Norwegian School of Economics

Bergen, Spring, 2022

Predicting corporate bond returns in the US bond market via machine learning

By

Fredrik Storihle
Erlend L. Støylen

Supervisor: Nils Friewald

Master thesis, Economics and Business Administration

Major: Financial Economics/Business Analytics

Norwegian School of Economics

Acknowledgements

We would like to thank our supervisor, Nils Friewald, for valuable guidance and constructive feedback throughout the writing process. His expertise within the US corporate bond market and insights have been essential for the quality of this paper.

Norwegian School of Economics

Bergen, June 2022

Fredrik Storihle

Erlend L. Støylen

Abstract

We perform a comparative analysis of two machine learning methods to predict corporate bond return in the US bond market. In contrast to previous studies, we find that the most influential variables are associated with size risk and past return. However, credit and liquidity risks are more prominent when negative externalities impact the market. Further, high predictability at short horizons combined with the investment strategy employed translates into highly significant alphas. We identify the best-performing method to be a decision-tree-based model utilizing boosting. The out-of-sample performance for this method remains statistically significant after accounting for transaction costs.

Keywords— R, LaTeX, Machine learning, Corporate bonds

Contents

1 Introduction 1

2 Literature Review 6

3 Data Description 9

4 Models 14

4.1 Fama-French model 14

4.2 Machine learning 15

4.2.1 Ensemble methods 15

4.2.2 Bootstrapping, Bagging & Boosting 16

4.2.3 Model validation 17

5 Methodology 19

5.1 Dependent variable 19

5.2 Independent variables 19

5.3 Period of interest 22

5.4 Machine learning methods 22

5.4.1 XGBoost 22

5.4.2 Random forest 23

5.5 Trading mechanism 24

6 Results 26

6.1 Model accuracy 26

6.2 Variable importance 27

6.3 Portfolio positions 29

6.4 Return series 31

6.5 Alpha 33

7 Conclusion 38

8 Critique and further research 39

9 Bibliography 40

10 Appendix 42

List of Figures

1 Transaction volume by sub-product 10

2 Bagging process 16

3 Boosting process 17

4 Cross-validation 18

5 Model performance 26

6 Variable importance XGB 28

7 Variable importance Random Forest 29

8 Portfolio positions 30

9 Portfolio performance 32

10 Sharpe ratio 35

11 Cumulative return 37

A1 Marginal decrease in MSE 42

A2 Cumulative return for Institutional investors 44

A3 Cumulative return for Retail investors 44

List of Tables

1 Outcome from filters 12

2 Summary statistics 13

3 Variable explanation 21

4 Fama-regression 34

A1 Fama-regression for Institutional investors 42

A2 Fama-regression for Retail investors 43

A3 Sharpe ratio scenarios 43

1 Introduction

In recent years, stock and bond trading has received more attention from investors. This can be largely attributed to technological advances, lowering the barriers to entering the market, thus increasing the number of market participants. Additionally, market information has become more accessible, where investors take advantage of the extensive information available electronically to research potential investments (Report to the Congress: Impact of Technology on Securities Markets, 1997).

Further, computational power has become exponentially more powerful and inexpensive, allowing investors to utilize more computational demanding models. In other words, predicting the stock and bond market has entered a technologically advanced era, making it hard for investors to search for new tools and techniques to generate excess return while reducing risk (Rouf et al., 2021). This has never been an easy task, given the stochastic nature of returns. In fact, the Efficient Market Hypothesis (EMH) states that it is not possible.

The EMH explains that stocks and bonds always trade at their fair value and that all information in the market is reflected in the price. Consequently, the prices follow a random walk pattern, and thus cannot be predicted, making it impossible to generate consistent alpha. The EMH was a widely accepted theory in the past. However, with the advent of technology, hedge funds began to show evidence that market prices, to some extent, could be predicted, consequently being able to construct portfolios that yield returns in excess of the market.

One advanced technological approach is to use machine learning models to predict returns. Machine learning is used in stock prediction to discover patterns in the data, allowing analysts to quickly analyze more complex data while generating more accurate results, setting a new standard for trading models. There are many different machine learning models, but those categorized as ensemble decision trees, such as Random Forest and XGBoost, are believed to closely mirror human decision-making (James et al., 2013), partially justifying their application to the financial markets. The latter is further supported by Gu et al. (2020), which emphasize that regression trees are more prominent than other methods in predicting returns and state that such methods can help improve the empirical understanding of asset pricing.

However, there are only a few acknowledged studies regarding machine learning methods to predict stock returns and an absence of literature taking advantage of machine learning to predict bond returns. The bond market is generally less analyzed than the stock market, especially when pricing individual securities. There is, conceivably, a considerable amount of undiscovered information regarding the use of machine learning models to predict the bond market, thus making this an exciting research topic for this paper.

The main focus of this paper is to determine the impact of the machine learning methods, Random Forest and XGBoost, as a tool to predict corporate bond returns. The main difference between Random Forest and XGBoost is that they utilize bagging and boosting, respectively, to increase prediction accuracy. Ensemble decision trees are known to discover complex data patterns. Thus, we want to test whether the complex data patterns discovered can generate a significant alpha while controlling for credit risk, liquidity risk, and other risk characteristics. In theory, any discoveries should be explained by the long-established 3-factor model, including two term-structure factors proposed by Fama and French (1993) or the liquidity measure proposed by Amihud (2002), indicating an efficient market.

For our empirical study, we use transactions from the Trade Reporting and Compliance Machine (TRACE). This database provides detailed information about all US corporate bond market transactions, maintained by the Financial Industry Regulatory Authority. Additional bond-specific information is sourced from the Mergent Fixed Income Database, along with quarterly company-specific financials from Compustat and credit ratings from Standard and Poor's. However, the TRACE data set contains reporting- and pricing errors due to human mistakes in the reporting process. If these errors are not accounted for, our analysis will be prone to bias concerning liquidity and return (Dick-Nielsen, 2009). Thus, filters proposed by Dick-Nielsen (2009) and Edwards, Harris, and Piwowar (2004) are employed, including other limitations necessary. Lastly, we aggregate the data to monthly frequency for each bond. The resulting data set consists of 141,794 transactions covering 3,186 bonds.

Furthermore, we construct a long-short portfolio, allowing for various ratios between long and short positions, with quarterly rebalancing. More specifically, we specify the model to place long positions into the top 5% performing bonds with a predicted return above zero. Conversely, we place short positions to the bottom 5% performing bonds with a predicted return below zero. In other words, due to quarterly rebalancing, we will predict only three months at a time, which will increase the accuracy compared to predicting the whole period at once. This is a sophisticated way of modeling, also called time-series cross-validation, enabling us to monitor the prediction accuracy over time. However, this leads to an extensive number of models to be estimated, resulting in a computationally demanding process, requiring certain limitations for feasibility.

Finally, to evaluate if the machine learning models can construct a portfolio generating a significant alpha over time, we regress the portfolios' realized return series on the long-established factors proposed by Fama and French (1993). We will use the stock-market factors, market premium (MKT), small minus big size (SMB), high minus low book-to-market (HML), and the term-structure factors term premium (TERM) and default premium (DEF). Additionally, we include a liquidity measure proposed by Amihud (2002), which has proven explanatory significance on bond returns in numerous articles.

Results from the Fama-regression indicate that machine learning methods bring better measurements of complex relationships, evidently through generating return excess of the market. Both portfolios provide highly significant alphas, with low corresponding adjusted R-squared values. This signalizes that the Fama-factors and Amihud can not entirely explain the variance of the excess return.

Nevertheless, these results do not account for transaction costs. According to Harris (2015), transaction costs vary between institutional and retail trades, being 52.1 and 84 basis points, respectively. Followingly, XGBoost provides a statistically significant performance even under the two scenarios of transaction costs. Whereas the Random Forest method becomes highly affected, turning insignificant for Retail investors.

More interestingly, although the results prove a significant return excess of the market, the Sharpe ratios are substantially lower than the market. This indicates that a large portion of the alpha is due to taking more risk. Thus, investors would prefer the market portfolio given the possibility of borrowing/lending money at a risk-free rate. Conversely, if this is not possible, investors with low-risk aversion might prefer the XGBoost portfolio, given its excess return.

The remainder of the paper is structured as follows: we present a short description of corporate bonds and the US over-the-counter (OTC) bond market. Section 2 presents a cohesive story of the relevant literature of the paper, focusing mainly on papers relating to Fama and French and machine learning. In section 3, we detailedly explain the composition of our data set, its matching procedures for combining the different data sets, and the filters employed. Section 4 presents the essential models used. Section 5 outlines the methodology, whereas section 6 presents the machine learning results. Finally, the last two sections, 7 and 8, present our conclusion and corresponding critiques, respectively.

Corporate bonds and the OTC-market

A corporate bond is a fixed income instrument categorized as debt security issued by corporations. In practice, the company sells a debt obligation to investors to finance capital expenditures. In return, the investor gets paid a pre-established number of interest payments, coupons, at a fixed rate or, in some cases, a variable interest rate. The interest payments cease when the bond reaches maturity, and the original investment, the principal amount, is paid back.

The ability of the company to repay depends on its outlook for future income and profitability, including the capability to meet short-term financial obligations. The least creditworthy bonds are categorized as “junk,” “speculative,” or “high-yield,” and the investors demand a premium for the increased credit risk. Conversely, bonds embedding a credit rating equal to or above “BBB” are categorized as “Investment-grade.” They are a collective designation for all bonds with low default risk, thus yielding a lower coupon. Investment-grade corporate bonds are considered a relatively safe and conservative investment, where investors tend to include these bonds to offset riskier investments, such as stocks. In general, bondholders have a higher claim on the company’s assets than equity holders. Consequently, in the event of default, all bondholders would be prioritized before shareholders regarding repayment, reducing its risk compared to stocks.

Unlike stocks traded on the stock exchange, most corporate bonds are traded OTC. The OTC-market has no central marketplace or clearinghouse and tends to be less- regulated and liquid than exchanges. However, after the near-collapse of the credit default swap market in 2008, this market has come under more strict monitoring (Friewald et al., 2012). A set of rules approved by the Securities and Exchange Commission legislates that all Corporate debt transactions conducted by a designated dealer must be reported to TRACE.

