outperform the long-only portfolios. This difference could be caused by the difference in portfolio structuring. Leippold et al. constructs value-weighted portfolios, whilst we construct equally-weighted portfolios. In addition, Leippold et al.'s portfolio only includes stocks based on the highest and lowest deciles in predicted next month's returns. In our case, with equally-weighted portfolios and not excluding any stocks, our short-portfolios yield negative returns.

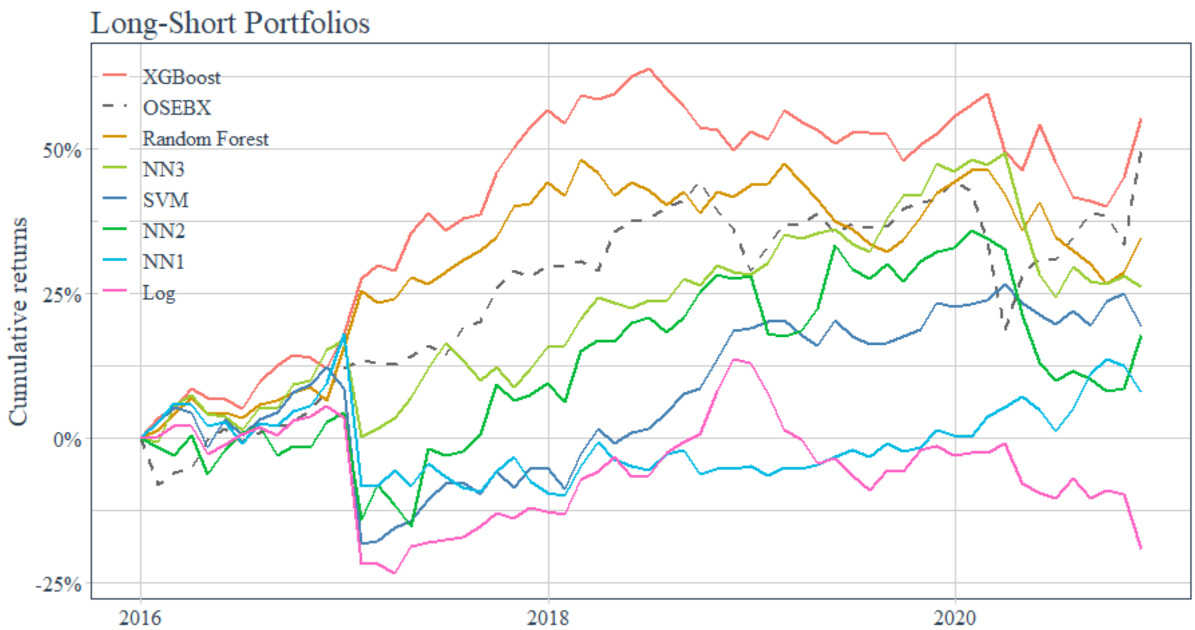
After risk-adjusting the returns, two less of the long-only portfolios outperforms the OSEBX. All the long-only portfolios except logistic regression, NN1 and NN2 have higher Sharpe ratios than the index. Even though XGBoost has the highest returns, it also carries the highest standard deviation, or risk. This is also reflected in the maximum drawdown and maximum 1 month loss, where XGBoost shows the most volatile numbers. Interestingly, the portfolio with the highest Sharpe ratio is the portfolio generated by the Support Vector Machine algorithm, which has the 5th highest overall return. This is because of the portfolio's relatively low standard deviation compared to the other portfolios.

As for the long-short portfolios, the only portfolio that outperforms its benchmark is the XGB-portfolio. It is also the only long-short portfolio that has a higher Sharpe ratio than its long-only counterpart.

In figure 6.4 we show a graphical representation of the cumulative returns of the portfolios throughout the testing period. The legends are sorted from highest to the lowest end-of-period return. Solid lines indicate machine learning portfolios and dashed lines indicates benchmark portfolios.

**Figure 6.4:** Gross portfolio returns
The graphs below show the cumulative portfolio returns over time for each machine learning portfolio, in addition to the OSEBX and 1/N benchmarks. Legends are sorted from highest to lowest "Total Return". Solid lines indicate machine learning portfolios and dashed lines indicates benchmark portfolios.




### 6.3.2 Effects of transaction costs and liquidity

Trading and investing are not free and can be expensive. Transaction costs decreases returns of any investment and therefore increases the associated risk. The returns generated by the machine learning portfolios presented in the previous sub-chapter are raw, and include no considerations for trading costs when repositioning, interest rates when shorting or other hindrances. Therefore, they are only theoretical and unachievable. That being the case, we will in this sub-chapter consider several market frictions to estimate the real, achievable returns.

As a market player, we affect the market. Whenever we buy or sell stock, we influence the supply and demand. The degree to which we influence depends on our size, in terms of capital. The returns presented in the previous sub-chapter are calculated using the closing price, this means we assume that we will have no influence on the supply or demand when entering positions or exiting them. This is in reality not the case. We will first assume that we are a big market player so that we can have a relatively large influence on the price. One way we can try to adjust for this is to use the volume-weighted average price (VWAP) of the stock price. This is the price where the trading activity was the highest, in terms of volume of shares. Given that we assume there is a possibility our trading activity can move the market, we can assume that the price we buy and sell at is the volume-weighted average price.

Transaction costs is another consideration we will account for. Brokerage houses, or platforms, will take a fee for executing the trade we task them to. We incur this cost to our portfolio every time we buy or sell shares. The size of this fee can vary depending on size and frequency. Brokers like Nordnet and DNB Markets list a fee of 0.035% (Nordnet, 2022) and 0.04% (DNB, 2022) respectively as of May 2022, for clients with portfolios of larger sizes. So-called "Private Banking" clients. We are being conservative when accounting for this consideration and will use a large fee. We will assume a 0.2% fee for buying and selling shares. 5 times larger than DNB Markets' rate. Furthermore, we will assume that the entire portfolio is replaced each month. Meaning, if a model predicts that a stock will have a positive return next month, and that particular stock is already in the portfolio, it will still be sold and bought again. So that the maximum trading costs that can be sustained, are incurred on the portfolio each month.

Thirdly, we will exclude the smallest 25% stocks by market capitalization. This is because we assume that their trading liquidity is too small for us to participate. Meaning, their daily trading activity does not have enough volume for us to fill the positions in the portfolios the models generate.

Lastly, for the long/short portfolios we introduce an interest rate cost for borrowing the shares we short. Nordnet offers an interest rate of 4.5% for borrowing Norwegian stocks. We are somewhat conservative and assume an interest rate of 5% in this case.

**Table 6.4:** Effect and net return summary
The table below shows how the different considerations affect the monthly gross average returns. All numbers in %. "Gross average returns" is the monthly average return from the ML-portfolios in table 6.3 without any consideration, returns calculated using the closing prices. "VWAP" shows the gross average return after using the volume-weighted average price to calculate the return, instead of the closing price. "Brokerage fee (20bp)" shows the gross average return after considering a 0.2% transaction fee for each buy and sell in the portfolio. "Interest rate (5%)" shows the gross average return after considering a 5% interest rate for borrowing the shares for shorting. "Exluding smallest 25%" shows the gross average return after removing the smallest 25% companies from the sample. "Net average return" shows the gross average return after simultaneously taking all the considerations above into account. "Avg. Δ" shows the average difference between "gross average return" and "VWAP", "Brokerage fee (20bp)", "Interest rate (5%)" or "Excluding smallest 25%".

| | | *Machine Learning Portfolios* | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 | |
| **Long-Only** | | | | | | | | | |
| Gross average return | 0.80 | 1.31 | 1.12 | 1.00 | 0.61 | 0.90 | 1.05 | 1.06 | Avg. |
| *Returns after considering:* | | | | | | | | | Δ |
| - VWAP | 0.72 | 1.18 | 1.03 | 0.97 | 0.64 | 0.81 | 1.12 | 1.00 | -0.05 |
| - Brokerage fee (20 bp) | 0.40 | 0.91 | 0.71 | 0.60 | 0.21 | 0.50 | 0.65 | 0.66 | -0.40 |
| - Excluding smallest 25% | 0.48 | 0.42 | 0.48 | 0.70 | 0.56 | 1.00 | 0.75 | 1.00 | -0.31 |
| *Returns after all considerations:* | | | | | | | | | |
| Net average return | -0.01 | -0.10 | 0.03 | 0.30 | 0.18 | 0.54 | 0.36 | 0.51 | -0.76 |
| **Long-Short** | | | | | | | | | |
| Gross average return | | 0.94 | 0.59 | 0.33 | -0.33 | 0.13 | 0.30 | 0.44 | Avg. |
| *Returns after individual effects:* | | | | | | | | | Δ |
| - VWAP | | 0.86 | 0.59 | 0.39 | -0.17 | 0.12 | 0.46 | 0.46 | 0.04 |
| - Brokerage fee (20 bp) | | 0.54 | 0.19 | -0.08 | -0.72 | -0.27 | -0.10 | 0.04 | -0.40 |
| - Interest rate (5%) | | 0.52 | 0.17 | -0.09 | -0.74 | -0.28 | -0.12 | 0.03 | -0.42 |
| - Excluding smallest 25% | | -0.03 | 0.02 | 0.46 | 0.10 | 0.80 | 0.33 | 0.77 | 0.01 |
| *Returns after all effects:* | | | | | | | | | |
| Net average return | | -0.93 | -0.74 | -0.24 | -0.56 | 0.00 | -0.42 | -0.08 | -0.77 |

Table 6.4 shows how the considerations we account for affect the monthly average returns presented in table 6.3. The table shows how the monthly average return, which we call "Gross average return" in the table, change after implementing each consideration. All the numbers in the table are percentages. "VWAP", "Brokerage fee (20bp)", "Interest rate (5%)" and "Excluding smallest 25%" show how "Gross average return" change after only

considering their respective factor alone. On the right-hand side of the table, we show the average changes across the portfolios. These are the average differences between the gross average return and the return after implementing their respective factor. "VWAP" shows the gross average returns after using the volume-weighted average price to calculate the returns, rather than the closing price. We can see that the returns mainly decrease after using the volume-weighted average price. The returns of 6 out of 8 portfolios decrease after using VWAP. The exceptions are the Log and NN2 portfolios, where the returns increase. On average, the monthly returns decrease by 0.05% in the long-only portfolios. The XGB portfolio is the most affected, as its monthly returns decrease by 0.13%. By looking at the changes in the 1/N portfolio, we can see how the considerations affect returns on the whole test sample. This gives an overall view of the effects, considering the different variations of number of stocks in each portfolio. The returns decrease from 0.8% monthly return to 0.72% in the 1/N portfolio, indicating that on average, the volume-weighted average prices tend to be lower than the closing prices.

"Brokerage fee (20bp)" shows the gross average monthly return after considering a transaction fee of 0.2% for each position that is bought or sold each month. The transaction cost of 0.20% decreases the returns on average by 0.40%, which is not surprising given that the fee is incurred twice each month. At the start of the month, when entering a position, and at the end of the month, when exiting the position. The returns decrease by a little bit more than 0.4%, because the fee is also paid on the returns when we sell the positions at the end of each month. However, when rounding down to two decimals, the fee is 0.40%.

"Excluding smallest 25%" show the gross average return after removing the smallest 25% companies, by market capitalization. The returns in the long-only portfolios are negatively affected by this constraint. This is consistent with literature, as smaller companies carry higher risk and therefore have a higher expected return than bigger companies (Fama and French, 2015). Therefore, by excluding these from our portfolio, the total expected return should also decrease. On average the reduction in returns is 0.31%, for the long-only portfolios. The XGB and RF portfolios are the most affected, as their returns are reduced by 0.89% and 0.64% respectively. The long-short portfolios are on average unaffected by this constraint. Though, again the tree-based portfolios, XGB and RF, are the most negatively affected. Their returns are reduced by 0.97% and 0.57%. However, they are also the only portfolios negatively affected. The rest of the portfolios' returns are increased because of this constraint. This is likely caused by the fact that XGB and RF have a higher share of small companies in their portfolios than the rest. An overview of the portfolio returns, Sharpe ratios and average number of stocks for the portfolios, after excluding certain firm sizes, is given in table A3.2 and A3.3 in the appendix. The change in number of stocks in the portfolios after excluding the largest 25% firms, is smaller for the tree-based portfolios than the others. This could explain the difference in effect of this constraint. The tree-based portfolios have a smaller degree of small companies in their short-portfolios than the others. This gives them the highest return when there are no constraints, but after removing the smallest quartile of the companies, they have the lowest return. This is also true for them in the long-only portfolios. The returns that are removed can be seen in table A3.2 and A3.3 in "Bottom 25%". This could indicate that the tree-based models perform better on small stocks.