2 Literature Review

To be prepared for the upcoming analysis, we review relevant literature for both bonds and stocks that have contributory power to this paper's structure. Additionally, we will compare findings between the references and make a cohesive story.

Fama and French (1993) is considered a solid empirical paper on corporate bond pricing, expanding the 3-factor model with two term-structure factors, term premium and default premium. They have shown that both term- and default premium are important factors. This is also emphasized by other researchers, mainly through Fama and MacBeth's (1973) methodology, which can be seen as a cross-sectional regression developed to empirically test the Capital Asset Pricing Model.

Fama and French (1993) emphasize that, when used alone, stock market factors seem to capture common variation in bond returns. They are linked to bond returns through their shared variation in the term-structure factors, more specifically;

1. Since both bonds and stocks are the firm's claim on the value of the same underlying assets, size, and book-to-market could impact common variations in stock and bond returns.
2. Expected default loss fluctuates with the equity price, where the loss appreciates when the equity value decreases and vice versa.

However, when the two term-structure factors are included, the explanatory power of the stock-market factors disappears for all but the low-grade corporate bonds. Further, Bai et al. (2019) argue that using reconstructed factors, all being bond-specific, is more accurate. For example, reconstructing the MKT factor, originally the stock market excess return, to be a value-weighted bond market return in excess of the one-month Treasury bill. They state that the reconstructed factors have economic and significant risk premiums that cannot be explained by the long-established stock-market and term-structure factors.

Nevertheless, Lin et al. (2011) discovered that, after expanding Fama and French's bond-specific model (3-factor plus TERM and DEF) with a liquidity measure, bonds with high liquidity sensitivities are expected to generate approximately 4 % higher average returns than bonds with low sensitivity. Thus suggesting that liquidity risk is an important determinant of corporate bond returns. Similar findings are discussed by Friewald et al. (2012), which discover that liquidity effects stand for around 14% of the explained marketwide corporate bond yield spread changes and conclude that these measures are significantly more prominent during periods of crisis. However, Chung et al. (2019) state that liquidity measures cannot account for the high returns associated with high idiosyncratic bond volatility even though it tends to increase with liquidity.

Further, due to technological advances, new methods categorized as machine learning can handle large amounts of complex data while discovering relationships and patterns more accurately than older models. Gu et al. (2020) describe that the overall success of machine learning algorithms for return predictions brings promise for practical aspects of portfolio choice. Moreover, they state that, by using return prediction as a proving ground, their findings through neural networks and regression trees demonstrate that machine learning methods can help improve the empirical understanding of asset pricing. Such findings contribute to the justification of the growing role of machine learning as a tool to predict financial returns.

Using Gu et al. (2020) article as a framework, Leippold et al. (2021) wanted to test if they achieved similar results if applied to emerging markets. They tested multiple machine learning methods in predicting the Chinese stock market which has entirely different characteristics than the US stock market. Further, they constructed two types of portfolios, a zero-net-investment, and one solely holding long positions. Their findings are robust, and they discuss that machine learning methods can be even more successfully applied to such markets, emphasizing the generalization capabilities of machine learning.

Further, Harris (2015) investigate transaction costs in the corporate bond market and found that transaction costs are higher than equities but decrease significantly with trade size. Additionally, bonds with a high rating, recently issued, and bonds close to maturity have lower transaction costs. Their results suggest that more transparent bond prices would be beneficial, especially for retail investors. Given major differences regarding transaction costs, Leippold et al. (2021) emphasize that one must include transaction costs to assess the economic significance. Through transaction cost scenarios, they discover that the different strategies' performance remains economically significant, also under conservative assumptions regarding the magnitude of the transaction costs.

3 Data Description

This section presents the composition of our data set, its matching procedures, and the filters employed. Acquiring and merging data from different sources enables us to analyze various parameters impacting bond returns. The individual data sets we have merged are;

1. Transaction data from the Trade Reporting and Compliance Engine (TRACE).
2. Bond-specific information from Mergent Fixed Income Database.
3. Credit ratings from Standards & Poor's.
4. Firm financials from Compustat/CRSP.
5. Risk factors (Fama & French).

TRACE provides detailed information about all US corporate bond market transactions and is maintained by the Financial Industry Regulatory Authority. A set of rules approved by the Securities and Exchange Commission legislates that all transactions of treasuries, securitized products, agency bonds, and corporate debt conducted by a designated dealer must be reported to TRACE. The complete TRACE data set consists of 278,582,266 transactions for 248,489 bonds. The time period of the data stretches from 1st of July 2002 to 31st of March 2021. Given its size, this data set is computationally demanding, and for this thesis, we filter out bonds embedding specific characteristics to reduce the computational requirement.

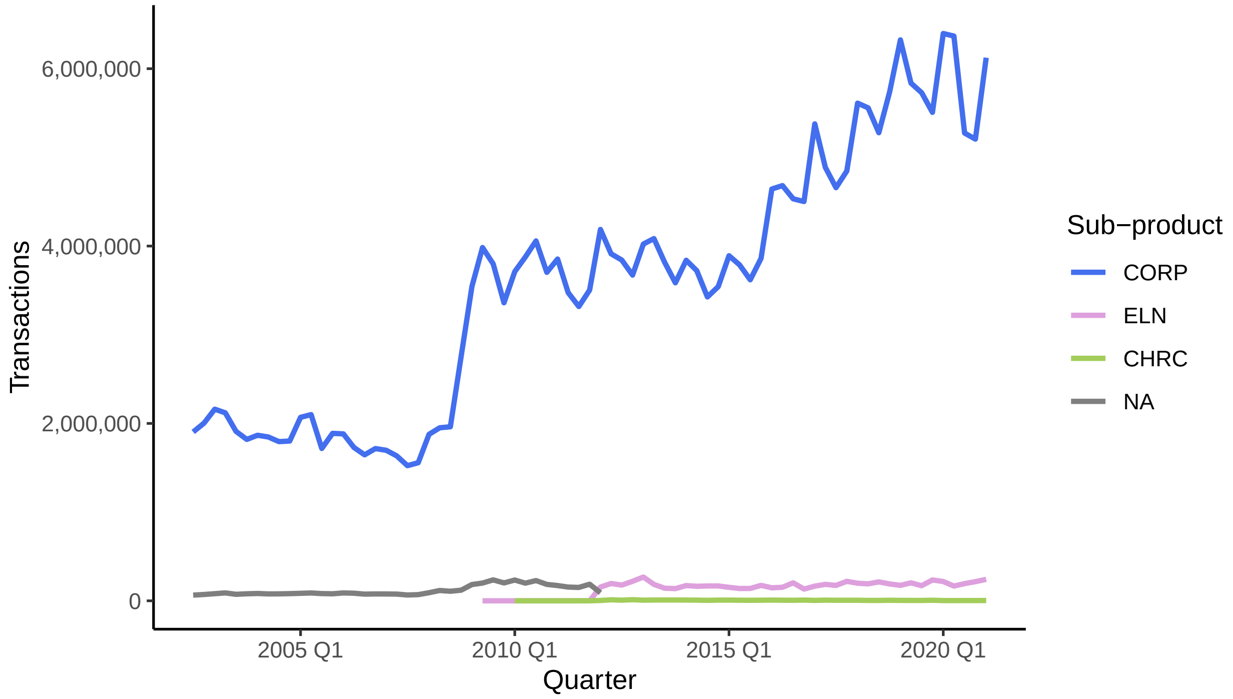

Figure 1: Transaction volume by sub-product

This plot visualizes the quarterly number of transactions per sub-product. CORP stands for Corporate bonds, ELN is equity-linked notes, and CHRC is short for Church bonds. Lastly, some bonds have not received any classifications, thus addressed as NA.

From figure 1, we observe the quarterly number of transactions for the entire TRACE data set. The number of bonds being bought and sold each quarter has increased since the beginning of the data collection. We see a substantial increase after the financial crisis and a steady incline from 2016. Further, we restrict our data only to contain corporate bonds.

Additional information about the bond's characteristics is sourced from the Mergent Fixed Income Database, containing issue details, such as offering amount, offering date, coupon rate, and maturity. After merging the data set by matching CUSIP IDs, we filter out all bonds with a maturity of less than six months and above 30 years. Next, we exclude financial and utility firms from our data due to vast inequalities in capital structures. More specifically, using the Standard Classification Code (SIC), we differentiate financial firms and utility firms between SIC 6000-6999 and 4900-4999. Additionally, bonds categorized as convertible, perpetual, putable, privately placed, exchangeable, preferred securities, or quoted in a foreign currency are also removed.

After filtering for all the above limitations, the data set contains 105,139,947 transactions covering 69,016 unique bonds. Next, we account for reporting errors using the filters proposed by Dick-Nielsen (2009). This filtering mainly focuses on deleting true duplicates, reversals, and adjustments regarding same-day corrections. A reporting error can easily be corrected if the correction is made within the same day as the original report was filed. However, the original report containing the error will still remain in the TRACE database. Moreover, if a correction is made on a later date, a report identical to the original report marked as "Reversal" is filed. Then, a new report is filed containing the correct information for the transaction. Failing to correct these errors will bias liquidity measures and return (Dick-Nielsen, 2009). Additionally, we follow Edwards, Harris, and Piwowar (2004) to eliminate data errors subject to its pricing. They propose two filters to detect potential outliers in the reported prices;

1. A daily median filter is applied to the bonds experiencing low price deviations. Oppositely, a nine-trading-day median centered on the trading day is used on bonds experiencing high price deviations. For both median-filters, elimination of transactions occurs if the absolute price deviation is greater than 10%.
2. A reversal filter to identify any unusual price movements. This filter eliminates transactions with an absolute price change deviating from the lead, lag, and mean of lead plus lag price change by at least 10%.

Finally, we exclude all transactions with a volume that is not an integer and those executed before the first active date or after the maturity date. Now that both the reporting error and pricing error filters are employed, we will present the total loss of observations and bonds.

Table 1: Outcome from filters

This table presents the total loss of transactions and bonds due to the filters proposed by Dick-Nielsen (2009) and Edwards, Harris, and Piwowar (2004). Transactions refer to the number of observations removed. The "Bonds" column reflects that all transactions for that particular bond are removed.

Filter	**Transactions**	**Bonds**
Dick-Nielsen	16,259,625	405
Edwards, Harris and Piwowar	13,802,181	52,126

After running the above filters, the remaining data consist of 75,078,141 transactions covering 16,485 bonds. Further, we aggregate the data for each bond into monthly frequency based on end-of-month observations by volume-weighting intraday transactions. However, some bonds are not traded on the last possible trading day each month, and to balance this out, we accept the transaction if it is observed at most three trading days before. After converting to monthly frequency, our data set consists of 495,015 transactions covering 12,330 unique bonds. Next, we use Standard & Poor's credit ratings to control for default risk. We factorize the credit-rating variable by assigning integer numbers (e.g., AAA $= 22$, AA+ $= 21, \dots$) to address its summary statistics. Lastly, we filter out all bonds that have not received a rating.

Quarterly reported firm-specific financials are sourced from Compustat and merged by matching CUSIP IDs. This data set provides information such as; cash holdings, equity, outstanding shares, share price, total debt, and total assets. Firm-specific information will be essential in our models. Thus we exclude all bonds related to firms not included in Compustat. The resulting data set consists of 141,794 transactions covering 3,186 unique bonds. A summary statistics of the final data set is presented below.

Table 2: Summary statistics

This table presents summary statistics of our combined data set. The table includes the number of observations, mean, standard deviation, and 5%, 50%, and 95% quantiles. We assign integer numbers to the credit ratings, where AAA = 22, AA = 21, A = 20 and so on. Offering amount is presented in units of millions, and the thousand separator is ",".

Statistic	**N**	**Mean**	**SD**	**Q05**	**Q50**	**Q95**
Offering amount	141,794	781.98	655.72	250.00	600.00	2,000.00
Maturity (Years)	141,794	10.34	4.64	5.25	10.02	20.04
Coupon (in pct)	141,794	4.78	1.82	2.30	4.60	8.00
Credit rating	141,794	14.57	3.09	9.00	15.00	20.00

The average bond size in our sample is approximately $782 million, with a corresponding interquintile range between $250 million and $2 billion. Further, the average bond has a remaining time to maturity of 10.34 years and a coupon rate of 4,78%. The coupon rate shows large variations during the time period, with an interquintile range between 2.3% and 8%. Large variations in the coupon rate are also reflected in the credit rating, having an interquintile range between 9 (B+) and 20 (AA). The summary statistics suggest our sample contains a broad range of corporate bonds.

Finally, the three stock-market factors proposed by Fama and French (1993) MKT, HML, and SMB are downloaded from: **French's webpage**. Following Fama and French (1993), we calculate two term-structure factors, TERM, and DEF. We estimate TERM as the difference between a long-term government bond and the one-month Treasury bill. The DEF is estimated as the difference between the return of an investment-grade portfolio constructed from our sample and a long-term government bond. The rate for the one-month Treasury bill and the long-term government bond is sourced from the: **Federal Reserve Bank** (FED). These factors will be used to test if our portfolios can generate a significant alpha throughout the time period.

4 Models

This section will explain the models used to conduct our analysis, including Fama and French, and the underlying processes in XGBoost and Random Forest.

4.1 Fama-French model

Fama and French's model attempts to explain variations in the returns given its different exposures to the market. Their model uses three stock market factors, market risk premium (MKT), small minus big (SMB), and high minus low (HML). The MKT is defined as the stock returns in excess of the risk-free rate. SMB is the difference in return between a portfolio of small-sized stocks and a portfolio of big-sized stocks. HML is the return difference between a portfolio with high book-to-market stocks and a portfolio of low book-to-market stocks. Next, they include two term-structure factors, term premium (TERM) and default premium (DEF). TERM is the difference between the monthly long-term government bond return and the one-month Treasury bill. DEF is the difference between a market portfolio of long-term corporate bonds and the long-term government bond return. The regression equation is;

$$r_t - rf = \alpha + \beta_1 MKT_t + \beta_2 SMB_t + \beta_3 HML_t + \beta_4 TERM_t + \beta_5 DEF_t \tag{1}$$

From the equation above, $r_t - r_f$ is the portfolio return in excess of the risk-free rate at time t. The MKT beta measures the portfolio's exposure to the systematic risk, also referred to as non-diversifiable risk. The SMB beta measures the portfolio's exposure to size risk. The HML beta measures the portfolio's exposure to book-to-market, where high book-to-market stocks are considered riskier. Higher book-to-market stocks are being traded at a discount compared to their book value, reflecting low expected growth and return on equity. The TERM beta measures the portfolios's exposure to risk related to maturity. Long-term bonds are more sensitive to interest rate changes due to having a greater duration than short-term bonds. The DEF beta measures the portfolio's exposure to default risk. For example, bonds awarded a credit rating below “BBB” provides a return premium due to higher default risk. Finally, the intercept term is the alpha and should be indistinguishable from zero when regressing the realized returns from the market portfolio on the aforementioned factors.

4.2 Machine learning

Machine learning can roughly be divided into two categories;

1. Supervised learning is used to predict a dependent variable using independent variables.
2. Unsupervised learning is used to discover patterns within variables.

Supervised learning aims to fit a model that can use independent values to accurately predict the dependent variable by discovering complex relationships (James et al., 2013). One supervised approach uses machine learning models categorized as an ensemble decision tree, such as Random Forest and XGBoost, utilizing bagging or boosting, respectively, to improve prediction accuracy.

4.2.1 Ensemble methods

Ensemble learning suggests that the decision-making of a larger group of models is better than an individual model. The models will work collectively to achieve a better prediction. More specifically, ensembling refers to a single model as a weak learner that may not perform well due to high variance. However, when the weak learners are combined, they can form a strong learner (highly accurate model), as their combination reduces bias and variance (James et al., 2013). A typical implementation of the ensemble methods is using decision trees, which learn simple decision rules from the training data to predict the target variable. A decision tree is prone to overfitting, which means that a single-tree model provides high variance with a corresponding low bias. Conversely, underfitting means low variance and high bias. When overfitting or underfitting, it cannot generalize its discoveries well onto new data sets or unobserved data points. However, when the trees are ensembled, the idea is to counteract this behavior allowing for better model generalization. Bagging and boosting are ensemble learning methods that utilize bootstrapped samples.

4.2.2 Bootstrapping, Bagging & Boosting

Bootstrapping allows for the generation of new samples without collecting additional training data. More specifically, it draws samples with replacements from the original data. Bootstrap aggregation (Bagging) is when multiple models are trained on bootstrapped samples from the original training set (James et al., 2013). The final prediction will be the average of all models to increase prediction accuracy. The diagram below displays the process of bagging.

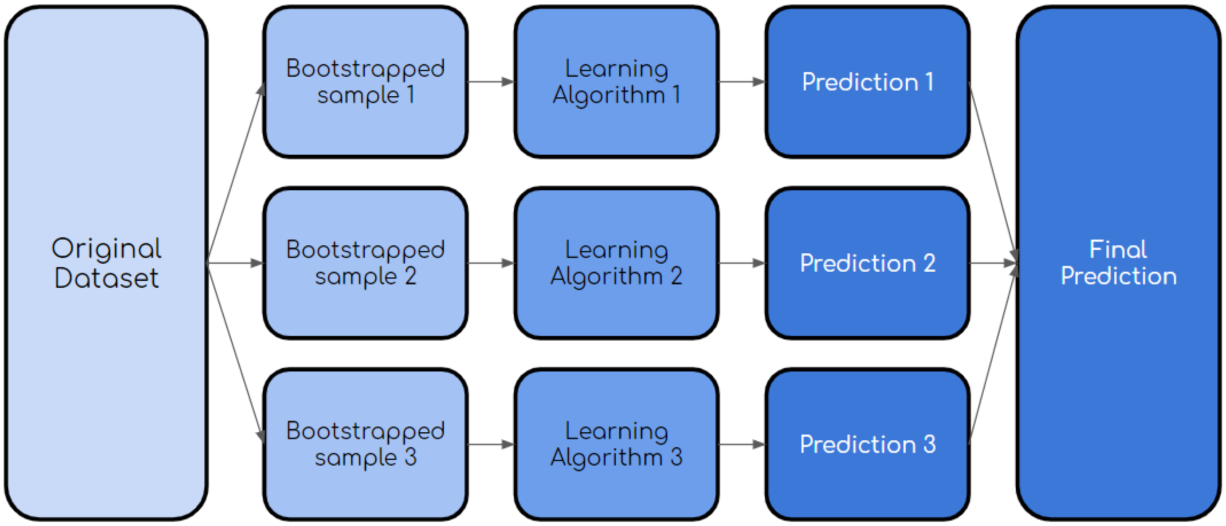

Figure 2: Bagging process

This Figure shows the process of bagging, and is sourced from: **Towards Data Science**. Learning algorithm refers to the machine learning model fitted on the bootstrapped training sample. This process can be expanded infinitely, only limited by computational power.

Conversely, boosting will fit an individual model on a bootstrapped sample, then use the residual errors of this model as the dependent variable in the next model when trained on a new bootstrapped sample (James et al., 2013). This process is repeated n number of times, where the final prediction is the last model's results. The diagram below displays the process of boosting.

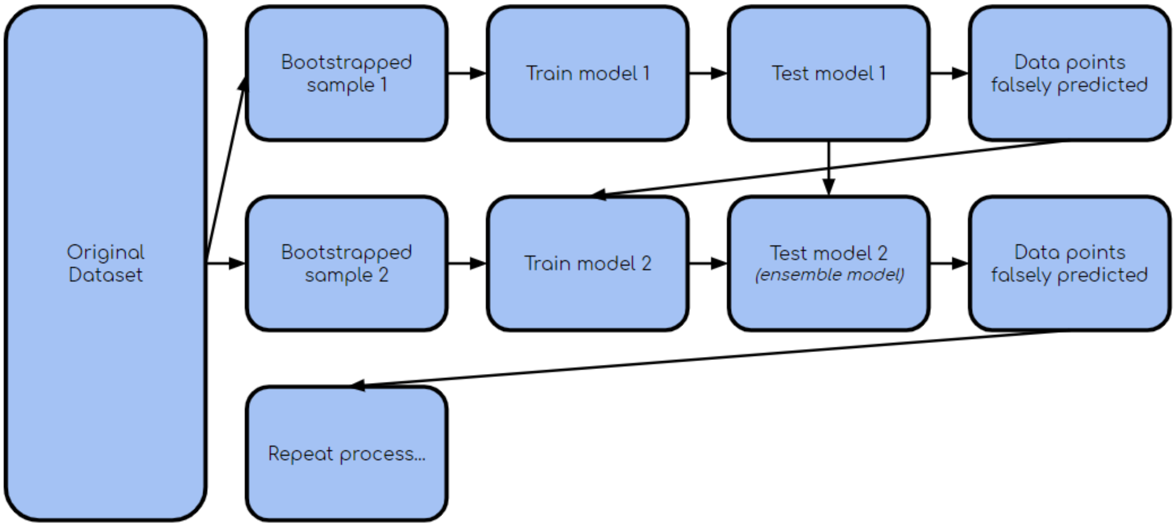

Figure 3: Boosting process

This figure shows the process of boosting, and is sourced from: **Towards Data Science**. Train model refers to the machine learning model fitted on the bootstrapped training sample. Test model refers to predicting. The last node, "Data points falsely predicted," refers to the residual error from model 1, which is then used as the dependent variable in model 2.