"Interest rate (5%)" shows the gross average return after considering a yearly 5% interest rate cost for borrowing the shares shorted in the portfolio. "Net average return" shows the gross average return after considering VWAP, brokerage fee, excluding smallest 25% and interest rate simultaneously. On average the return decrease by 0.76% for the long-only portfolios and 0.77% for the long-short portfolios, after all considerations. We can see that all the excess returns from the gross average returns are removed by these constraints. This means that investors who operate with these specific constraints are better off investing in the index, which earned an average monthly return of 0.84% in our sample period. These constraints are one of the disadvantages that institutional investors face, especially given that the Norwegian stock market's trading volume is rather small compared to Nasdaq or S&P 500. We find that having all constraints imposed simultaneously removes the excess returns for any investor but removing one or several of them could open the door for excess returns. Investors likely cannot invest without fees, but if they can operate with a lower transaction fee than 0.2%, which they likely can, they can improve the expected returns. Using VWAP, we can see that this has some, but little effect on the returns. We therefore consider excluding the smallest companies to be the biggest hindrance to big investors and funds. If they can invest in smaller companies, we see that they can improve their expected returns. This is, however, an advantage of retail investors. Investing in small companies is often not a constraint directly for retailers. So, while the returns from the machine learning portfolios might be unachievable for institutional investors, they might not be for small, retail investors.

In table 6.5, we show the returns from the machine learning portfolios with a retail perspective. Its structure is similar to table 6.4. The only exceptions are "Excluding top 25%", where we instead remove the largest quartile of firms rather than the smallest, and "Excluding top 50%" where we remove the two largest quartiles. We also show the average number of stocks that are included in the portfolios each month in "Avg. N". "NAR w/o top 25%" and "NAR w/o top 50%" are abbreviations for "Net Average Return Without Top 25%" and "Net Average Return Without Top 50%", meaning the average monthly returns for the portfolios without the top quartile and without the top two quartile firm sizes, in addition to using the VWAP and brokerage fees.

By excluding the largest quartile, we see an average increase in gross returns of 0.37% in the long-only portfolios and 0.32% in the long-short portfolios. A relatively large improvement, which is improved further by excluding the top two quartiles. After excluding the top two quartiles, the long-only and long-short returns are increased by 0.65%. To calculate the net average returns we are still using the value-weighted average price. The motivation for this is twofold: It is to ensure that the strategy could potentially be viable for even somewhat large retail investors, that have at least some ability to move markets. Secondly, it is a robustness test considering that it is easier to have an influence on smaller companies' stock prices, as their stock liquidity is usually lower. The same transaction fee of 0.2% is also considered, which again is a rather large and conservative fee. Many retail investors would likely enjoy a smaller fee, but again, this is to ensure viability for investors with worse market terms.

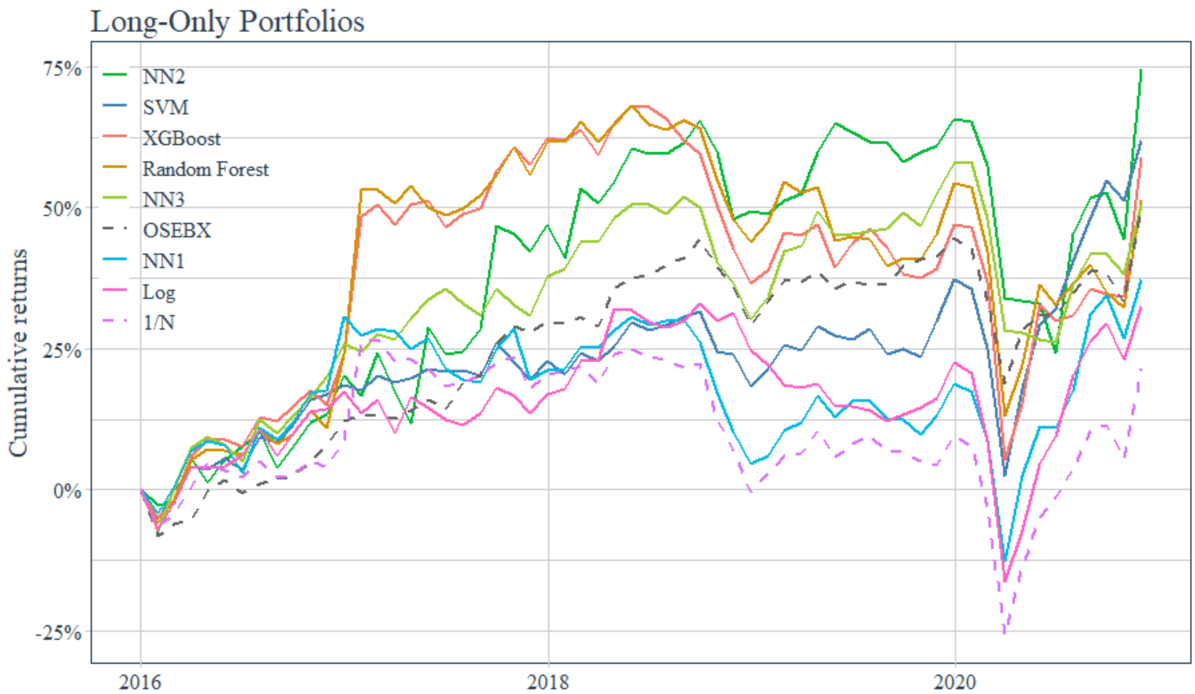
**Table 6.5:** Effect and Net Return Summary - Retail Perspective
The table below shows how the different considerations affect the monthly gross average returns. All numbers in %, except "Avg.N" which is a numeric. "Gross average returns" is the monthly average return from the ML-portfolios without any consideration, returns calculated using the closing prices. "VWAP" show the gross average return after using the volume-weighted average price to calculate the return, instead of the closing price. "Brokerage fee (20bp)" show the gross average return after considering a 0.2% transaction fee for each buy and sell in the portfolio. "Interest rate (5%)" shows the gross average return after considering a 5% interest rate for borrowing the shares for shorting. "Exluding top 25%" show the gross average return after removing the largest 25% companies from the sample. "Exluding top 50%" show the gross average return after removing the largest 50% companies from the sample. "NAR w/o top 25%" is the Net Average Return and shows the gross average return after simultaneously having "VWAP", "Brokerage fee (20 bp)" and "Excluding top 25%" as constraints. "Avg. N" shows the average number of stocks in the portfolio. "NAR w/o top 50%" is the Net Average Return and shows the gross average return after simultaneously having "VWAP", "Brokerage fee (20 bp)" and "Excluding top 50%" as constraints. "Avg. Δ" shows the average difference between "Gross average return" and the item on that specific row.

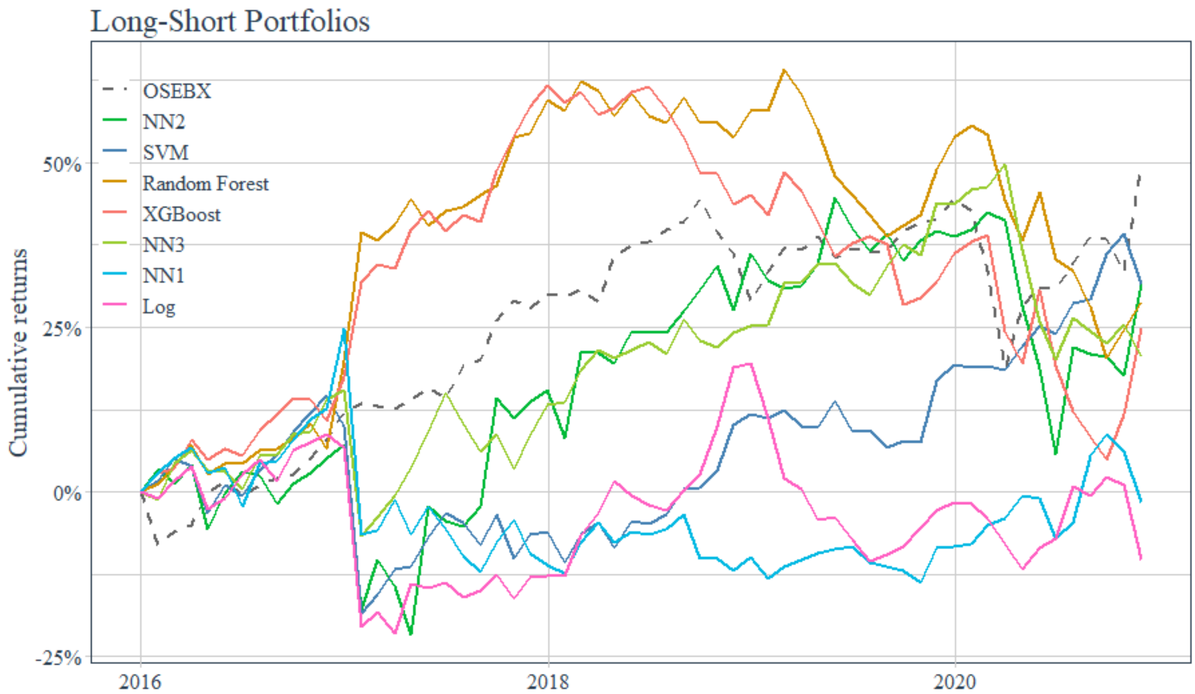
| | | *Machine Learning Portfolios* | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 | |
| **Long-Only** | | | | | | | | | |
| Gross average return | 0.80 | 1.31 | 1.12 | 1.00 | 0.61 | 0.90 | 1.05 | 1.06 | Avg. |
| *Returns after individual effects:* | | | | | | | | | Δ |
| - VWAP | 0.72 | 1.18 | 1.03 | 0.97 | 0.64 | 0.81 | 1.12 | 1.00 | -0.05 |
| - Brokerage fee (20 bp) | 0.40 | 0.91 | 0.71 | 0.60 | 0.21 | 0.50 | 0.65 | 0.66 | -0.40 |
| - Excluding top 25% | 0.90 | 1.63 | 1.45 | 1.59 | 1.02 | 1.20 | 1.62 | 1.41 | 0.37 |
| - Excluding top 50% | 0.94 | 2.29 | 2.43 | 1.94 | 1.21 | 0.57 | 2.00 | 1.67 | 0.65 |
| *Returns after all effects:* | | | | | | | | | |
| NAR w/o top 25% | 0.36 | 1.00 | 0.87 | 1.05 | 0.55 | 0.63 | 1.26 | 0.86 | -0.16 |
| - Avg. N | 50.10 | 20.53 | 18.27 | 11.78 | 10.29 | 14.05 | 3.50 | 9.54 | |
| NAR w/o top 50% | 0.41 | 1.51 | 1.71 | 1.51 | 0.76 | -0.10 | 1.87 | 0.99 | 0.10 |
| - Avg. N | 33.41 | 12.37 | 8.54 | 1.54 | 3.31 | 7.05 | 1.34 | 4.08 | |
| **Long-Short** | | | | | | | | | |
| Gross average return | | 0.94 | 0.59 | 0.33 | -0.33 | 0.13 | 0.30 | 0.44 | Avg. |
| *Returns after individual effects:* | | | | | | | | | Δ |
| - VWAP | | 0.86 | 0.59 | 0.39 | -0.17 | 0.12 | 0.46 | 0.46 | 0.04 |
| - Brokerage fee (20 bp) | | 0.54 | 0.19 | -0.08 | -0.72 | -0.27 | -0.10 | 0.04 | -0.40 |
| - Interest rate (5%) | | 0.52 | 0.17 | -0.09 | -0.74 | -0.28 | -0.12 | 0.03 | -0.42 |
| - Excluding top 25% | | 1.28 | 0.95 | 0.95 | -0.47 | 0.42 | 0.75 | 0.78 | 0.32 |
| - Excluding top 50% | | 2.11 | 2.05 | 1.09 | 0.24 | -0.49 | 1.03 | 0.91 | 0.65 |
| *Returns after all effects:* | | | | | | | | | |
| NAR w/o top 25% | | 0.42 | 0.49 | 0.53 | -0.17 | -0.03 | 0.54 | 0.35 | -0.04 |
| - Avg. Long-N | | 20.53 | 18.27 | 11.78 | 10.29 | 14.05 | 3.50 | 9.54 | |
| - Avg. Short-N | | 29.58 | 31.83 | 39.32 | 38.32 | 36.05 | 36.59 | 40.56 | |
| NAR w/o top 50% | | 1.07 | 1.17 | 0.46 | -0.77 | -1.69 | 0.39 | -0.05 | -0.26 |
| - Avg. Long-N | | 12.37 | 8.54 | 1.54 | 3.31 | 7.05 | 1.34 | 4.08 | |
| - Avg. Short-N | | 21.03 | 24.86 | 31.86 | 30.1 | 26.36 | 32.07 | 29.32 | |

The net average returns when excluding the top quartile and top two quartiles improve from the net average returns reported in table 6.4. After implementing these exclusions, 5 out of 7 of the long-only portfolios outperform the index. However, only 2 of the long-short portfolios do. The top long-only performer is the NN2 portfolio. Having said that, the number of stocks included in this portfolio averages at 3.5. This is because our neural network models have a higher tendency to predict negative returns, so it will include few stocks in its long-only portfolio. Having only 3-4 stocks in one's portfolio is considered risky and undiversified. Therefore, we consider this portfolio to be unrealistic. If we ignore NN2, the portfolio with the highest return is the SVM portfolio, closely followed by the XGB portfolio. Even though SVM had the third highest return before considering VWAP and transaction fee. This means that the reason SVM has a higher return after considering all constraints, is that it has a lower VWAP-cost than XGB. This could be explained by the fact that the SVM portfolio consists of a higher degree of larger sized firms than the portfolios of XGB and RF. As it is likely that larger firms will have a lower volatility in their stock price, and therefore the difference between the closing price and the value-weighted average price is smaller, the cost will be lower.

The improvements in net average returns increase even further when excluding the top two quartiles, all though with a caveat. The portfolios that are skewed towards larger sized firms now contain few total positions after excluding the top two quartiles. This can be observed in the changes in "Avg. N". So naturally, as we are excluding firms from our "pool" of possible picks, we are introducing the possibility of poor diversification and increase the riskiness of the portfolios. This is especially true for the SVM and Log portfolios, as they contain mainly firms with sizes in the top two quartiles category. Statistical summary tables, similar to table 6.3, for the portfolios after excluding quartiles, are included in the appendix i tables A3.3 and A3.4. After excluding the top two quartiles the average number of firms in their long-only portfolio, respectively, decrease from around 24 to 1.5 and 3.31. Having limited positions could be considered very risky, and therefore not realistic as a potential investment strategy. This might simply be a sampling issue, as our test-sample contains a smaller degree of small companies, as presented in figure 5.3. However, the portfolio returns in these portfolios should be taken with a heavy grain of salt, as they could be overperforming because of having so few stocks in their portfolios. This is the advantage of the long-short portfolios in this case. They, by design, include more stocks in the monthly portfolios, as all the stocks that are not included in the long-portfolio are predicted to have negative returns, and will therefore be shorted.

**Figure 6.5:** Net portfolio returns after excluding top quartile
Graphical representation of the NAR w/o 25% cumulative returns of gained over the period for each machine learning portfolio, in addition to the OSEBX and 1/N benchmarks. Legends are sorted from highest to lowest "Total Return". Solid lines indicate machine learning portfolios and dashed lines indicates benchmark portfolios.



The net average returns, after excluding the top two quartiles for the long-short portfolios, differ vastly. Only XGB and RF outperform the index, though at a smaller degree than their long-only counterparts. This is caused by the fact that the short portfolios have negative returns, in addition to the extra cost incurred on the portfolio for shorting.

The XGB, RF and SVM long-only portfolios could be viable options for retail investors with fewer constraints and less market frictions. In the next sub-chapter we investigate the reason for the excess returns.

## 6.4 Explanation of the returns

Bernt Arne Ødegaard has a public database of asset pricing data for the Oslo Stock Exchange. The database contains similar asset pricing factors as those developed by Fama and French, only for Norwegian data. The factors are the three Fama-French factors: Small-Minus-Big (SMB), High-Minus-Low (HML) and Up-Minus-Down (UMD), Carhart's momentum factor: prior one-year return (PR1YR), and Næs, Skjeltorp and Ødegaard's liquidity factor: Liquidity (LIQ). Table 6.6 shows the results of a regression analysis of the net returns on the factor returns. The results show a positive alpha for all models, except Log and NN1. However, none of the alphas are significant at the 5% level. Looking at the long-only portfolios, we can see from the market beta, which is significant across all models, that most of our portfolios are moving more than that of the market. This could be a result of our portfolios holding a small number of stocks at times, which would increase their risk and volatility, compared to that of the market. In the long-short portfolios, we can see that most of them have a negative market beta, meaning they have an inverse relation to the market. Interestingly, XGBoost and RF still have a positive market beta. Likely because these portfolios have more stocks in the long part of their portfolios. Only XGBoost, Log and NN3 of the long-short portfolios have a market beta that is significant at the 5% level. All our SMB betas, that are significant at the 5% level, are positive. As we have seen earlier, this is a result of the model's preferences for small-cap stocks. Our portfolios are therefore more sensitive to movements in the small-cap market. For the momentum factors PR1YR and UMD we receive no betas that are significant at the 5% level. For our liquidity beta, we get positive and significant levels for XGBoost and RF. Here liquidity is calculated from the bid-ask spread. The p-values for the table is included in table A3.7 in the appendix.

**Table 6.6:** Portfolio Factor Exposure
The table shows the results of a regressing the monthly portfolio returns on Ødegaard's factor returns. "Alpha" is the intercept. "Market beta" is the correlation coefficient between the portfolio and the market. "SMB", "HML", "PR1YR", "UMD", "LIQ" shows the correlation coefficients between the portfolios and the associated factor. "Adj. $R^2$" is the adjusted $R^2$. The *-symbol indicates significance at the 5% level.

| | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 |
| **Long-Only** | | | | | | | | |
| Alpha | -0.005 | 0.002 | 0.002 | 0.001 | 0.000 | -0.001 | 0.008 | 0.008 |
| Market beta | 1.359* | 1.630* | 1.494* | 1.147* | 0.879* | 1.060* | 1.153* | 0.863* |
| SMB | 0.301* | 0.336 | 0.444 | 0.556* | 0.441* | 0.649* | -0.088 | 0.132 |
| HML | 0.084 | 0.238 | 0.291 | -0.054 | 0.051 | 0.135 | 0.093 | 0.069 |
| PR1YR | -0.030 | -0.329 | -0.264 | -0.050 | -0.075 | -0.157 | -0.016 | -0.368 |
| UMD | 0.023 | 0.218 | 0.080 | 0.065 | 0.057 | 0.144 | 0.137 | 0.255 |
| LIQ | 0.412* | 0.600* | 0.555* | 0.166 | -0.223 | -0.086 | -0.048 | -0.002 |
| Adj. R2 | 0.756 | 0.667 | 0.628 | 0.725 | 0.562 | 0.665 | 0.372 | 0.568 |
| **Long-Short** | | | | | | | | |
| Alpha | | 0.012 | 0.010 | 0.008 | 0.005 | 0.005 | 0.012 | 0.015 |
| Market beta | | 0.470* | 0.216 | -0.317 | -0.605* | -0.418 | -0.237 | -0.545* |
| SMB | | 0.016 | 0.212 | 0.343 | 0.215 | 0.457 | -0.382 | -0.163 |
| HML | | 0.239 | 0.313* | -0.189 | -0.040 | 0.069 | 0.010 | -0.036 |
| PR1YR | | -0.458 | -0.320 | -0.058 | -0.001 | -0.189 | 0.013 | -0.391 |
| UMD | | 0.307 | 0.052 | 0.085 | 0.016 | 0.183 | 0.131 | 0.267 |
| LIQ | | 0.364 | 0.246 | -0.377 | -0.784 | -0.703 | -0.530 | -0.455 |
| Adj. R2 | | 0.165 | 0.126 | 0.060 | 0.093 | 0.033 | -0.007 | 0.060 |

### 6.4.1 Information Ratio

Information ratio is another metric, developed by Treynor & Black, to measure returns and risk. It is used in the portfolio management industry to compare portfolio managers and to assess their ability to generate returns above that of their benchmark.

$$Information\ Ratio = \frac{\overline{ER}}{\hat{\sigma}_{ER}}$$

Where $\overline{ER}$ is the arithmetic average of excess returns over the benchmark over the period, and $\hat{\sigma}_{\mathrm{ER}}$ is the standard deviation of the excess returns above the benchmark, for the same period, also called the tracking error (kilde Goodwin):

$$\hat{\sigma}_{ER} = \sqrt{\frac{\sum_{t=1}^{T}(ER_{\mathrm{t}} - \overline{ER})^2}{T-1}}$$

Where T is the length of the period.

The information ratio can be interpreted as the ratio of average excess returns over the benchmark per unit of risk in the excess returns. Compared to the Sharpe ratio which shows the excess returns over the risk-free option. Table 6.7 shows the information ratios and tracking errors for the portfolios with the top quartile excluded. Information ratios for the other quartiles are attached in the appendix, in table A3.9 and A3.10.

**Table 6.7:** Portfolio Information Ratios

| | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 |
| **Long-Only** | | | | | | | | |
| Information Ratio | -8.54 | 1.69 | 0.26 | 3.59 | -4.49 | -3.24 | 4.02 | 0.20 |
| Tracking Error | 0.03 | 0.05 | 0.05 | 0.03 | 0.04 | 0.04 | 0.06 | 0.03 |
| **Long-Short** | | | | | | | | |
| Information Ratio | | -4.32 | -3.40 | -2.47 | -7.85 | -6.92 | -2.14 | -3.92 |
| Tracking Error | | 0.06 | 0.06 | 0.07 | 0.08 | 0.07 | 0.08 | 0.07 |

The long-only portfolios all have positive ratios, except Log and NN1, while all the long-shorts have negative ratios. NN2 has the highest ratio, however, its portfolio only has 3.5 stocks included each month. So, its returns should be critically regarded. SVM, interestingly, has a significantly higher ratio than XGB and RF. Even though SVM does not generate that much more returns than XGB or RF. The high ratio is caused by its lower tracking error than XGB and RF.

### 6.4.2 Liquidity on Rebalance-Day

The alphas and abnormal returns generated by the machine learning portfolios should not be there. According to the efficient market hypothesis, any existing abnormal returns should be exploited by the market and arbitraged away very quickly. So if there in fact exists an alpha, the question arises: why hasn't it been arbitraged away?

We see that, when we exclude the smallest firms, this strategy likely does not work for big institutional investors. So, one answer could be that because those institutional investors, who have the resources to implement machine learning algorithms, do not have alpha. Hence, there is no incentive for them to do so. But why could not these investors set up smaller funds, trade these inefficiencies and exploit the returns until they were efficient again? Perhaps they are impossible, or at least very hard, to arbitrage away?

In the strategy implemented in this study, we assume that positions are sold at the end of each month and the new positions are taken immediately. Meaning we exit and enter the positions at the last day of every month. In table 6.8 we show how the gross returns from the portfolios, with the top quartile excluded, change after excluding the least liquid stocks.

**Table 6.8:** Returns' Liquidity
The table shows how the total gross portfolio returns change after excluding stocks with different levels of illiquidity on the rebalancing days.

| | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 |
| **Long-Only** | | | | | | | | |
| Gross Total Return w/o 25% | 53.05 | 93.77 | 85.74 | 93.86 | 60.01 | 70.96 | 95.47 | 83.24 |
| *Gross returns after excluding stocks with:* | | | | | | | | |
| Trading volume <1M | 47.01 | 46.71 | 41.49 | 70.59 | 53.7 | 55.06 | 63.2 | 48.13 |
| Trading volume <5M | 26.94 | 22.36 | 13.12 | 33.32 | 21.97 | 45.46 | 33.02 | 20.39 |
| Trading volume <10M | 16.41 | 8.22 | 4.46 | 26.36 | 9.87 | 36.63 | 2.95 | 18.36 |
| **Long-Short** | | | | | | | | |
| Gross Total Return w/o 25% | | 53.05 | 56.08 | 55.99 | 8.5 | 24.99 | 44.13 | 46.06 |
| *Gross returns after excluding stocks with:* | | | | | | | | |
| Trading volume <1M | | -4.56 | -7.88 | 28.03 | 7.65 | 13.85 | 19.31 | 7.74 |
| Trading volume <5M | | -11.87 | -18.01 | 9.26 | -9.20 | 27.36 | 6.52 | -4.31 |
| Trading volume <10M | | -14.03 | -15.68 | 13.18 | -10.45 | 30.74 | -14.06 | 5.29 |

The table shows how much of the gross total returns come from stocks with a volume larger than the constraint given on each row. For example, XGB's total return is 93.77%. Of those 93.77%, 46.71% comes from stocks with a daily trading volume of 1M NOK or more. Meaning that a little over half of the returns are contributions from stocks with a trading volume of less than 1M NOK. This makes the strategy significantly less feasible for investors. As the liquidity is so low in these stocks, the risk of moving the stock price can be considered very high. The ability to enter and exit the positions, at the given prices, is significantly lowered for investors with any reasonably sized portfolio. Especially if we continue to assume that we need to enter the positions in one day, as well as exit. The prices have a higher probability of moving and decreasing the returns. The risk only decreases with smaller investors. Only very small investors, if even they, could try to exploit this inefficiency. This could be an answer as to why it has not been arbitraged away. Only very small investors could have the ability to exploit this, or it could simply be impossible because of liquidity issues. There is also the fact that brokerage houses have a minimum fee for executing the transaction costs. Investors will pay the minimum fee if the brokerage fee times the size of the trade is below the minimum fee. This is more likely to be the effective transaction cost for small, or micro-investors, as the size of their trades would trigger the minimum fee. Giving the investors a larger transaction fee in percentage terms. It could be that these returns are exploited and arbitraged away to the point where the minimum fee is too large and removes the possibility for any excess returns. Suggesting that the returns are observable, but unattainable. Like rainbow-returns.

# 7 Conclusion

In this thesis we have implemented seven different machine learning models, with three objectives in mind. First, to assess the models' ability to predict whether a stock's next month return is positive or negative. We do this with the use of 83 firm characteristics, constructed from Norwegian stock market trading data, as well as company accounting data. Secondly, to investigate which of these characteristics are considered the most important by the models, in making the next month's predictions. Effectively, investigating what matters to Norwegian investors. Lastly, to evaluate the monthly portfolio performances of the model predictions.

We find that, of the seven machine learning models, an extreme-gradient boosted decision tree algorithm performs best, with a monthly prediction accuracy of 53.16% and Area-Under-Curve of 53.49%.

Of the 83 firm characteristics, we find that those related to risk measures are the most important when predicting the Norwegian stock market. The second most important group of characteristics involve price trends/momentum, while the third most important is fundamentals and valuation ratios.