4.2.3 Model validation

To validate machine learning models, we divide the original data into two subsets called train- and test sets. More specifically, the models are fitted exclusively on the training set, and then the fitted model is validated on the test set. It is important to note that none of the observations in the training set is included in the test set. A more sophisticated version of training and testing sets is time-series cross-validation. This procedure produces numerous training and testing sets, where each test set consists of observations in a pre-specified time interval (Hyndman & Athanasopoulos, 2021). The corresponding train set consists of all observations that occured prior to the test set. This procedure is repeated throughout the entire time period for the data, adding the observations in the previous test set to the train set. The following figure will display an example of a series of training and test sets.

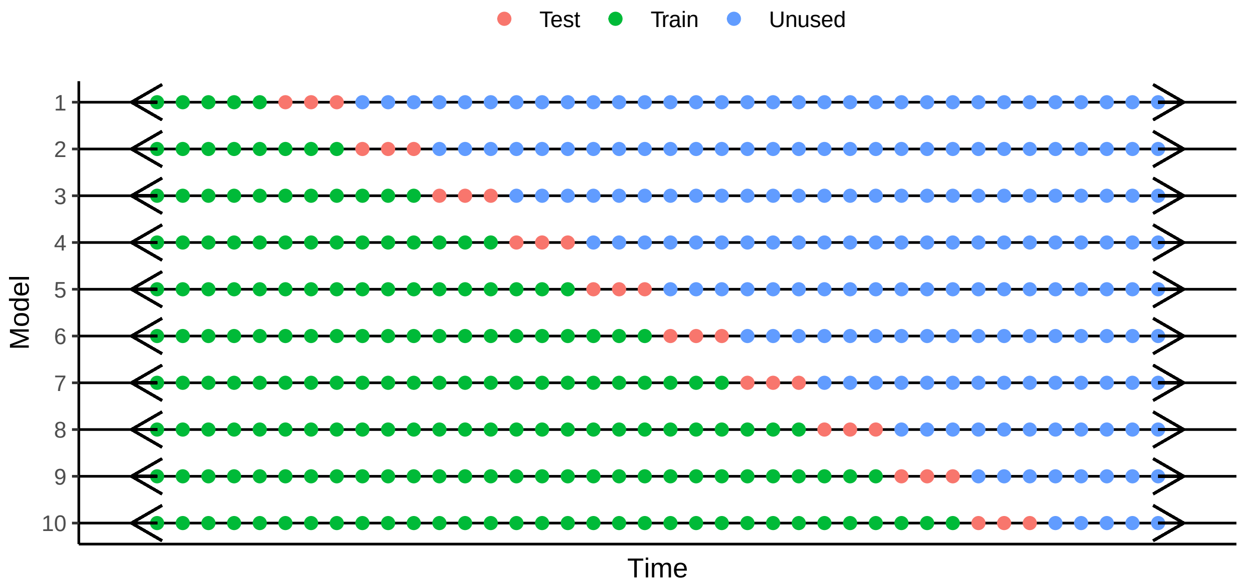

Figure 4: Cross-validation
The green color represents the training sets, expanding with three months for each model, where one point refers to all observations in that month. The red color represents the test sets, which are moved three months at a time. The blue color represents the remaining data points in the original data not used in the model.

This method allows the model to have the most data possible to predict the upcoming time interval. Additionally, the prediction accuracy will be higher when using shorter prediction horizons, compared to predicting the whole period at once (Leippold et al., 2021).

In order to evaluate the performance of a statistical learning method, we need to measure how well its predictions match the observed data. Quantifying the extent to which the predicted value for a given observation is close to the true observed value is essential to assess its accuracy (James et al., 2013). The Root Mean Squared Error (RMSE) is a good measure of accuracy when comparing prediction errors between different models. The RMSE is calculated as follows;

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\widehat{r}_i - r_i)^2} \tag{2}$$

From the equation above, $\widehat{r}_i$ is the predicted return for observation i, r_i is the observed return of observation i, and n is the total number of observations.

5 Methodology

This section outlines our approach to predicting corporate bond returns in the US bond market. We present our definition of corporate bond return and the independent variables used for predicting said return. Finally, we explain the specifications of the machine learning models and the trading mechanism used to construct the portfolios.

5.1 Dependent variable

The dependent variable is the monthly corporate bond returns calculated from end-of-month observations. However, some bonds are not traded on the last possible trading day each month, and to balance this out, we accept the price if it is observed at most three trading days before. This may result in multiple price observations for a particular bond on a given day. The aforementioned is because a bond can be traded multiple times intra-day. However, Friewald et al. (2012) believe that yield information is reflected more strongly in large trades, and therefore we will use volume-weighted returns to account for this. Additionally, some bonds are not traded every month, and for these bonds, we calculate the return for the holding period. We have used the following formula to calculate the returns;

$$r_t = \frac{(P_t+AI_t)+C_t-(P_{t-1}+AI_{t-1})}{P_{t-1}+AI_{t-1}} \tag{3}$$

P_t is the price in period t, AI_t is accrued interest in period t, and C_t is the coupon amount paid in period t. However, if the bond is not traded every month, and the return for the holding period is estimated, P_t refers to the price today, P_{t-1} is the last observable price, and C_t includes all coupon-payments between t and $t-1$.

5.2 Independent variables

In the machine learning models, we use credit ratings from Standard & Poor's to control for default risk, the Amihud measure to control for illiquidity, and market leverage as a proxy for a firm's creditworthiness. The Amihud illiquidity measure focuses on the price impacts of trades. In other words, if a bond can be traded with a large volume without any significant impact on its price, the liquidity is high, and vice versa (Amihud, 2002). The Amihud measure is estimated as:

$$ILLIQ_{it} = \frac{1}{DAYS_{it}} \sum_{i=1}^{DAYS_{it}} \frac{|r_{ijt}|}{Vol_{ijt}} \tag{4}$$

From the equation above, $|r_{ijt}|$ is the absolute return for bond i on the day j of month t. $DAYS_{it}$ tell us how many days bond i has been traded during month t, and Vol_{ijt} is the dollar volume. However, to avoid including the realized returns in the models, we use the lagged Amihud; $ILLIQ_{it-1}$.

Since the Amihud measure is additionally used as a market factor in the Fama-French regression, we expand equation (1) with a market-wide illiquidity measure constructed by aggregating individual measures over the entire sample quarter by quarter;

$$ILLIQ_{Market} = \frac{1}{N_t} \sum_{i=1}^{N_t} ILLIQ_{it} \tag{5}$$

Next, we define market leverage as the firm's total debt over the sum of total debt and the market value of equity. The market value of equity is calculated as the number of outstanding shares times the share price. Additionally, we use cash and working capital measures as controls for financial flexibility and firm value to control size risk. Lastly, we will include lagged returns. We believe that, given that bond returns are pre-specified, next month's return should equal its lagged value, given everything else equal.

We will present a table below that summarizes all variables used, the estimation, and its source.

Table 3: Variable explanation

This table presents our estimated and given variables. Additionally, it shows the mathematical notation, the estimation, and the source. Explaining the sources written as abbreviations, FISD is the Mergent Fixed Income database. FED is the Federal Reserve Bank. TRACE is the Trade Reporting and Compliance Engine. LT is short for "long-term." The Fama factors are downloaded from: http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/, thereby we specify Dartmouth in the table as a source.

Parameter	Sign	Estimation	Source
Control Variables			
Amihud	$ILLIQ_{t-1}$	Sum of absolute mean return divided by average dollar volume	FISD/TRACE
Credit ratings	$RATING$	Given	Standard and Poor's
Lagged return	R_{t-1}	$lag(R, 1)$	TRACE/FISD
Firm value	FV	Total debt plus equity	Compustat
Market leverage	ML	Total debt divided by sum of total debt and market equity	Compustat
Cash	CHQ	Given	Compustat
Working capital	WC	Current assets minus the short-term debt	Compustat
Fama factors and Amihud			
Market premium	MKT	Given	Dartmouth
High minus low book to market	HML	Given	Dartmouth
Small minus big (size)	SMB	Given	Dartmouth
Term premium	$TERM$	LT (30y) government bond minus 1M T-bill	Dartmouth/FED
Default premium	DEF	Investment grade portfolio minus LT (30y) government bond	FED/TRACE/Standard and Poor's
Market-wide Amihud	$ILLIQ$	Quarterly-aggregated $ILLIQ_t$	FISD/TRACE

5.3 Period of interest

We are interested in how the importance of the independent variables differs during the financial crisis and Covid-19 compared to normal market conditions. Therefore, we monitor two sub-periods: Financial crisis 2007-2009 and Covid-19 from the 11th of March 2020 until the 31st of March 2021. In the financial crisis, many bonds had overly-optimistic credit ratings, where the credit- and liquidity quality quickly deteriorated. Both liquidity- and credit quality are important determinants of bond prices, and it will be interesting to monitor both during and post the financial crisis. Next, the World Health Organisation declared the Covid-19 outbreak a pandemic on the 11th of March 2020, which justifies the starting date of this sub-period. This is of interest due to the declining interest rates observed, where the FED cuts lowered the rate to as low as 0%.