To assess portfolio performance, we constructed long-only and long-short portfolios for each model. In terms of monthly average returns, before considering any market frictions, 6 out of the 7 models outperform the OSEBX by between 0.06% and 0.47%. However, after including market frictions that institutional investors might face, we find that all the models significantly underperform. When considering possible portfolios for retail investors, we first find excess returns and alphas. However, upon investigating the possible explanations for the abnormal returns, we find that these returns are mainly driven by highly illiquid stocks. Meaning, the returns are only available to be exploited by very small investors, or in fact unattainable all together. We argue that the abnormal returns are impossible to arbitrage away because of the illiquidity in their associated stocks. We call this phenomenon “Rainbow returns”, given that these are probably only observable, unattainable returns. Our findings support the efficient market hypothesis, in that our models are unable to "beat the market" using public information, after accounting for market frictions and constraints associated with investing.

## 7.1 Further research

As our models are based on monthly data, we open for the possibility of introducing significant amounts of randomness. For further research, it would be interesting seeing how the models would perform on daily data points, with firm characteristics based solely on stock trading data, in addition to including popular technical analysis indicators as variables. Using only trading data would allow for the inclusion of significantly more stocks in the dataset, as stock trading data is notably more available and have higher frequency, than full accounting reports and revenue statements. In our study, the accounting data is incomplete and forces us to remove quite many stocks. Furthermore, by having a larger sample of stocks, it would open for being more meticulous in which stocks to include in the construction of the portfolios. Alternative ways to pick the stocks to include, for example only including stocks with a higher probability than 60%, rather than 50%. It would also be interesting to see how a stop loss strategy would affect the monthly returns.

# References

Abarbanell, J. S. and Bushee, B. J. (1998). Abnormal returns to a fundamental analysis strategy. *Accounting Review*, pages 19–45.

Abdi, H. (2003). Partial least square regression (pls regression). *Encyclopedia for research methods for the social sciences*, 6(4):792–795.

Ackermann, C., McEnally, R., and Ravenscraft, D. (1999). The performance of hedge funds: Risk, return, and incentives. *The Journal of Finance*, 54(3):833–874.

Ali, A., Hwang, L.-S., and Trombley, M. A. (2003). Arbitrage risk and the book-to-market anomaly. *Journal of Financial Economics*, 69(2):355–373.

Almeida, H. and Campello, M. (2007). Financial constraints, asset tangibility, and corporate investment. *The Review of Financial Studies*, 20(5):1429–1460.

Amihud, Y. (2002). Illiquidity and stock returns: cross-section and time-series effects. *Journal of financial markets*, 5(1):31–56.

Anderson, C. W. and Garcia-Feijoo, L. (2006). Empirical evidence on capital investment, growth options, and security returns. *The Journal of Finance*, 61(1):171–194.

Ang, A., Hodrick, R. J., Xing, Y., and Zhang, X. (2006). The cross-section of volatility and expected returns. *The journal of finance*, 61(1):259–299.

Asness, C. S., Porter, R. B., and Stevens, R. L. (2000). Predicting stock returns using industry-relative firm characteristics. *Available at SSRN 213872.*

Balakrishnan, K., Bartov, E., and Faurel, L. (2010). Post loss/profit announcement drift. *Journal of Accounting and Economics*, 50(1):20–41.

Balatbat, M. (2006). Discussion of explaining the short and long-term ipo anomalies in the us by r&d. *Journal of Business Finance & Accounting*, 33(3-4):580–586.

Bali, T. G., Cakici, N., and Whitelaw, R. F. (2011). Maxing out: Stocks as lotteries and the cross-section of expected returns. *Journal of financial economics*, 99(2):427–446.

Bandyopadhyay, S. P., Huang, A. G., and Wirjanto, T. S. (2010). The accrual volatility anomaly. *Unpublished Manuscript, University of Waterloo.*

Banz, R. W. (1981). The relationship between return and market value of common stocks. *Journal of financial economics*, 9(1):3–18.

Barbee Jr, W. C., Mukherji, S., and Raines, G. A. (1996). Do sales–price and debt–equity explain stock returns better than book–market and firm size? *Financial Analysts Journal*, 52(2):56–60.

Barth, M. E., Elliott, J. A., and Finn, M. W. (1999). Market rewards associated with patterns of increasing earnings. *Journal of Accounting Research*, 37(2):387–413.

Bhandari, L. C. (1988). Debt/equity ratio and expected common stock returns: Empirical evidence. *The journal of finance*, 43(2):507–528.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Brown, D. P. and Rowe, B. (2007). The productivity premium in equity returns. *Available at SSRN 993467.*

Carhart, M. M. (1997). On persistence in mutual fund performance. *The Journal of finance*, 52(1):57–82.

Chandrashekar, S. and Rao, R. K. (2009). The productivity of corporate cash holdings and the cross-section of expected stock returns. *McCombs Research Paper Series No. FIN-03-09.*

Chen, L. and Zhang, L. (2010). A better three-factor model that explains more anomalies. *Journal of Finance*, 65(2):563–595.

Chordia, T., Subrahmanyam, A., and Anshuman, V. R. (2001). Trading activity and expected stock returns. *Journal of financial Economics*, 59(1):3–32.

Cooper, M. J., Gulen, H., and Schill, M. J. (2008). Asset growth and the cross-section of stock returns. *the Journal of Finance*, 63(4):1609–1651.

Datar, V. T., Naik, N. Y., and Radcliffe, R. (1998). Liquidity and stock returns: An alternative test. *Journal of financial markets*, 1(2):203–219.

Desai, H., Rajgopal, S., and Venkatachalam, M. (2004). Value-glamour and accruals mispricing: One anomaly or two? *The Accounting Review*, 79(2):355–385.

DNB (2022). Dnb prisliste. https://www.dnb.no/markets/aksjer/prisliste-aksjehandel. Accessed: 2022-31-05.

Eberhart, A. C., Maxwell, W. F., and Siddique, A. R. (2004). An examination of long-term abnormal stock returns and operating performance following r&d increases. *The Journal of Finance*, 59(2):623–650.

Evgeniou, T. and Pontil, M. (1999). Support vector machines: Theory and applications. In *Advanced Course on Artificial Intelligence*, pages 249–257. Springer.

Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The journal of Finance*, 25(2):383–417.

Fama, E. F. and French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of financial economics*, 33(1):3–56.

Fama, E. F. and French, K. R. (2015). A five-factor asset pricing model. *Journal of financial economics*, 116(1):1–22.

Fama, E. F. and MacBeth, J. D. (1973). Risk, return, and equilibrium: Empirical tests. *Journal of political economy*, 81(3):607–636.

Francis, J., LaFond, R., Olsson, P. M., and Schipper, K. (2004). Costs of equity and earnings attributes. *The accounting review*, 79(4):967–1010.

Gettleman, E. and Marks, J. M. (2006). Acceleration strategies. *SSRN Electronic Journal.*

Green, J., Hand, J. R., and Zhang, X. F. (2017). The characteristics that provide independent information about average us monthly stock returns. *The Review of Financial Studies*, 30(12):4389–4436.

Greenacre, M. J. (1984). Theory and applications of correspondence analysis.

Grossman, S. J. and Stiglitz, J. E. (1980). On the impossibility of informationally efficient markets. *The American Economic Review*, 70(3):393–408.

Gu, S., Kelly, B., and Xiu, D. (2020). Empirical Asset Pricing via Machine Learning. *The Review of Financial Studies*, 33(5):2223–2273.

Hafzalla, N., Lundholm, R., and Matthew Van Winkle, E. (2011). Percent accruals. *The Accounting Review*, 86(1):209–236.

Hand, D. and Till, R. (2001). A simple generalisation of the area under the roc curve for multiple class classification problems. *Hand, The*, 45:171–186.

Hofmann, T., Schölkopf, B., and Smola, A. J. (2008). Kernel methods in machine learning. *The annals of statistics*, 36(3):1171–1220.

Holthausen, R. W. and Larcker, D. F. (1992). The prediction of stock returns using financial statement information. *Journal of accounting and economics*, 15(2-3):373–411.

Hou, K. and Robinson, D. T. (2006). Industry concentration and average stock returns. *The Journal of Finance*, 61(4):1927–1956.

Hou, K., Xue, C., and Zhang, L. (2015). Digesting anomalies: An investment approach. *The Review of Financial Studies*, 28(3):650–705.

Huang, A. G. (2009). The cross section of cashflow volatility and expected stock returns. *Journal of Empirical Finance*, 16(3):409–429.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*, volume 112. Springer.

Jegadeesh, N. (1990). Evidence of predictable behavior of security returns. *The Journal of finance*, 45(3):881–898.

Jegadeesh, N. and Titman, S. (1993). Returns to buying winners and selling losers: Implications for stock market efficiency. *The Journal of finance*, 48(1):65–91.

Jensen, M. C. (1968). The performance of mutual funds in the period 1945-1964. *The Journal of finance*, 23(2):389–416.

Kama, I. (2009). On the market reaction to revenue and earnings surprises. *Journal of Business Finance & Accounting*, 36(1-2):31–50.

Lakonishok, J., Shleifer, A., and Vishny, R. W. (1994). Contrarian investment, extrapolation, and risk. *The journal of finance*, 49(5):1541–1578.

Leippold, M., Wang, Q., and Zhou, W. (2021). Machine learning in the chinese stock market. *Journal of Financial Economics*.

Lev, B. and Nissim, D. (2004). Taxable income, future earnings, and equity values. *The accounting review*, 79(4):1039–1074.

Litzenberger, R. H. and Ramaswamy, K. (1982). The effects of dividends on common stock prices tax effects or information effects? *The Journal of Finance*, 37(2):429–443.

Liu, W. (2006). A liquidity-augmented capital asset pricing model. *Journal of financial Economics*, 82(3):631–671.

Michaely, R., Thaler, R. H., and Womack, K. L. (1995). Price reactions to dividend initiations and omissions: Overreaction or drift? *the Journal of Finance*, 50(2):573–608.

Mohanram, P. S. (2005). Separating winners from losers among lowbook-to-market stocks using financial statement analysis. *Review of accounting studies*, 10(2):133–170.

Monroe, W. (2017). Logistic regression. *Recall*, 1(1).

Nordnet (2022). Nordnet prisliste. https://www.nordnet.no/no/kundeservice/prisliste. Accessed: 2022-31-05.

Novy-Marx, R. (2013). The other side of value: The gross profitability premium. *Journal of financial economics*, 108(1):1–28.

Ou, J. A. and Penman, S. H. (1989). Financial statement analysis and the prediction of stock returns. *Journal of accounting and economics*, 11(4):295–329.

Palazzo, B. (2012). Cash holdings, risk, and expected returns. *Journal of Financial Economics*, 104(1):162–185.

Pearson, E. S. (1931). The test of significance for the correlation coefficient. *Journal of the American Statistical Association*, 26(174):128–134.

Piotroski, J. D. (2000). Value investing: The use of historical financial statement information to separate winners from losers. *Journal of Accounting Research*, pages 1–41.

Pontiff, J. and Woodgate, A. (2008). Share issuance and cross-sectional returns. *The Journal of Finance*, 63(2):921–945.

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Richardson, S. A., Sloan, R. G., Soliman, M. T., and Tuna, I. (2005). Accrual reliability, earnings persistence and stock prices. *Journal of accounting and economics*, 39(3):437–485.

Rosenberg, B., Reid, K., and Lanstein, R. (1985). Persuasive evidence of market inefficiency. *The Journal of Portfolio Management*, 11(3):9–16.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61:85–117.

Sloan, R. G. (1996). Do stock prices fully reflect information in accruals and cash flows about future earnings? *Accounting review*, pages 289–315.

Soliman, M. T. (2008). The use of dupont analysis by market participants. *The accounting review*, 83(3):823–853.

The International Energy Agency (2018). Global ev outlook 2018, towards cross-modal electrification.

Thomas, J. and Zhang, F. X. (2011). Tax expense momentum. *Journal of Accounting Research*, 49(3):791–821.

Thomas, J. K. and Zhang, H. (2002). Inventory changes and future returns. *Review of Accounting Studies*, 7(2):163–187.

Titman, S., Wei, K. J., and Xie, F. (2004). Capital investments and stock returns. *Journal of financial and Quantitative Analysis*, 39(4):677–700.

Tuzel, S. (2010). Corporate real estate holdings and the cross-section of stock returns. *The Review of Financial Studies*, 23(6):2268–2302.

# Appendix

## A1 Variable Construction

The variables created are similar to those used in earlier studies on the US and Chinese stock market, closely following the definitions in Green et al. (2017).

1. *positive_return*: is an indicator variable for positive return, during the last month, taking the value of 1 if positive return and 0 if negative return. Data is acquired from børsprosjektet at NHH

2. *absacc*: is the absolute value of *acc* (see variable 3, acc, for further explanation). Data is acquired from SNF.

3. *acc*: Is the working capital accrues, constructed using the definition of Sloan (1996). Data is acquired from SNF.

$$Accruals = (\Delta CA - \Delta Cash) - (\Delta CL - \Delta STD - \Delta TP) - Dep$$

   Where: $\Delta CA$ = Change in current assets

   $\Delta Cash$ = Change in cash/cash equivalents

   $\Delta CL$ = Change in current liabilities

   $\Delta STD$ = Change in debt included in current liabilities

   $\Delta TP$ = Change in income taxes payable

   $Dep$ = depreciation and amortization expense

4. *agr*: is the asset growth, calculated as the yearly percentage change in total assets. Data is acquired from SNF.