5.4 Machine learning methods

When predicting future returns, a longer horizon increases the uncertainty of the results. Leippold et al. (2021) emphasize that model predictability is higher at short horizons, and to combat this, we utilize a more sophisticated approach of modeling called time-series cross-validation. Therefore, since we are rebalancing quarterly, we predict only three months at a time, starting with the first quarter of 2008. As displayed in Figure 4, our train set will expand with three months for each model, allowing the machine learning algorithm to have the maximum available information when predicting the upcoming three months. To ensure that the first model has sufficient data, the initial training set will cover the time period from the second quarter of 2002 until the end of 2007. As a result, we will fit 53 models per machine learning method used.

5.4.1 XGBoost

XGBoost is a decision-tree-based ensemble machine learning algorithm utilizing boosting. Unlike other boosting methods, XGBoost uses regularization extensively. The model will add penalty terms to avoid overfitting by setting the coefficients of unexplanatory independent variables toward zero, which improves generalization capabilities (James et al., 2013). In other words, the model selects its predictors, discarding the need for variable selection. However, García et al. (2016) emphasize that removing irrelevant predictors to obtain a subset that appropriately correlates with the dependent variable increases the models' ability to generalize. By selecting fewer variables, we also reduce the search space for the model, making the process of fitting our models faster and less memory-consuming. Therefore, we include only the variables mentioned in section 5.2. The same will apply to the Random Forest model.

For the model to be precise, it tunes a set of hyperparameters. The optimal value for each parameter is different for each data set. However, XGBoost does this automatically if we provide the model with an extensive list of different values for each parameter, allowing the parameters to adjust to more data being added. One parameter of noticeable impact is the number of rounds. This parameter specifies how many sequences the boosting process is repeated, as visualized in Figure 3. Additionally, this parameter is the most computationally demanding. Thus we do not provide a list of different values, thereby specifying the number of rounds to 50.

5.4.2 Random forest

Random Forest is a decision-tree-based ensemble machine learning algorithm utilizing bagging. The random forest builds n amount of trees using a unique bootstrapped sample from the data for each tree. When the trees in a random forest are split, only a random sample is considered from the independent variables specified in the model. However, only one of the considered variables is used to split the data. In other words, the random sample of predictors is different every time a split is made. Further, each tree has a high variance isolated, thus, bagged together to create a final prediction (James et al., 2013).

The main strength of using this approach for variable selection is that the trees become highly uncorrelated. Without the random subsetting, one strong predictor among many weak predictors would make most trees use this predictor in the first split making the trees highly correlated, decreasing the gain from fitting multiple trees. The gain in variance reduction is substantial when averaging uncorrelated quantities, which is why Random Forest is preferred over simply bagging trees.

Although bagging increases model accuracy, it is at the expense of the interpretability of the model. For instance, it is challenging to display how a model of multiple trees works because every tree will be unique, making it unclear which variable is key to making good predictions. However, we can obtain an overall summary of importance based on each variable's contribution to reducing the Residual Sum of Squares (RSS).

When tuning the hyperparameters, we will consider the number of predictors the model will be allowed to choose from at each split and the number of decision trees to build. The prediction subset allowed at each split is commonly set to the square root of the total amount of predictors. We have included seven predictors as outlined in section 5.2. Therefore, we round up the squared number of predictors and allow the model to consider three randomly chosen at each split. We decided how many trees to grow based on the marginal decrease in the prediction error from growing one more tree. Growing many trees is also computationally expensive, so to prevent the model from taking a long time to fit, we found the suitable value to be 50 trees. A plot of the marginal error reduction for fitting one additional tree is supplied in the appendix; Figure A1, emphasizing that 50 trees are sufficient.

5.5 Trading mechanism

Our portfolio will be quarterly rebalanced, allowing for short and long positions. The rebalancing takes place on each quarter's last possible trading day, selling all positions held, and placing new positions for the upcoming quarter. In other words, a buy-and-hold strategy. To be able to fulfill this strategy, a set of assumptions are made;

1. Bonds must be traded at the end of two consecutive quarters.
2. All bonds can be shorted.
3. All returns are reinvested.
4. Equal weights in all positions.
5. The investor has the capital required to fulfill all quarterly positions.

To be able to rebalance all positions quarterly, all bonds we take a position in must have an observed transaction at the end of the next quarter. This makes our strategy forward-looking, consequently introducing liquidity bias by not allowing us to buy less frequently traded bonds. Additionally, if firms have several outstanding bonds, a portfolio with equal weights will be overweighted in those firms. Since we have equal weights in our portfolios, such overweights will increase idiosyncratic risk.

When constructing the portfolios, we calculate the quarterly return for each bond based on the monthly predicted returns using the following formula;

$$\widehat{r}_{quarterly} = \left(\prod_{t \epsilon month} (1 + \widehat{r}_t) \right) - 1 \tag{6}$$

Then, using the quarterly returns, $\widehat{r}_{quarterly}$, we select bonds for our long-short portfolio based on the following;

1. A long position will be taken in the top 5% performing bonds.
2. A long position will not be taken if $\widehat{r}_{quarterly} < 0$.
3. A short position will be taken in the 5% worst performing bonds.
4. A short position will not be taken if $\widehat{r}_{quarterly} > 0$.

This implies that if there are no bonds with a $\widehat{r}_{quarterly} < 0$, no short positions will be taken, and vice versa, conceivably leading to quarters with no short or long positions. In other words, this is not a zero-investment strategy where short and long positions have to null each other out. If we were to create a self-financing strategy, we would potentially have to force short positions in bonds predicted to have $\widehat{r}_{quarterly} > 0$ during time with conservative predictions. Finally, we will create a value-weighted market portfolio as a benchmark to compare the results.

6 Results

This section will compare the two models by assessing model accuracy, variable importance, portfolio positions, and the predicted return series. Furthermore, we will test for significant alpha to determine whether the models can construct portfolios that generate returns in excess of the market portfolio. We present our results without accounting for transaction costs due to inequalities between institutional and retail investors. However, we will discuss the effects regarding transaction costs and supply results for both scenarios in the appendix.

6.1 Model accuracy

To be able to see the development of the prediction accuracy, we will plot the RMSE for all models, visualizing it over the entire prediction time period;

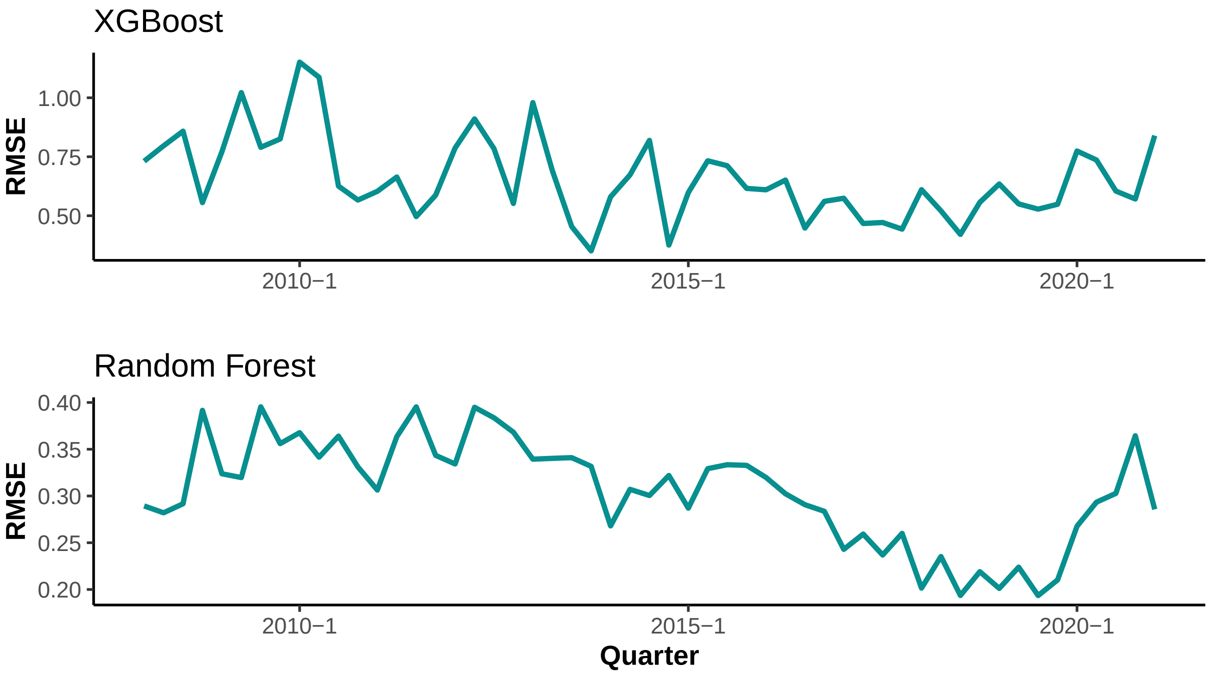

Figure 5: Model performance
This figure displays the Root Mean Squared Error (RMSE) of all 53 models for both XGBoost and Random Forest. One quarter indicates one model.

From the figure above, we observe that the variance of the RMSE for both methods is declining up until 2019. However, in the last few years, RMSE seems to show evidence of an upward trend in coherence to the Covid-19 outbreak. A more unstable RMSE tied to the years of the financial crisis and the Covid-19 outbreak is expected due to high default risk and drastic interest rate cuts. This signalizes that externalities such as recessions and pandemics are problematic for a model to capture. Another explanation could be the amount of data included in the training sets. We can observe a more unstable RMSE when the models only have a few years to be trained on compared to the end of our time period with over 15 years of data.

The main difference between the two methods is that the prediction accuracy for the Random Forest is less volatile, given a more narrow range interval of the RMSE values. Additionally, the prediction errors are generally lower than XGBoost given the lower scale, signaling more accurate predictions. Next, to better understand how the models work, we assess the importance of the model's independent variables.

6.2 Variable importance

We will plot an importance score over time to better understand how the models weigh and use their control variables. The importance score for each variable is dependent on its contribution to reducing the RSS. Visualizing this allows us to see if some variables are more prominent in periods impacted by a financial crisis, pandemics, or under normal market conditions.

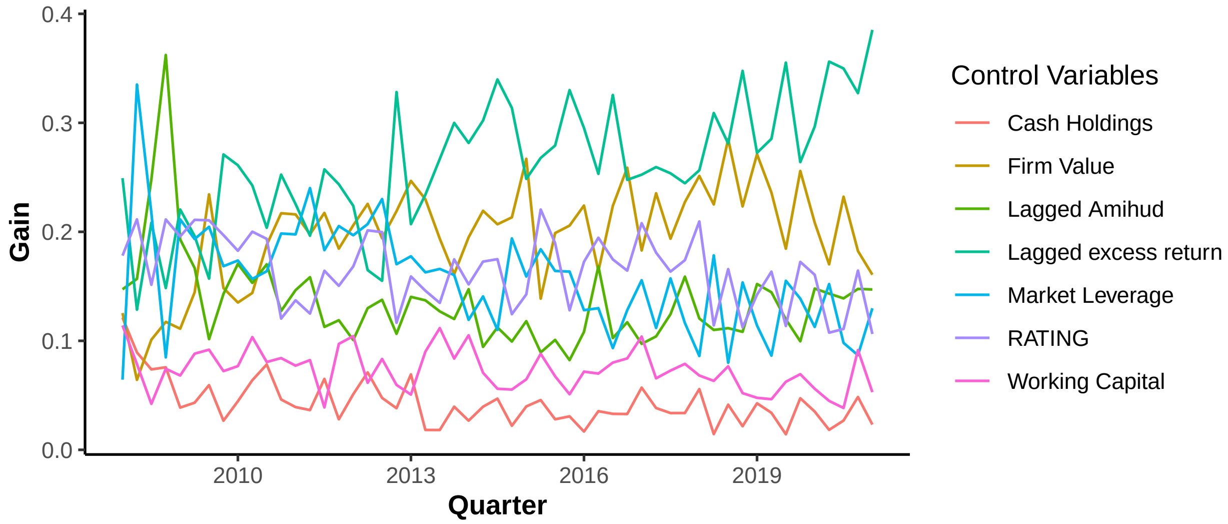

Figure 6: Variable importance XGB

The importance scores are measured by the variables' contribution to a loss in Residual Sum of Squares as a consequence of a decision made by a given variable. The higher value, the more influential the variable is in the decisive process. One quarter refers to one model.

Figure 6 shows which variables contribute the most to XGBoost's performance. During the financial crisis the credit- and liquidity quality quickly deteriorated. This emphasizes the importance of market leverage, which proxies for creditworthiness, and the lagged illiquidity measure in those years.

Further, it shows that all models after 2015 heavily rely on the lagged excess return. Figure 1 shows an increase in transaction amounts after 2015, which may indicate that lagged excess return becomes more effective when more data is included. On the other hand, lagged excess return should, conceivably, be more influential in times of low volatility. Thus more prominent after the financial crisis. However, the high importance in recent volatile years does not support this.

Lastly, firm value shows consistent importance throughout the entire time period. We interpret this as coupons varying due to risk measured by firm size being relatively influential throughout the entire time period. Similarily to the firm value, credit ratings seem to have stable importance. Its importance is likely because credit ratings indicate the quality and stability of bonds. The remaining variables, cash holdings and working capital, have low overall importance. The same assessment will be done for the Random Forest method.

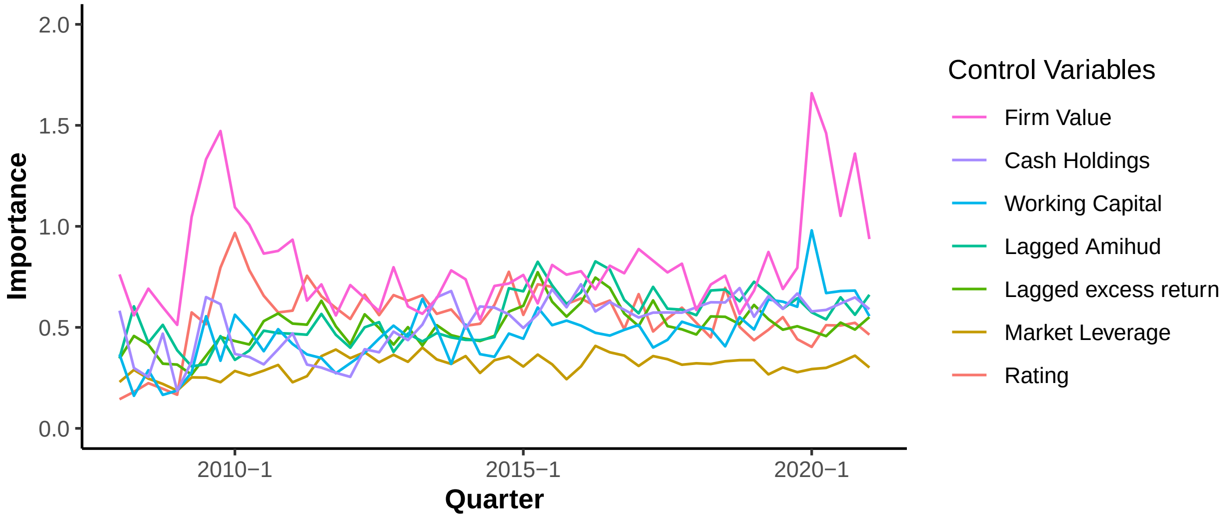

Figure 7: Variable importance Random Forest

The importance scores are measured by the variables' contribution to a loss in Residual Sum of Squares as a consequence of a decision made by a given variable. The higher value, the more influential the variable is in the decisive process. One quarter refers to one model.

Figure 7 shows that the firm value measure scores the highest out of all the predictors, indicating that the firm's size is essential in determining how much excess return an investor can expect. Additionally, the importance of firm value is spiky, with one peak tied to the financial crisis and one to Covid-19, which indicates that this variable is particularly helpful in splitting the data for these periods. This may signalize that during periods of economic distress, big firms are more likely to fulfill their obligations to the bondholders in the event of default. In contrast to the XGboost, the Random Forest does not find lagged excess return as influential and seems to have similar gains in reducing RSS compared to the remaining variables. Next, we will visualize how the trading mechanism outlined in section 5.5 impacts the distribution of long and short positions.

6.3 Portfolio positions

The assumptions and criteria outlined in section 5.5 will impact the distribution between the number of long and short positions. Below we visualize the portfolio positions derived from both methods' predictions.

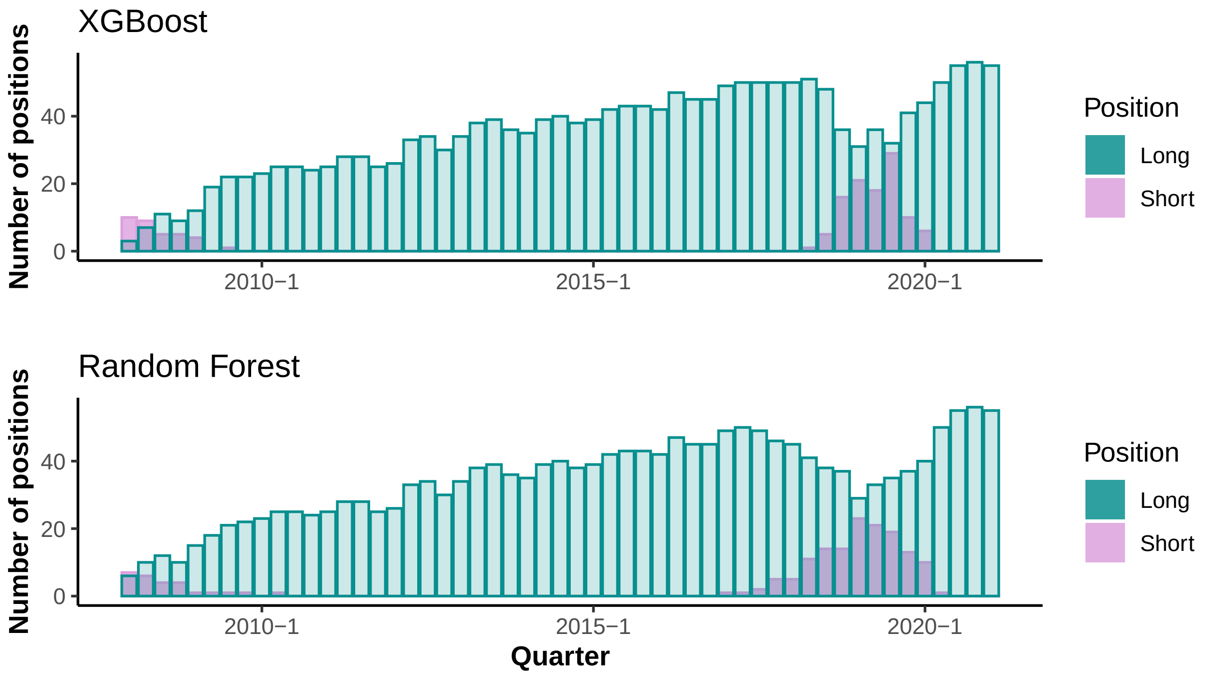

Figure 8: Portfolio positions

A long position means buying the bond at the start of the current quarter and selling at the end. Shorting means borrowing the bond, then selling it before repurchasing it at the end of the quarter. One quarter refers to one model. Lastly, number of positions refers to the number of unique bonds we have bought or short-selled.

Figure 8 shows the distribution between the portfolio's long and short positions for each quarter. As expected, the number of positions in total increased during the time period and can be seen as a natural outcome of more transactions in recent years, also illustrated in Figure 1. More interestingly, the number of short positions exceeded the number of long positions during the beginning of the financial crisis in 2008, indicating that there were more bonds predicted to provide negative returns. During the financial crisis, many bonds had overly-optimistic credit ratings, misleading investors to buy "safe" bonds. Consequently, many high-quality bonds declined a lot, which might explain why our models predict such high ratios of short positions in those years.

Prior to the drastic interest rate cuts observed in 2020, the rates were approximately 2.5%, the highest since the financial crisis. The interest rates were expected to keep increasing, which may be the reason for the high short exposure before the entry of Covid-19. Moreover, the number of long positions appreciated to new heights during 2020. The bullish bond market arising from the interest rate cuts can somewhat explain the high number of long positions because bond prices appreciate when the interest rates drop.

Additionally, several quarters only include long positions. This means that the models do not predict any bonds to have a negative return. Given our constraint to not short any bonds with $\widehat{r}_{quarterly} > 0$, multiple quarters have zero short positions. This indicates that our models, in general, are more conservative in predicting negative returns.