5. *beta*: Calculated using daily stock returns and daily OSEBX returns as the market return. The regression is done using a 2 year rolling window and later converted to monthly by taking the last beta of that month. The beta of a given month is then the 1 month lagged beta, meaning we get the last observed beta in the previous month. Stock returns and OSEBX returns are acquired from børsprosjektet at NHH.

6. *betasq*: is the beta squared (see variable 5, beta, for further explanation).

7. *bm*: is the book-to-market ratio, which is the book value of equity divided by the market capitalization. Book value of equity is acquired from SNF while market capitalization is calculated from shares issued and the daily closing price using data from børsprosjektet at NHH.

8. *bm_ia*: is the book-to-market ratio adjusted for the industry introduced by Asness et al. (2000).

$$bm\_ia_{it} = bm_{it} - bm\_I_{it}$$

Where: $bm_{it}$ = the firms, i, book-to-market ratio at time t.
$bm\_I_{it}$ = the average of the equally weighted book-to-market ratio in the firms, i, industry. The firms industries are taken from the last reported "Sector" in the SNF data.

9. *cash_1*: is the firms cash and cash equivalents divided by the average of the last two years total assets. Data used is acquired from SNF.

10. *cashdebt*: is cash flow to debt ratio, which is calculated from dividing earning on total liabilities. Data is acquired from SNF.

11. *cashspr*: is the cash productivity, which is calculated from the market capitalization plus the long term debt minus total assets divided by cash and cash equivalents. Data is acquired from Børsprosjektet at NHH and SNF.

12. *cfp*: is the operating cash flows divided by fiscal-year-end market capitalization. Data is acquired from børsprosjektet at NHH and SNF.

13. *cfp_ia*: is industry adjusted cfp. The firms industries are taken form the last reported "Sector" in the SNF data.

14. *chato*: yearly average sales divided by the yearly average total assets. Data is acquired from SNF.

15. *chato_ia*: is chato with sales adjusted for industry. Data is acquired from SNF. The firms industries are taken form the last reported "Sector" in the SNF data.

16. *chinv*: is the change in inventory scaled by the total assets. Data is acquired from SNF.

17. *chmom*: is the 6 months momentum (see variable 43, mom6m, for further explanation) from t-6 to t-1 minus the 6 months momentum from t-12 to t-7. Data is acquired from Børsprosjektet at NHH.

18. *chpm*: the yearly change in income before extraordinary items. Data is acquired from SNF.

19. *chpm_ia*: is chpm with industry adjusted income. The firms industries are taken from the last reported "Sector" in the SNF data.

20. *chsho*: the yearly percentage change in shares issued. Data is acquired from børsprosjektet at NHH.

21. *chtx*: the yearly percentage change in total taxes. data is acquired from SNF.

22. *cinvest*: yearly change in fixed assets divided by sales, then taking it and subtracting the average over the last three years. Data is acquired from SNF.

23. *currat*: is current assets divided by current liabilities. Data is acquired from SNF.

24. *depr*: depreciation divided by fixed assets. Data is acquired from SNF.

25. *divi*: a dummy variable that takes the value of 1 if the company pays dividend this year but did not last year. Data is acquired from SNF.

26. *divo*: a dummy variable that takes the value of 1 if the company does not pay dividend this year but did the last year. Data is acquired from SNF.

27. *dolvol*: price per share multiplied with the natural logarithm of the traded volume, both from t-2. Data is acquired from børsprosjektet at NHH.

28. *dy*: yearly calculation of the dividends payed divided by the market capitalization. Data is acquired from børsprosjektet at NHH.

29. *egr*: yearly change in book value of equity from dividing this years equity on the prior year. Data is acquired from SNF.

30. *gma*: is the revenue minus cost of goods sold divided by lagged total assets. Data is

acquired from SNF.

31. $grCAPX$: The percentage change in capital expenditures from year t-2 to t. data is acquired from SNF.

32. $herf$: sum of squared percent of sales in industry for each company. The firms industries are taken from the last reported "Sector" in the SNF data.

33. *idiovol*: standard deviation of residuals of daily returns on daily market returns for three years prior to month end. Data is acquired from børsprosjektet at NHH.

34. *ill*: is the absolute return divided by the trading volume. data is acquired from børsprosjektet at NHH.

35. *invest*: the yearly change in fixed assets plus the yearly change in inventories divided by lagged total assets. Data is acquired from SNF.

36. *lev*: total liabilities divided by the market capitalization. Data is acquired from børsprosjektet at NHH and SNF.

37. *lgr*: the yearly percentage change in total liabilities. Data is acquired from SNF.

38. *ma_price_short*: is the 10 day moving average, calculated as the average price in the last 10 days. Data is acquired from børsprosjektet at NHH.

39. *ma_price_int*: is the 50 day moving average, calculated as the average price in the last 50 days. Data is acquired from børsprosjektet at NHH.

40. *ma_price_long*: is the 200 day moving average, calculated as the average price in the last 200 days. Data is acquired from børsprosjektet at NHH.

41. *maxret*: the highest daily return observed in the last month. data is acquired from børsprosjektet at NHH.

42. $mom1m$: is the 1 month cumulative returns. Data is acquired from børsprosjektet at NHH.

43. $mom6m$: is the 5 months cumulative returns, ending 1 month before. Data is acquired from børsprosjektet at NHH.

44. $mom12m$: is the 11 months cumulative returns, ending 1 month before. Data is

acquired from børsprosjektet at NHH.

45. *mom36m*: is the cumulative returns from t-36 to t-13. Data is acquired from børsprosjektet at NHH.

46. *ms*: using the Mohanram G-score, which is a financial indicator involving 8 variables and taking the sum of those. In our ms one variable is not present, advertising intensity, as it is not present in the data acquired. Data is acquired from SNF. For further reading Mohanram (2005).

47. *mve*: natural logarithm of the market capitalization at the end of month t-1. Data is acquired from børsprosjektet at NHH.

48. *mve_ia*: narutal logarithm of the market capitalization at the end of month t-1 adjusted for industry. The firms industries are taken form the last reported "Sector" in the SNF data.

49. *nincr*: is the number of consecutive years with an increase in earnings. Data is acquired from SNF.

50. *operprof*: is the operating profit divided by the lagged common shareholders equity. Data is acquired from SNF.

51. *pchcapx_ia*: is the yearly percentage change in capital expenditures adjusted for industry. The firms industries are taken form the last reported "Sector" in the SNF data.

52. *pchcurrat*: is the yearly percentage change in currat (see variable 23, currat, for further explanation). Data is acquired from SNF.

53. *pchdepr*: is the yearly percentage change in depr (see variable 24, depr, for further explanation). Data is acquired from SNF.

54. *pchgm_pchsale*: is the yearly percentage change in gross margin minus yearly percentage change in sales. Data is acquired from SNF.

55. *pchquick*: is the yearly percentage change in quick (see variable 61, quick, for further explanation). Data is acquired from SNF.

56. *pchsale_pchinvt*: Yearly percentage change in sales minus yearly percentage change

in inventory. Data is acquired from SNF.

57. *pchsale_pchrect*: Yearly percentage change in sales minus yearly percentage change in receivables. Data is acquired from SNF.

58. *pchsaleinv*: yearly percentage change in saleinv (see variable 72, saleinv, for further explanation). Data is acquired from SNF.

59. *pctacc*: acc (see variable 3, acc, for further explanation) divided on absolute value of net income. Net income is set to 0.01 in cases where it would be zero. Data is acquired from SNF.

60. *ps*: is a score to determine the strength of a firms financial position. The final score involves 4 profitability criteria, 3 leverage, liquidity and source of funds criteria and 2 on operating efficiency. If the criteria is met, a value of 1 will be added to the score. The maximum score a firm can achieve, if all criteria is met, is therefore 9. for further explanation see Piotroski (2000). Data is acquired from SNF.

61. *quick*: creating by dividing the sum of current assets minus inventory on current liabilities. Data is acquired from SNF.

62. *rd*: a variable taking the value of 1 if research and development expense as a percentage of total assets increased with 5% or more from last year. If it doesn't increase with 5% or more it takes the value of 0. Data is acquired from børsprosjektet at NHH and SNF.

63. *rd_mve*: Research and development expense divided by market capitalization. Data is acquired from SNF.

64. *rd_sale*: Research and development expense divided by sales. Data is acquired from SNF.

65. *realestate*: assets divided by fixed assets. Data is acquired from SNF.

66. *roaq*: income before extraordinary items divided by lagged total assets. Data is acquired from SNF.

67. *roavol*: the standard deviation of the last 2 years income before extraordinary items divided the average assets over that period. Data is acquired from SNF.

68. *roeq*: Income before extraordinary items divided by lagged common shareholders equity. Data is acquired from SNF.

69. *roie*: yearly earnings before interest and taxes minus non operating income divided by non-cash enterprise value. Data is acquired from SNF.

70. *rsup*: sales from year t-1 minus sales from year t-2 divided by market capitalization. Data is acquired from børsprosjektet at NHH and SNF.

71. *salecash*: Yearly sales divided by cash and cash equivalents. Data is acquired from SNF.

72. *saleinv*: yearly sales divided by total inventory. Data is acquired from SNF.

73. *salerev*: Yearly sales divided by account receivables. Data is acquired from SNF.

74. *sgr*: The yearly percentage change in sales. Data is acquired from SNF.

75. *sp*: yearly change divided by market capitalization. Data is acquired from børsprosjektet at NHH and SNF.

76. *std_nokvol*: Monthly standard deviation of NOK trading volume. Data is acquired from børsprosjektet at NHH.

77. *str_turn*: Monthly standard deviation of monthly share turnover. Data is acquired from børsprosjektet at NHH.

78. *stdacc*: the standard deviation of the last 3 years acc (see variable 1, acc, for further explanation). Data is acquired from SNF.

79. *stdcf*: the standard deviation of cash flows over the last 3 years. Data is acquired from SNF.

80. *tang*: the formula for tang is :

    Cash holdings + 0.715 * recievables + 0.547 * inventory + 0.535 * fixed assets / total assets

    For further explanation see Almeida and Campello (2007). Data is acquired from SNF.

81. *tb*: is the tax income, calculated from current tax expense divided by the maximum federal tax rate, which in Norway is 22%, divided by income before extraordinary

income. Data is acquired from SNF.

82. *turn*: is the average monthly trading volume over the last 3 months divided by number of shares outstanding in the current month. Data is acquired from børsprosjektet at NHH.

83. *volatility*: standard deviation of daily returns from the last month. Data is acquired from børsprosjektet at NHH.

84. *zerotrade*: Turnover weighted number of zero trading days in the last month. Data is acquired from børsprosjektet at NHH.

**Table A1.1:** Variable Authors

| No. | Variable name | Description | Author(s) | Year | Data Source | Frequency |
|---|---|---|---|---|---|---|
| 1 | positive_return | Indicator for positive or negative return | | | Børsprosjektet at NHH | Monthly |
| 2 | absacc | Absolute accruals | Bandyopadhyay, Huang & Wirjant | 2010 | SNF | Yearly |
| 3 | acc | Working capital accruals | Sloan | 1996 | SNF | Yearly |
| 4 | agr | Asset growth | Cooper, Gulen & Schill | 2008 | SNF | Yearly |
| 5 | beta | Beta | Fama & MacBeth | 1973 | Børsprosjektet at NHH | Monthly |
| 6 | betasq | Beta squared | Fama & MacBeth | 1973 | Børsprosjektet at NHH | Monthly |
| 7 | bm | Book-to-market | Rosenberg, Reid & Lanstein | 1985 | Børsprosjektet at NHH, SNF | Monthly |
| 8 | bm_ia | Industry-adjusted book to market | Asness, Porter & Stevens | 2000 | Børsprosjektet at NHH, SNF | Monthly |
| 9 | cash | Cash holdings | Palazzo | 2012 | SNF | Yearly |
| 10 | cashdebt | Cash flow to debt | Ou & Penman | 1989 | SNF | Yearly |
| 11 | cashspr | Cash productivity | Chandrashekar & Rao | 2009 | Børsprosjektet at NHH, SNF | Monthly |
| 12 | cfp | Cash flow to price ratio | Desai, Rajgopal & Venkatachalam | 2004 | Børsprosjektet at NHH, SNF | Monthly |
| 13 | cfp_ia | Industry-adjusted cash flow to price ratio | Asness, Porter & Stevens | 2000 | Børsprosjektet at NHH, SNF | Monthly |
| 14 | chato | Change in asset turnover | Soliman | 2008 | SNF | Yearly |
| 15 | chaotia | Industry-adjusted change in asset turnover | Soliman | 2008 | SNF | Yearly |
| 16 | chinv | Change in inventory | Thomas & Zhang | 2002 | SNF | Yearly |
| 17 | chmom | Change in 6-month momentum | Gettleman & Marks | 2006 | Børsprosjektet at NHH | Monthly |
| 18 | chpm | Change in profit margin | Soliman | 2008 | SNF | Yearly |
| 19 | chpmia | Industry-adjusted change in profit margin | Soliman | 2008 | SNF | Yearly |
| 20 | chsho | Change in shares outstanding | Pontiff & Woodgate | 2008 | Børsprosjektet at NHH | Monthly |
| 21 | chtx | Change in tax expense | Thomas & Zhang | 2011 | SNF | Yearly |
| 22 | cinvest | Corporate investment | Titman, Wei & Xie | 2004 | SNF | Yearly |
| 23 | currat | Current ratio | Ou & Penman | 1989 | SNF | Yearly |
| 24 | depr | Depreciation / PP&E | Holthausen & Larcker | 1992 | SNF | Yearly |
| 25 | divi | Dividend initiation | Michaely, Thaler & Womack | 1995 | SNF | Yearly |
| 26 | divo | Dividend omission | Michaely, Thaler & Womack | 1995 | SNF | Yearly |
| 27 | dolvol | NOK trading volume | Chordia, Subrahmanyam & Anshuman | 2001 | Børsprosjektet at NHH | Monthly |
| 28 | dy | Dividend to price | Litzenberger & Ramaswamy | 1982 | Børsprosjektet at NHH | Monthly |
| 29 | egr | Growth in common shareholder equity | Richardson, Sloan, Soliman & Tuna | 2005 | SNF | Yearly |
| 30 | gma | Gross profitability | Novy-Marx | 2013 | SNF | Yearly |