Lastly, the main difference between the two methods is that the Random Forest portfolio has longer periods of short positions during times of uncertainty. This indicates that the Random Forest has more periods where it predicts bonds with negative returns resulting in a broader distribution of short positions. Conversely, the XGBoost model predicts fewer periods with a negative return, thus a more narrow distribution of periods going short. However, when initially predicting a negative return, the ratio of short positions is higher.

6.4 Return series

When the positions are made, we calculate the portfolios equally weighted predicted- and realized return each quarter throughout the entire time period from January 2008 until the end of March 2021.

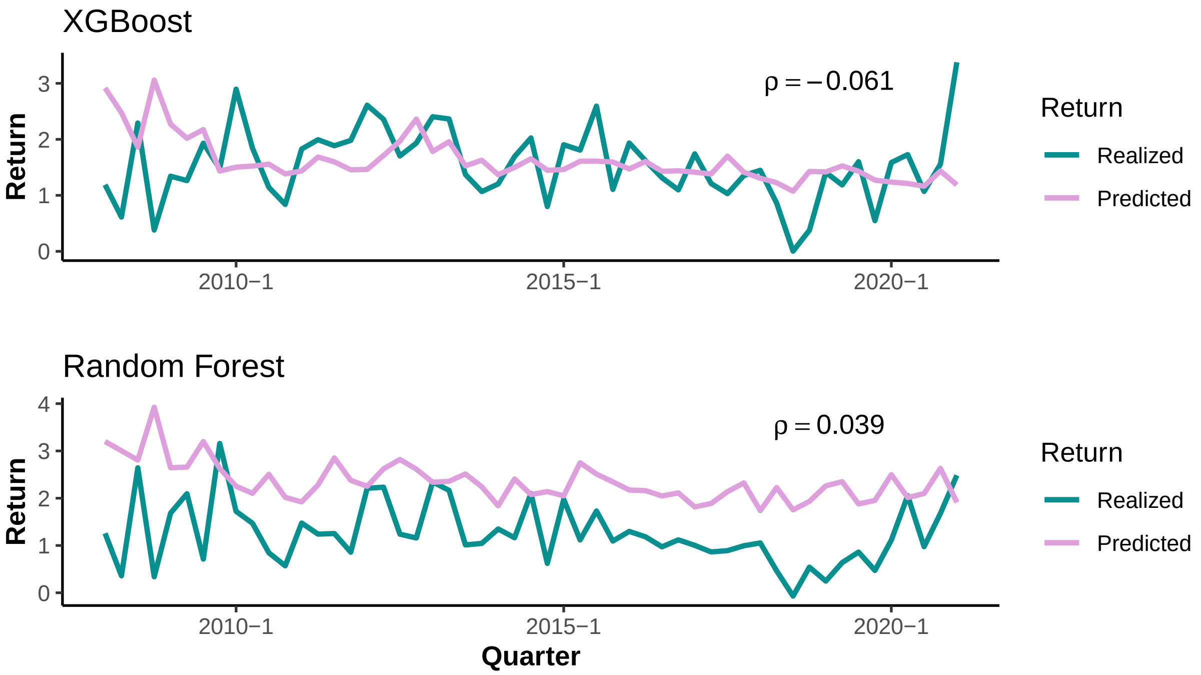

Figure 9: Portfolio performance

The realized return is the portfolio's actual return, whereas the predicted returns are the model's guess. All returns are outcomes from an equally weighted portfolio. One quarter refers to one model. ρ is the correlation between the predicted- and realized return for the entire period.

Figure 9 indicates that all models in both methods struggle to capture the variance in the excess returns, especially during times of high volatility. The models' lack of capability to capture the variance is also emphasized by low correlation and corresponding R-squared values consistent around 2-4% for both methods.

For XGBoost, the models' predictions are rarely extreme past 2012 and seem to stabilize at 1.5%, which in most periods seems reasonable. On the other hand, this could potentially lead to negative realized returns when volatility increases, as seen in 2018. Unlike XGboost, which seemingly finds a reasonable value for its predictions that are neither too high nor too low, the Random Forest is consistently overpredicting. Nevertheless, the overall model has lower RMSE values than XGboost, outlined in Figure 5, and it evidently predicts specific periods quite accurately, particularly in the period from 2012 to 2015. However, contradictory to the RMSE, the portfolio predictions are more inaccurate than the realized returns during the period with lower corresponding RMSE. Because our portfolios consist solely of the most extreme bonds, we may interpret this as the Random Forest not being as adequate compared to XGBoost at predicting the most volatile bonds.

Moreover, there is no quarter where our portfolios provide negative returns. However, given a consistent gap between realized and predicted returns, we can not entirely credit that all quarters provide a positive return to the models' ability to predict returns accurately. We believe that our trading strategy limits the risk of negative realized returns by only selecting the most extreme predictions. In other words, implementing this strategy in combination with the models' predictions provides a low probability of negative realized returns.

6.5 Alpha

So far, we can not conclude that either the XGBoost or the Random Forest portfolio of bonds provides a consistent return in excess of the market. Therefore, we run a Fama and French regression as outlined in section 4.1, including Amihud, on the realized return series of both models to check whether the intercept term (alpha) is significant. If we observe a significant alpha, the models find relations between the control variables and excess return that are not entirely explained by the Fama-factors or Amihud.

Fama and French (1993) proved that the stock-market factors alone explain some of the variation in bond returns. However, when the two term-structure factors are included, the explanatory power of the stock-market factors disappears for all but the low-grade corporate bonds. The table below emphasizes this, providing both term-structure factors a significance of 1%, where the stock-market factors are insignificant for the market portfolio. Additionally, as expected, the alpha term is indistinguishable from zero.

Table 4: Fama-regression

	Dependent variable: Excess return		
	XGBoost	Random Forest	Market
	(1)	(2)	(3)
MKT	−0.005	−0.025	−0.012
	(0.018)	(0.017)	(0.008)
SMB	0.046	0.022	−0.004
	(0.036)	(0.034)	(0.016)
HML	0.014	−0.004	−0.004
	(0.030)	(0.028)	(0.009)
TERM	0.455	1.138^{***}	1.625^{***}
	(0.342)	(0.316)	(0.236)
DEF	0.594^{*}	1.234^{***}	1.507^{***}
	(0.327)	(0.302)	(0.235)
Amihud	−0.275	−0.146	0.023
	(0.195)	(0.180)	(0.118)
Alpha	2.111^{***}	1.446^{***}	0.430
	(0.542)	(0.501)	(0.328)
Observations	53	53	53
Adjusted R^2	0.068	0.245	0.525
Note:		*p<0.1; **p<0.05; ***p<0.01	

Similar to the market results, both the TERM and DEF factors are significant at 1% for the Random Forest method, indicating that those market factors can somewhat explain the excessive returns generated from this portfolio. However, since the adjusted R-squared is only at 24.5%, and the alpha is significant at 1%, there is still unexplained information left in the variance of the excess return. Furthermore, XGBoost provides similar results regarding a significant alpha. However, TERM is not significant, and DEF is only significant at 10%, thus explaining less of the variance in the excess return, which is also emphasized by the lower adjusted R-squared of 6.8%.

However, these results do not account for transaction costs. According to Leippold et al. (2021), the inclusion of transaction costs is essential to determine the economic significance. According to Harris (2015), transaction costs on institutional trades (trades > 100,000\$) are on average 52.1 basis points and 84 basis points for Retail trades (trades < 100,000\$). In other words, our alpha will be reduced by 0.521% if implemented by institutional investors. This impacts the alpha generated by the Random Forest, reducing its significance to 10%, while XGBoost's alpha remains unaffected. Moreover, if implemented by Retail investors, alpha becomes insignificant for Random Forest and reduced to 5% significance for XGBoost. Given XGBoost's robust results, this reassures the institutional investors with a quite high margin on the transaction costs before alpha becomes less significant. Both scenarios are visualized in the appendix: Table A1 & A2.

These results imply that using machine learning methods, specifically decision trees utilizing boosting, to predict bond returns brings better measurements of complex relationships, evidently generating significant alphas. Further, this disproves the EMH and indicates that the bond prices do not reflect all information.

Finally, since generating alpha on behalf of taking upon enormous risk is not desirable, we monitor the portfolios' risk-adjusted return over time. However, each investor has a different risk aversion, meaning that investors would pick a different portfolio based on their risk preference. Nevertheless, a portfolio that maximizes the Sharpe ratio would be preferred for all investors if the assumption about lending or borrowing the risk-free asset is viable. Although all investors would pick the same portfolio, they would weigh the risky portfolio and the risk-free asset differently.

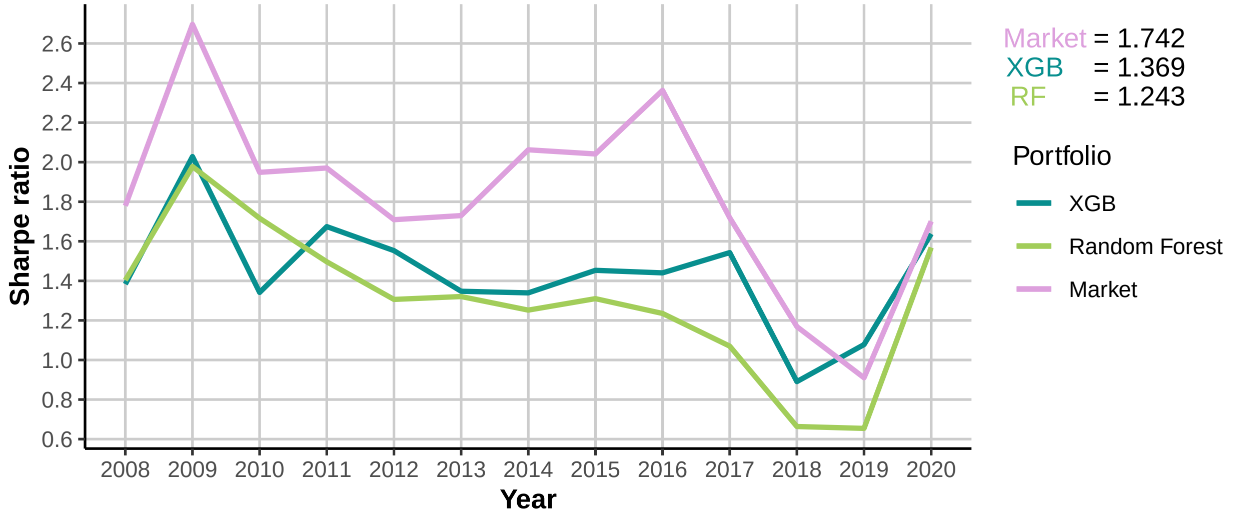

Figure 10: Sharpe ratio

This table displays the yearly Sharpe ratio for all portfolios. Sharpe ratio is calculated as the return minus the risk-free rate divided by the standard deviation. The Sharpe ratio indicates how much return the investor will receive per unit of risk taken. In the top right corner of the figure, the mean Sharpe ratios for the entire time period is provided.