| No. | Variable name | Description | Author(s) | Year | Data Source | Frequency |
|---|---|---|---|---|---|---|
| 31 | grCAPX | Growth in capital expenditures | Anderson & Garcia-Feijoo | 2006 | SNF | Yearly |
| 32 | herf | Industry sales concentration | Hou & Robinson | 2006 | SNF | Yearly |
| 33 | idiovol | Idiosyncratic return volatility | Ali, Hwang & Trombley | 2003 | Børsprosjektet at NHH | Monthly |
| 34 | ill | Illiquidity | Amihud | 2002 | Børsprosjektet at NHH | Monthly |
| 35 | invest | Capital expenditures and inventory | Chen & Zhang | 2010 | SNF | Yearly |
| 36 | lev | Leverage | Bhandari | 1988 | Børsprosjektet at NHH, SNF | Yearly |
| 37 | lgr | Growth in long-term debt | Richardson, Sloan, Soliman & Tuna | 2005 | SNF | Yearly |
| 38 | ma_price_int | Moving average of 50 days | | | Børsprosjektet at NHH | Monthly |
| 39 | ma_price_long | Moving average of 200 days | | | Børsprosjektet at NHH | Monthly |
| 40 | ma_price_short | Moving average of 10 days | | | Børsprosjektet at NHH | Monthly |
| 41 | maxret | Maximum daily return in the given month | Bali, Cakici & Whitelaw | 2011 | Børsprosjektet at NHH | Monthly |
| 42 | mom12m | 12-month momentum | Jegadeesh | 1990 | Børsprosjektet at NHH | Monthly |
| 43 | mom1m | 1-month momentum | Jegadeesh & Titman | 1993 | Børsprosjektet at NHH | Monthly |
| 44 | mom36m | 36-month momentum | Jegadeesh & Titman | 1993 | Børsprosjektet at NHH | Monthly |
| 45 | mom6m | 6-month momentum | Jegadeesh & Titman | 1993 | Børsprosjektet at NHH | Monthly |
| 46 | ms | Financial statement score | Mohanram | 2005 | SNF | Yearly |
| 47 | mve | Size | Banz | 1981 | Børsprosjektet at NHH | Monthly |
| 48 | mve_ia | Industry-adjusted size | Asness, Porter & Stevens | 2000 | Børsprosjektet at NHH, SNF | Monthly |
| 49 | nincr | Number of earnings increases | Barth, Elliott & Finn | 1999 | SNF | Yearly |
| 50 | operprof | Operating profitability | Fama & French | 2015 | SNF | Yearly |
| 51 | pchcapx_ia | % Industry adjusted % change in capital expenditures | Ou & Penman | 1989 | SNF | Yearly |
| 52 | pchcurrat | % change in current ratio | Ou & Penman | 1989 | SNF | Yearly |
| 53 | pchdepr | % change in depreciation | Holthausen & Larcker | 1992 | SNF | Yearly |
| 54 | pchgm_pchsale | % change in gross margin - % change in sales | Abarbanell & Bushee | 1998 | SNF | Yearly |
| 55 | pchquick | % change in quick ratio | Ou & Penman | 1989 | SNF | Yearly |
| 56 | pchsale_pchinvt | % change in sales - % change in inventory | Abarbanell & Bushee | 1998 | SNF | Yearly |
| 57 | pchsale_pchrect | % change in sales - % change in A/R | Abarbanell & Bushee | 1998 | SNF | Yearly |
| 58 | pchsaleinv | % change sales-to-inventory | Ou & Penman | 1989 | SNF | Yearly |
| 59 | pctacc | Percent accruals | Hafzalla, Lundholm & Van Winkle | 2011 | SNF | Yearly |
| 60 | ps | Financial statements score | Piotroski | 2000 | SNF | Yearly |

| No. | Variable name | Description | Author(s) | Year | Data Source | Frequency |
|---|---|---|---|---|---|---|
| 61 | quick | Quick ratio | Ou & Penman | 1989 | SNF | Yearly |
| 62 | rd | R&D increase | Eberhart, Maxwell & Siddique | 2004 | SNF | Yearly |
| 63 | rd_mve | R&D to market capitalization | Guo, Lev & Shi | 2006 | Børsprosjektet at NHH, SNF | Yearly |
| 64 | rd_sale | R&D to sales | Guo, Lev & Shi | 2006 | SNF | Yearly |
| 65 | realestate | Real estate holdings | Tuzel | 2010 | SNF | Yearly |
| 66 | roaq | Return on assets | Balakrishnan, Bartov & Faurel | 2010 | SNF | Yearly |
| 67 | roavol | Earnings volatility | Francis, LaFond, Olsson & Schipper | 2004 | SNF | Yearly |
| 68 | roeq | Return on equity | Hou, Xue & Zhang | 2015 | SNF | Yearly |
| 69 | roic | Return on invested capital | Brown & Rowe | 2007 | SNF | Yearly |
| 70 | rsup | Revenue surprise | Kama | 2009 | Børsprosjektet at NHH, SNF | Yearly |
| 71 | salecash | Sales to cash | Ou & Penman | 1989 | SNF | Yearly |
| 72 | saleinv | Sales to inventory | Ou & Penman | 1989 | SNF | Yearly |
| 73 | salerev | Sales to receivables | Ou & Penman | 1989 | SNF | Yearly |
| 74 | sgr | Sales growth | Lakonishok, Shleifer & Vishny | 1994 | SNF | Yearly |
| 75 | sp | Sales to price | Barbee, Mukherji, & Raines | 1996 | Børsprosjektet at NHH, SNF | Yearly |
| 76 | std_nokvol | Volatility of liquidity (NOK turnover) | Chordia, Subrahmanyam & Anshuman | 2001 | Børsprosjektet at NHH | Monthly |
| 77 | std_turn | Volatility of liquidity (share turnover) | Chordia, Subrahmanyam & Anshuman | 2001 | Børsprosjektet at NHH | Monthly |
| 78 | stdacc | Accrual volatility | Bandyopadhyay, Huang & Wirjanto | 2010 | SNF | Yearly |
| 79 | stdcf | Cash flow volatility | Huang | 2009 | SNF | Yearly |
| 80 | tang | Debt capacity/firm tangibility | Almeida & Campello | 2007 | SNF | Yearly |
| 81 | tb | Tax income to book income | Lev & Nissim | 2004 | SNF | Yearly |
| 82 | turn | Share turnover | Datar, Naik & Radcliffe | 1998 | Børsprosjektet at NHH | Monthly |
| 83 | volatility | Return volatility | Ang, Hodrick, Xing & Zhang | 2006 | Børsprosjektet at NHH | Monthly |
| 84 | zerotrade | Number of zero-trading-days | Liu | 2006 | Børsprosjektet at NHH | Monthly |

# A2 Hyperparameters

Table A2.1 is a summary of all hyperparameters for all models and the corresponding specifications used.

**Table A2.1:** Summary of hyperparameters
Shows the hyperparameters used in all models and their ranges.

| | XGBoost | RF | NN1-NN3 |
|---|---|---|---|
| Specification | # mtry<br>M = 1 ~84<br># min_n<br>N = 2<br>#Depth<br>L = 1 ~15<br># Trees<br>B = 5000<br>Learning rate<br>LR \in {10^-10, 0.082}<br>Loss reduction<br>LossR \in {10^-10, 30.1} | # mtry<br>M = 2 ~76<br># min_n<br>N = 4 ~38<br># Trees<br>B = 1000 | Learning Rate<br>LR = 10^-2<br>Batch size<br>B \in {256, 512, 768}<br>epochs = 100<br>patience = 5<br>Adam Para. = Default |

# A3 Portfolio Returns Descriptive Analysis

**Table A3.1:** Gross long-only returns within firm sizes
Table A3.1 shows the returns and statistics the machine learning long-only portfolios achieve, after removing specific quantiles of firms based on market capitalization. All the firms were sorted into four quartiles based on their market capitalization. "All firms" show the numbers for the entire test-sample. "Top 75%" show the numbers for the numbers after removing smallest quartile. "Bottom 75%" show the numbers after removing the largest quartile. "Bottom 50%" show the numbers by only including the smallest and second smallest quartiles. "Bottom 25%" shows the number for just the smallest quartile. "Total return" is the cumulative return at the end of period. "Avg. monthly return" is the average monthly return. "Std. Dev." is the standard deviation of the monthly returns. "Sharpe Ratio" is the Sharpe ratio. "Avg. N" is the average number of stocks in the portfolio each month.

| | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 |
| **Long-Only** | | | | | | | | |
| ***All firms:*** | | | | | | | | |
| Total return | 47.43 | 77.47 | 65.86 | 59.00 | 35.99 | 53.18 | 62.15 | 62.51 |
| Avg. monthly return | 0.80 | 1.31 | 1.12 | 1.00 | 0.61 | 0.90 | 1.05 | 1.06 |
| Std. Dev. | 5.27 | 6.53 | 6.06 | 4.34 | 4.43 | 5.11 | 6.32 | 4.85 |
| Sharpe Ratio | 0.42 | 0.61 | 0.54 | 0.66 | 0.34 | 0.5 | 0.48 | 0.64 |
| Avg. N | 66.79 | 29.05 | 27.51 | 24.59 | 24.54 | 21.1 | 5.78 | 14.27 |
| ***Top 75%:*** | | | | | | | | |
| Total return | 28.19 | 24.54 | 28.32 | 41.54 | 32.9 | 59.09 | 44.01 | 58.92 |
| Avg. monthly return | 0.48 | 0.42 | 0.48 | 0.70 | 0.56 | 1.00 | 0.75 | 1.00 |
| Std. Dev. | 4.49 | 5.49 | 4.86 | 4.17 | 4.33 | 4.47 | 5.99 | 4.74 |
| Sharpe Ratio | 0.24 | 0.16 | 0.22 | 0.44 | 0.31 | 0.64 | 0.33 | 0.61 |
| Avg. N | 50.08 | 23.61 | 24.78 | 24.08 | 23.05 | 18.02 | 4.97 | 12.68 |
| ***Bottom 75%:*** | | | | | | | | |
| Total return | 53.05 | 93.77 | 85.74 | 93.86 | 60.01 | 70.96 | 95.47 | 83.24 |
| Avg. monthly return | 0.90 | 1.59 | 1.45 | 1.59 | 1.02 | 1.20 | 1.62 | 1.41 |
| Std. Dev. | 5.85 | 8.12 | 7.49 | 5.28 | 5.58 | 5.98 | 7.5 | 5.09 |
| Sharpe Ratio | 0.43 | 0.61 | 0.59 | 0.93 | 0.53 | 0.60 | 0.67 | 0.84 |
| Avg. N | 50.1 | 20.53 | 18.27 | 11.78 | 10.29 | 14.05 | 3.5 | 9.54 |
| ***Bottom 50%:*** | | | | | | | | |
| Total return | 55.6 | 135.05 | 143.48 | 114.50 | 71.12 | 33.83 | 117.83 | 98.61 |
| Avg. monthly return | 0.94 | 2.29 | 2.43 | 1.94 | 1.21 | 0.57 | 2.00 | 1.67 |
| Std. Dev. | 6.69 | 8.56 | 9.95 | 10.37 | 8.73 | 8.08 | 11.33 | 6.26 |
| Sharpe Ratio | 0.40 | 0.86 | 0.79 | 0.59 | 0.41 | 0.17 | 0.56 | 0.83 |
| Avg. N | 33.41 | 12.37 | 8.54 | 1.54 | 3.31 | 7.05 | 1.34 | 4.08 |
| ***Bottom 25%:*** | | | | | | | | |
| Total return | 110.85 | 247.68 | 409.73 | 248.59 | 38.28 | -4.69 | 123.37 | 207.43 |
| Avg. monthly return | 1.88 | 4.20 | 6.94 | 4.21 | 0.65 | -0.08 | 2.09 | 3.52 |
| Std. Dev. | 10.95 | 15.14 | 27.03 | 15.18 | 10.4 | 10.11 | 14.41 | 14.29 |
| Sharpe Ratio | 0.54 | 0.92 | 0.87 | 0.92 | 0.16 | -0.09 | 0.46 | 0.81 |
| Avg. N | 16.71 | 5.44 | 2.73 | 0.51 | 1.49 | 3.08 | 0.81 | 1.59 |

**Table A3.2:** Gross long-short return within firm sizes
Table A3.2 shows the returns and statistics the machine learning long-short portfolios achieve, after removing specific quantiles of firms based on market capitalization. All the firms were sorted into four quartiles based on their market capitalization. "All firms" show the numbers for the entire test-sample. "Top 75%" show the numbers for the numbers after removing smallest quartile. "Bottom 75%" show the numbers after removing the largest quartile. "Bottom 50%" show the numbers by only including the smallest and second smallest quartiles. "Bottom 25%" shows the number for just the smallest quartile. "Total return" is the cumulative return at the end of period. "Avg. monthly return" is the average monthly return. "Std. Dev." is the standard deviation of the monthly returns. "Sharpe Ratio" is the Sharpe ratio. "Avg. Long-N" is the average number of stocks in the long-part of the portfolio each month. "Avg. Short-N" is the average number of stocks in the short-part of the portfolio each month.

| | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 |
| **Long-Short** | | | | | | | | |
| ***All firms:*** | | | | | | | | |
| Total return | - | 55.39 | 34.75 | 19.21 | -19.2 | 7.79 | 17.75 | 26.18 |
| Avg. monthly return | - | 0.94 | 0.59 | 0.33 | -0.33 | 0.13 | 0.30 | 0.44 |
| Std. Dev. | - | 3.91 | 3.41 | 4.52 | 4.59 | 4.48 | 5.21 | 4.12 |
| Sharpe Ratio | - | 0.68 | 0.43 | 0.12 | -0.37 | -0.03 | 0.09 | 0.23 |
| Avg. Long-N | - | 29.05 | 27.51 | 25.27 | 24.54 | 21.10 | 5.78 | 14.27 |
| Avg. Short-N | - | 37.75 | 39.29 | 41.53 | 42.25 | 45.69 | 61.02 | 52.53 |
| ***Top 75%:*** | | | | | | | | |
| Total return | - | -1.62 | 1.29 | 26.91 | 6.00 | 46.93 | 19.7 | 45.64 |
| Average monthly return | - | -0.03 | 0.02 | 0.46 | 0.10 | 0.80 | 0.33 | 0.77 |
| Standard deviation | - | 3.5 | 3.27 | 2.89 | 3.05 | 2.96 | 4.28 | 3.29 |
| Sharpe Ratio | - | -0.20 | -0.16 | 0.34 | -0.08 | 0.73 | 0.13 | 0.64 |
| Avg. Long-N | - | 23.61 | 24.78 | 24.08 | 23.05 | 18.02 | 4.97 | 12.68 |
| Avg. Short-N | - | 26.47 | 25.31 | 26.00 | 27.03 | 32.07 | 45.12 | 37.41 |
| ***Bottom 75%:*** | | | | | | | | |
| Total return | - | 53.05 | 56.08 | 55.99 | 8.50 | 24.99 | 44.13 | 46.06 |
| Average monthly return | - | 0.90 | 0.95 | 0.95 | 0.14 | 0.42 | 0.75 | 0.78 |
| Standard deviation | - | 5.85 | 4.88 | 5.13 | 5.47 | 5.68 | 7.17 | 5.11 |
| Sharpe Ratio | - | 0.43 | 0.55 | 0.53 | -0.02 | 0.15 | 0.28 | 0.41 |
| Avg. Long-N | - | 20.53 | 18.27 | 10.78 | 11.78 | 14.05 | 3.51 | 9.54 |
| Avg. Short-N | - | 29.58 | 31.83 | 39.32 | 38.32 | 36.05 | 36.59 | 40.56 |
| ***Bottom 50%:*** | | | | | | | | |
| Total return | - | 124.43 | 120.80 | 64.23 | 14.27 | -29.19 | 60.59 | 53.54 |
| Average monthly return | - | 2.11 | 2.05 | 1.09 | 0.24 | -0.49 | 1.03 | 0.91 |
| Standard deviation | - | 6.86 | 7.34 | 10.31 | 8.85 | 8.05 | 12.44 | 6.96 |
| Sharpe Ratio | - | 0.98 | 0.89 | 0.31 | 0.03 | -0.29 | 0.24 | 0.37 |
| Avg. Long-N | - | 12.37 | 8.54 | 1.54 | 3.31 | 7.05 | 1.34 | 4.08 |
| Avg. Short-N | - | 21.03 | 24.86 | 31.86 | 30.10 | 26.36 | 32.07 | 29.32 |
| ***Bottom 25%:*** | | | | | | | | |
| Total return | - | 206.18 | 360.46 | 151.16 | -82.65 | -146.09 | 15.87 | 97.91 |
| Average monthly return | - | 3.49 | 6.11 | 2.56 | -1.40 | -2.48 | 0.27 | 1.66 |
| Standard deviation | - | 11.71 | 23.29 | 17.78 | 14.74 | 12.43 | 16.41 | 16.72 |
| Sharpe Ratio | - | 0.98 | 0.88 | 0.47 | -0.37 | -0.74 | 0.02 | 0.31 |
| Avg. Long-N | - | 5.44 | 2.73 | 0.51 | 1.49 | 3.08 | 0.81 | 1.59 |
| Avg. Short-N | - | 11.27 | 13.98 | 16.2 | 15.22 | 13.63 | 15.90 | 15.12 |

**Table A3.3:** Net returns summary w/o top 25%
Descriptive summary table of the net portfolio performances of the machine learning algorithms. The table reports the returns from the portfolios constructed from the model predictions on the test-sample, excluding the largest market capitalization quartile. The test sample consist of 59 months of returns from January 2016 until November 2020. "Total Return" (%) is the total cumulative return achieved during the period. "Avg. Return" (%) shows the monthly average return. "Std. Dev." (%) is the standard deviation of the monthly returns. "Sharpe Ratio" is the annualized Sharpe ratio. "Skewness" is skewness. "Kurtosis" is kurtosis. "Max DD" (%) is the maximum drawdown the portfolio experiences. "Max 1M Loss" (%) is the largest monthly loss the portfolio suffers during the period. "Avg. N" shows the average number of stocks that are included in the portfolio each month.

| 2016 - 2020 | *Benchmarks* | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| *rf = 0.17%* | OSEBX | 1/N | XGB | RF | Log | SVM | NN1 | NN2 | NN3 |
| **Long-Only** | | | | | | | | | |
| Total return | 49.83 | 21.48 | 58.73 | 51.25 | 32.64 | 61.97 | 37.35 | 74.41 | 50.54 |
| Avg. Return | 0.84 | 0.36 | 1.00 | 0.87 | 0.55 | 1.05 | 0.63 | 1.26 | 0.86 |
| Std. Dev. | 4.46 | 5.97 | 7.85 | 7.70 | 5.74 | 5.52 | 6.10 | 7.99 | 5.26 |
| Sharpe Ratio | 0.52 | 0.11 | 0.36 | 0.31 | 0.23 | 0.55 | 0.26 | 0.47 | 0.45 |
| Skew | -0.28 | -0.34 | -0.24 | 0.07 | -1.32 | -1.00 | -0.42 | -0.67 | -0.94 |
| Kurtosis | 3.62 | 3.58 | 5.39 | 4.98 | 4.88 | 4.59 | 1.97 | 3.08 | 2.55 |
| Max DD | 24.63 | 44.09 | 51.36 | 46.83 | 41.94 | 32.04 | 39.00 | 37.34 | 29.34 |
| Max 1M Loss | -14.83 | -22.64 | -31.63 | -29.02 | -24.85 | -22.66 | -21.66 | -23.54 | -19.58 |
| Avg. N | | 50.10 | 20.53 | 18.27 | 11.78 | 10.29 | 14.05 | 3.5 | 9.54 |
| **Long-Short** | | | | | | | | | |
| Total return | | | 24.81 | 28.80 | -10.32 | 31.53 | -1.59 | 31.70 | 20.51 |
| Avg. Return | | | 0.42 | 0.49 | -0.17 | 0.53 | -0.03 | 0.54 | 0.35 |
| Std. Dev. | | | 5.30 | 5.12 | 5.36 | 5.17 | 5.59 | 7.34 | 5.05 |
| Sharpe Ratio | | | 0.16 | 0.22 | -0.22 | 0.24 | -0.12 | 0.17 | 0.12 |
| Skew | | | -0.01 | 0.74 | -2.24 | -2.95 | -2.72 | -0.21 | -1.86 |
| Kurtosis | | | 0.79 | 2.18 | 9.24 | 13.96 | 14.27 | 1.98 | 5.49 |
| Max DD | | | 45.93 | 37.42 | 29.46 | 31.68 | 37.06 | 34.07 | 26.96 |
| Max 1M Loss | | | -14.65 | -10.26 | -27.32 | -28.26 | -31.23 | -25.05 | -22.25 |
| Avg. N | | | 50.10 | 50.10 | 50.10 | 50.10 | 50.10 | 50.10 | 50.10 |

**Table A3.4:** Net returns summary w/o top 50%
Descriptive summary table of the net portfolio performances of the machine learning algorithms. The table reports the returns from the portfolios constructed from the model predictions on the test-sample, excluding the two largest market capitalization quartiles. The test sample consist of 59 months of returns from January 2016 until November 2020. "Total Return" (%) is the total cumulative return achieved during the period. "Avg. Return" (%) shows the monthly average return. "Std. Dev." (%) is the standard deviation of the monthly returns. "Sharpe Ratio" is the annualized Sharpe ratio. "Skew" is skewness. "Kurtosis" is kurtosis. "Max DD" (%) is the maximum drawdown the portfolio experiences. "Max 1M Loss" (%) is the largest monthly loss the portfolio suffers during the period. "Avg. N" shows the average number of stocks that are included in the portfolio each month.

| 2016 - 2020 | *Benchmarks* | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| *rf = 0.17%* | OSEBX | 1/N | XGB | RF | Log | SVM | NN1 | NN2 | NN3 |
| **Long-Only** | | | | | | | | | |
| Total return | 49.83 | 24.05 | 110.94 | 119.23 | 47.66 | 96.86 | 10.12 | 101.37 | 75.04 |
| Avg. Return | 0.84 | 0.41 | 1.88 | 2.02 | 0.81 | 1.64 | 0.17 | 1.72 | 1.27 |
| Std. Deviation | 4.46 | 6.76 | 8.52 | 9.91 | 8.69 | 10.31 | 8.05 | 11.26 | 6.24 |
| Sharpe Ratio | 0.52 | 0.12 | 0.70 | 0.65 | 0.25 | 0.49 | 0.00 | 0.48 | 0.61 |
| Skew | -0.28 | 0.16 | 0.93 | 1.42 | -0.30 | 1.19 | -0.31 | 2.59 | -0.26 |
| Kurtosis | 3.62 | 3.83 | 6.39 | 7.55 | 1.98 | 3.24 | 0.64 | 10.63 | 1.01 |
| Max DD | 24.63 | 49.65 | 30.30 | 42.58 | 39.72 | 65.40 | 55.02 | 30.88 | 0.30 |
| Max 1M Loss | -14.83 | -23.73 | -27.86 | -28.99 | -30.56 | -18.83 | -25.36 | -16.41 | -18.87 |
| Avg. N | | 33.41 | 12.37 | 8.54 | 3.31 | 1.54 | 7.05 | 1.34 | 4.08 |
| **Long-Short** | | | | | | | | | |
| Total return | | | 100.36 | 96.74 | -9.36 | 40.40 | -52.65 | 36.77 | 29.75 |
| Avg. Return | | | 1.70 | 1.64 | -0.16 | 0.68 | -0.89 | 0.62 | 0.50 |
| Std. Deviation | | | 6.83 | 7.31 | 8.82 | 10.27 | 8.02 | 12.39 | 6.93 |
| Sharpe Ratio | | | 0.78 | 0.70 | -0.13 | 0.17 | -0.46 | 0.13 | 0.17 |
| Skew | | | 0.25 | 1.56 | -0.54 | 1.01 | -2.39 | 1.98 | -1.34 |
| Kurtosis | | | 1.93 | 6.42 | 2.96 | 4.89 | 11.06 | 8.90 | 5.49 |
| Max DD | | | 31.55 | 21.17 | 50.77 | 54.41 | 60.40 | 52.91 | 37.35 |
| Max 1M Loss | | | -18.84 | -14.96 | -34.47 | -32.34 | -43.32 | -31.23 | -30.36 |
| Avg. Long-N | | | 12.37 | 8.54 | 3.31 | 1.54 | 7.05 | 1.34 | 4.08 |
| Avg. Short-N | | | 21.03 | 24.86 | 30.10 | 31.86 | 26.36 | 32.07 | 29.32 |

**Table A3.5:** Net returns of long-only w/o top 25%, by year
The table below displays the yearly returns of machine learning portfolios, after excluding the top 25% of firm sizes. The returns are compared to the entire OSEBX and a 1/N portfolio of our sample with the same exclusion criteria. The table shows the total net returns gained in the year and the average monthly net return for each year.

| | | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | OSEBX | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 |
| **Total net return:** | | | | | | | | | |
| 2016 | 12.10 | 8.61 | 24.00 | 24.10 | 18.50 | 17.30 | 30.70 | 20.10 | 25.90 |
| 2017 | 17.90 | 12.00 | 38.40 | 37.50 | 4.37 | -0.38 | -9.50 | 26.80 | 12.00 |
| 2018 | -1.05 | -21.00 | -25.90 | -17.80 | -4.57 | 7.58 | -16.70 | 2.41 | -7.59 |
| 2019 | 15.60 | 10.10 | 10.50 | 10.50 | 19.10 | -2.17 | 14.30 | 16.30 | 27.50 |
| 2020 | 5.27 | 11.90 | 11.80 | -3.11 | 24.60 | 10.00 | 18.50 | 8.79 | -7.24 |
| **Average monthly net return:** | | | | | | | | | |
| 2016 | 1.01 | 0.72 | 2.00 | 2.01 | 1.54 | 1.44 | 2.56 | 1.68 | 2.15 |
| 2017 | 1.49 | 1.00 | 3.20 | 3.13 | 0.36 | -0.03 | -0.79 | 2.23 | 1.00 |
| 2018 | -0.08 | -1.75 | -2.16 | -1.49 | -0.38 | 0.65 | -1.39 | 0.20 | -0.63 |
| 2019 | 1.30 | 0.84 | 0.87 | 0.88 | 1.59 | -0.18 | 1.19 | 1.36 | 2.29 |
| 2020 | 0.48 | 1.08 | 1.07 | -0.28 | 2.23 | 0.91 | 1.68 | 0.80 | -0.66 |

**Table A3.6:** Effect and net return summary of bimonthly portfolios
Table A3.6 shows the effects of the market frictions if the portfolios were rebalanced bimonthly.

| | | | | BIMONTHLY REPOSITIONING | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | *Machine Learning Portfolios* | | | | | |
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 | |
| **Long-Only** | | | | | | | | | |
| Gross average return | 0.81 | 0.79 | 0.89 | 0.77 | 0.48 | 0.75 | 0.42 | 0.65 | Mean |
| *Returns after individual effects:* | | | | | | | | | effect |
| - VWAP | 0.73 | 0.68 | 0.84 | 0.78 | 0.50 | 0.67 | 0.50 | 0.61 | -0.03 |
| - Brokerage fee (20 bp) | 0.56 | 0.58 | 0.68 | 0.57 | 0.27 | 0.54 | 0.22 | 0.45 | -0.21 |
| - Excluding largest 25% | 0.91 | 0.89 | 1.13 | 1.36 | 0.69 | 0.98 | 1.33 | 0.59 | 0.29 |
| *Returns after all effects:* | | | | | | | | | |
| Net average return | 0.63 | 0.60 | 0.87 | 1.07 | 0.46 | 0.70 | 1.09 | 0.35 | |
| **Long-Short** | | | | | | | | | |
| Gross average return | | -0.02 | 0.22 | -0.01 | -0.59 | -0.08 | -0.43 | -0.11 | Mean |
| *Returns after individual effects:* | | | | | | | | | effect |
| - VWAP | | -0.07 | 0.27 | 0.12 | -0.43 | -0.08 | -0.23 | -0.07 | 0.08 |
| - Brokerage fee (20 bp) | | -0.23 | 0.01 | -0.21 | -0.79 | -0.29 | -0.64 | -0.31 | -0.20 |
| - Interest rate (5%) | | -0.44 | -0.20 | -0.43 | -1.01 | -0.50 | -0.85 | -0.53 | -0.42 |
| - Excluding largest 25% | | -0.01 | 0.47 | 0.59 | -0.31 | 0.15 | 0.39 | -0.24 | 0.30 |
| *Returns after all effects:* | | | | | | | | | |
| Net average return | | -0.73 | -0.07 | -0.08 | -1.00 | -0.66 | -0.27 | -1.07 | |

**Table A3.7:** Net return summary bimonthly portfolios
Table A3.7 shows the summary statistics for the bimonthly portfolios, after considering market frictions and excluding the largest quartile of firms. The test sample consist of 59 months of returns from January 2016 until November 2020. "Total Return" is the total cumulative return achieved during the period. "Avg. Return" shows the monthly average return. "Std. Dev." is the standard deviation of the monthly returns. "Sharpe Ratio" is the annualized Sharpe ratio. "Skew" is skewness. "Kurtosis" is kurtosis. "Max DD" is the maximum drawdown the portfolio experiences. "Max 1M Loss" is the largest monthly loss the portfolio suffers during the period.

| | | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | OSEBX | 1/N | XGB | RF | Log | SVM | NN1 | NN2 | NN3 |
| **Long-Only** | | | | | | | | | |
| Total return | 49.83 | 37.07 | 35.69 | 51.53 | 27.17 | 63.30 | 41.37 | 64.16 | 20.40 |
| Avg. Return | 0.84 | 0.63 | 0.60 | 0.87 | 0.46 | 1.07 | 0.70 | 1.09 | 0.35 |
| Std. Deviation | 4.46 | 5.94 | 7.90 | 7.20 | 5.73 | 5.29 | 6.00 | 8.41 | 5.58 |
| Sharpe Ratio | 0.52 | 0.27 | 0.19 | 0.34 | 0.18 | 0.59 | 0.31 | 0.38 | 0.11 |
| Skew | -0.28 | -0.36 | -0.16 | 0.20 | -1.37 | -0.41 | -0.51 | 0.80 | -0.51 |
| Kurtosis | 3.62 | 3.61 | 4.86 | 4.11 | 4.84 | 1.01 | 2.30 | 2.38 | 1.52 |
| Max DD | 24.63 | 38.93 | 51.40 | 41.12 | 41.95 | 28.56 | 37.89 | 36.71 | 27.65 |
| Max 1M Loss | -14.83 | -22.44 | -31.33 | -25.68 | -24.42 | -16.25 | -21.78 | -22.82 | -19.25 |
| **Long-Short** | | | | | | | | | |
| Total return | | | 43.36 | -4.08 | -59.15 | 4.64 | -38.69 | -15.64 | -62.96 |
| Avg. Return | | | 0.73 | -0.07 | -1.00 | 0.08 | -0.66 | -0.27 | -1.07 |
| Std. Deviation | | | 5.29 | 4.95 | 5.48 | 5.15 | 5.75 | 8.24 | 5.10 |
| Sharpe Ratio | | | 0.37 | -0.17 | -0.74 | -0.06 | -0.50 | -0.18 | -0.84 |
| Skew | | | 0.20 | 0.31 | -2.28 | -2.04 | -2.79 | 0.42 | -1.69 |
| Kurtosis | | | 0.77 | 1.17 | 9.63 | 9.97 | 14.36 | 2.13 | 4.43 |
| Max DD | | | 55.05 | 37.93 | 51.96 | 31.20 | 46.83 | 48.03 | 51.11 |
| Max 1M Loss | | | -14.77 | -12.36 | -29.10 | -26.60 | -32.89 | -25.95 | -22.84 |

**Table A3.8:** Portfolio Factor Exposure P-Values
Table A3.8 shows the p-values of the regression results from table 6.6. "Alpha" is the intercept. "Market beta" is the correlation coefficient between the portfolio and the market. "SMB", "HML", "PR1YR", "UMD", "LIQ" shows the correlation coefficients between the portfolios and the associated factor. "Adj. $R^2$" is the adjusted $R^2$.

| | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 |
| **Long-Only** | | | | | | | | |
| Alpha | 0.0053 | 0.8193 | 0.7507 | 0.5531 | 0.5074 | 0.3679 | 0.7110 | 0.4788 |
| Market beta | 0.0392* | 0.0000* | 0.0000* | 0.0000* | 0.0000* | 0.0000* | 0.0001* | 0.0000* |
| SMB | 0.0392* | 0.1297 | 0.0553 | 0.0002* | 0.0203* | 0.0003* | 0.7747 | 0.4330 |
| HML | 0.380 | 0.1089 | 0.0600 | 0.5642 | 0.6808 | 0.2402 | 0.6510 | 0.5387 |
| PR1YR | 0.8702 | 0.2437 | 0.3658 | 0.7821 | 0.7422 | 0.4740 | 0.9677 | 0.0892 |
| UMD | 0.8654 | 0.3048 | 0.7161 | 0.6321 | 0.7500 | 0.3856 | 0.6429 | 0.1176 |
| LIQ | 0.0163* | 0.0223* | 0.0404* | 0.3133 | 0.2979 | 0.6678 | 0.8931 | 0.9936 |
| **Long-Short** | | | | | | | | |
| Alpha | | 0.5200 | 0.3984 | 0.5812 | 0.9322 | 0.9006 | 0.4640 | 0.1401 |
| Market | | 0.0372* | 0.3212 | 0.1665 | 0.0114* | 0.0980 | 0.4799 | 0.0167* |
| SMB | | 0.9658 | 0.3628 | 0.1615 | 0.3895 | 0.0901 | 0.2856 | 0.4918 |
| HML | | 0.1347 | 0.0475* | 0.2453 | 0.8086 | 0.6995 | 0.9665 | 0.8222 |
| PR1YR | | 0.1317 | 0.2818 | 0.8516 | 0.9927 | 0.5782 | 0.9774 | 0.1985 |
| UMD | | 0.1807 | 0.8168 | 0.7165 | 0.9465 | 0.4754 | 0.7038 | 0.2446 |
| LIQ | | 0.1877 | 0.3638 | 0.1852 | 0.8304 | 0.0268* | 0.2051 | 0.1029 |

**Table A3.9:** Long-Only Information Ratios
Table A3.9 shows the information ratios for the different long-only portfolios after excluding certain market capitalization quartiles.

| | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 |
| **Long-Only** | | | | | | | | |
| **All firms:** | | | | | | | | |
| Information ratio | -10.06 | 0.97 | -2.18 | -6.76 | -14.93 | -7.31 | 5.91 | -3.89 |
| Tracking Error | 0.03 | 0.04 | 0.04 | 0.02 | 0.03 | 0.03 | 0.06 | 0.03 |
| **Top 75%:** | | | | | | | | |
| Information ratio | -26.79 | -16.84 | -17.34 | -15.15 | -17.32 | -5.54 | -6.78 | -4.73 |
| Tracking Error | 0.02 | 0.03 | 0.03 | 0.02 | 0.02 | 0.03 | 0.04 | 0.03 |
| **Bottom 75%:** | | | | | | | | |
| Information ratio | -8.54 | 1.69 | 0.26 | 3.59 | -4.49 | -3.24 | 4.02 | 0.20 |
| Tracking Error | 0.03 | 0.05 | 0.05 | 0.03 | 0.04 | 0.04 | 0.06 | 0.03 |
| **Bottom 50%:** | | | | | | | | |
| Information ratio | -5.79 | 8.84 | 8.10 | 5.09 | -0.28 | -6.66 | 4.75 | 5.21 |
| Tracking Error | 0.04 | 0.07 | 0.09 | 0.09 | 0.08 | 0.06 | 0.11 | 0.05 |
| **Bottom 25%:** | | | | | | | | |
| Information ratio | 6.04 | 12.17 | 12.59 | 12.36 | -2.90 | -9.53 | 4.18 | 10.15 |
| Tracking Error | 0.09 | 0.14 | 0.27 | 0.15 | 0.10 | 0.08 | 0.14 | 0.13 |

**Table A3.10:** Long-Short Information Ratios
Table A3.10 shows the information ratios for the different long-short portfolios after excluding certain market capitalization quartiles.

| | | *Machine Learning Portfolios* | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1/N | XGB | RF | SVM | Log | NN1 | NN2 | NN3 |
| **Long-Short** | | | | | | | | |
| **All firms:** | | | | | | | | |
| Information ratio | | -3.67 | -7.38 | -7.49 | -12.49 | -9.79 | -8.11 | -7.16 |
| Tracking Error | | 0.05 | 0.05 | 0.07 | 0.07 | 0.07 | 0.07 | 0.07 |
| **Top 75%:** | | | | | | | | |
| Information ratio | | -15.14 | -12.83 | -7.85 | -11.20 | -4.54 | -8.69 | -4.70 |
| Tracking Error | | 0.05 | 0.06 | 0.06 | 0.06 | 0.06 | 0.06 | 0.06 |
| **Bottom 75%:** | | | | | | | | |
| Information ratio | | -4.32 | -3.40 | -2.47 | -7.85 | -6.92 | -2.14 | -3.92 |
| Tracking Error | | 0.06 | 0.06 | 0.07 | 0.08 | 0.07 | 0.08 | 0.07 |
| **Bottom 50%:** | | | | | | | | |
| Information ratio | | 6.18 | 5.52 | -0.83 | -5.61 | -11.39 | -0.95 | -2.29 |
| Tracking Error | | 0.08 | 0.08 | 0.11 | 0.11 | 0.09 | 0.14 | 0.09 |
| **Bottom 25%:** | | | | | | | | |
| Information ratio | | 10.36 | 12.05 | 4.05 | -9.31 | -15.94 | -3.28 | 1.38 |
| Tracking Error | | 0.13 | 0.24 | 0.19 | 0.17 | 0.14 | 0.18 | 0.17 |