It is important to note that since our strategy is to buy and hold the portfolio with quarterly rebalancing for 13 years, the Sharpe ratio for the entire period is the one that is proposed to the investor. However, yearly changes are interesting to monitor due to their visualization of the portfolios' performance during the financial crisis and the Covid-19 outbreak.

The figure shows that the market portfolio offers the investor a Sharpe ratio of 1.742 over the entire period for the strategy, which is considered very good. The market portfolio's Sharpe ratio may be artificially high given our sample, outlined in section 3. Although the portfolios derived from the machine learning result in a significant alpha, evidence through lower Sharpe ratios states that the alpha generated results from taking more risk. This implies that the portfolios bet on some risky metrics not entirely captured by the factors controlled for in table 4.

Based on the Sharpe ratios, all investors would prefer the market portfolio, given the assumption of borrowing/lending the risk-free asset. Thus the optimal strategy for any investors would be to adjust the weights between the market portfolio and the risk-free asset, given their level of risk-aversion. However, that individual investors can borrow/lend money at a risk-free rate is unrealistic. Thus, it is more realistic to believe that some investors, based on risk-aversion, would prefer the XGBoost portfolio, given its excess returns.

The yearly Sharpe ratios contain high variation, and the market portfolio displays four periods of $Sharpe > 2$, being 2009, 2014, 2015, and 2016. In 2009, the bond market experienced extreme returns, conceivably due to interest rate cuts as a ripple effect of the financial crisis. Thus the market had to experience a relatively high increase in return compared to risk, consequently providing a high Sharpe ratio that year. Low-interest rates were observed until 2016 when the FED started to increase the interest rate from 0.25% to 2.5% in 2019. This increase negatively impacts the relative value of discounted cash flows, consequently reducing expected returns while increasing risk, thus providing lower Sharpe ratios. Finally, during the outbreak of Covid-19, the Sharpe ratios are recovering to higher values due to drastic interest rate cuts, emphasizing the interest rate's effect on bond returns.

Like Leippold et al. (2021), we find that portfolios derived from machine learning still provide adequate Sharpe ratios, even with transaction costs as high as 84 basis points. Both portfolios have *Sharpe* > 1 given the two scenarios of transaction costs, as outlined in the appendix: Table A3, signaling acceptable performance for the investors compared to the performance of the risk-free investment.

Next, to provide the investor information about how much return the portfolios would generate throughout the period, we visualize the cumulative return.

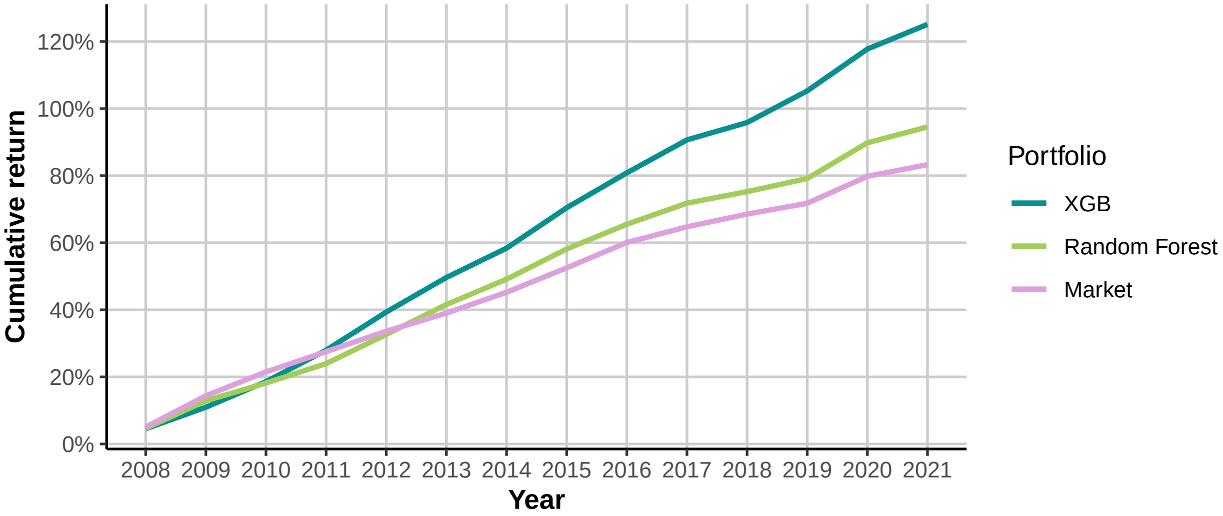

Figure 11: Cumulative return

The cumulative return displays the total return from investing in portfolio x in 2008, then reinvesting all returns every quarter until the first quarter of 2021.

Figure 11 shows that the market portfolio provides higher returns in the first three years. In other words, investors have to hold the XGBoost portfolio longer than three years to achieve cumulative returns above the market and more than five years with the Random Forest portfolio. Ultimately, XGBoost offers the highest cumulative return. However, if institutional investors implement this strategy, the Random Forest portfolio will generate a lower cumulative return than the market portfolio. Conversely, the XGBoost portfolio will still be able to generate a cumulative return of 30% excess of the market portfolio for institutional investors and ~20% for retail investors. This further emphasizes its significant alpha, indicating that the excess return generated by XGBoost is not entirely due to an assumption of no transaction costs. Both scenarios are visualized in the appendix; Figures A2 & A3.

7 Conclusion

We investigate two different machine learning methods' predictive power in the US corporate bond market. We find that the most influential variables are associated with size risk and historical return. What surprises us is that variables controlling for credit and liquidity risk are not as prominent after the financial crisis, especially during Covid-19. Further, the high predictability at short horizons combined with the investment strategy employed translates into highly significant alphas and superior cumulative returns for the long-short portfolios compared to the market. In particular, XGBoost provides a statistically significant performance even under the two scenarios of transaction costs. This strategy reassures Institutional investors with a high margin on the transaction costs before alpha becomes less significant.

Machine learning methods bring better measurements of complex relationships, evidently through generating return excess of the market. The overall success of XGBoost's return prediction signalizes an inefficient market, disproving the EMH. However, our discoveries indicate that a large portion of the excess return is explained by taking more risk. Nevertheless, no stock-market, term-structure, or liquidity factors could explain the said risk. Finally, our findings suggest that using machine learning methods, particularly decision trees that utilize boosting, to predict the corporate bond market brings promise for future portfolio choice and financial modeling.

8 Critique and further research

Even though our study finds a significant alpha, the assumptions made to create the portfolio likely bias our results. Moreover, our study is forward-looking, whereas, in the real world, we would not know if we could liquidate all positions each quarter, introducing liquidity bias. Additionally, we only have data on observed transactions. If a bond defaults and is never bought or sold again, we will not observe the negative return related to that bond. This means that our model will never be able to pick a bond and lose the investment. One possible solution would be to investigate which bonds default, thus setting their return equal to -100%.

Due to computational limitations, we had to reduce the number of observations in the data set for the feasibility of running the machine learning models. Consequently, we excluded numerous types of bonds embedding different types of characteristics mentioned in section 3. In other words, our results are likely not applicable to all types of corporate bonds.

Moreover, additional research could be done to investigate if using a rolling window of data to create the models could be more beneficial in predicting returns. The idea behind this is that old information might not be as relevant as market conditions and psychology might change. By discarding older information, one might better capture how the current market behaves.

Nevertheless, Table 4 shows that we have a significant alpha, and further research could be done to find a market factor that captures the information found by the machine learning models. Suppose it is possible to find what causes the alpha. In that case, one could better understand the underlying factors that generate excess return.

9 Bibliography

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

Bai, J., Bali, T. G., and Wen, Q. (2019). Common risk factors in the cross-section of corporate bond returns. *Journal of Financial Economics*, 131(3):619–642.

Chung, K. H., Wang, J., and Wu, C. (2019). Volatility and the cross-section of corporate bond returns. *Journal of Financial Economics*, 133(2):397–417.

Dick-Nielsen, J. (2009). Liquidity Biases in TRACE. *The Journal of Fixed Income*, 19(2):43–55.

Edwards, A. K., Harris, L., and Piwowar, M. S. (2004). Corporate Bond Market Transparency and Transaction Costs. *SSRN Electronic Journal*.

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.

Friewald, N., Jankowitsch, R., and Subrahmanyam, M. G. (2012). Illiquidity or credit deterioration: A study of liquidity in the US corporate bond market during financial crises. *Journal of Financial Economics*, 105(1):18–36.

García, S., Ramírez-Gallego, S., Luengo, J., Benítez, J. M., and Herrera, F. (2016). Big data preprocessing: methods and prospects. *Big Data Analytics*, 1(9):0.

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

Harris, L. (2015). Transaction Costs, Trade Throughs, and Riskless Principal Trading in Corporate Bond Markets. *SSRN Electronic Journal*.

Hyndman, R. J. and Athanasopoulos, G. (2021). *Forecasting: principles and practice*.

James, G., Witten, D., Hastie, T., and Tibshirani, R., editors (2013). *An introduction to statistical learning: with applications in R*. Number 103 in Springer texts in statistics. Springer, New York.

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

Lin, H., Wang, J., and Wu, C. (2011). Liquidity risk and expected corporate bond returns. *Journal of Financial Economics*, 99(3):628–650.

Rouf, N., Malik, M. B., Arif, T., Sharma, S., Singh, S., Aich, S., and Kim, H.-C. (2021). Stock Market Prediction Using Machine Learning Techniques: A Decade Survey on Methodologies, Recent Developments, and Future Directions. *Electronics*, 10(21):2717.

SEC (1997). Report to the Congress: Impact of Technology on Securities Markets. Technical report, U.S. Securities and Exchange Commission.

10 Appendix

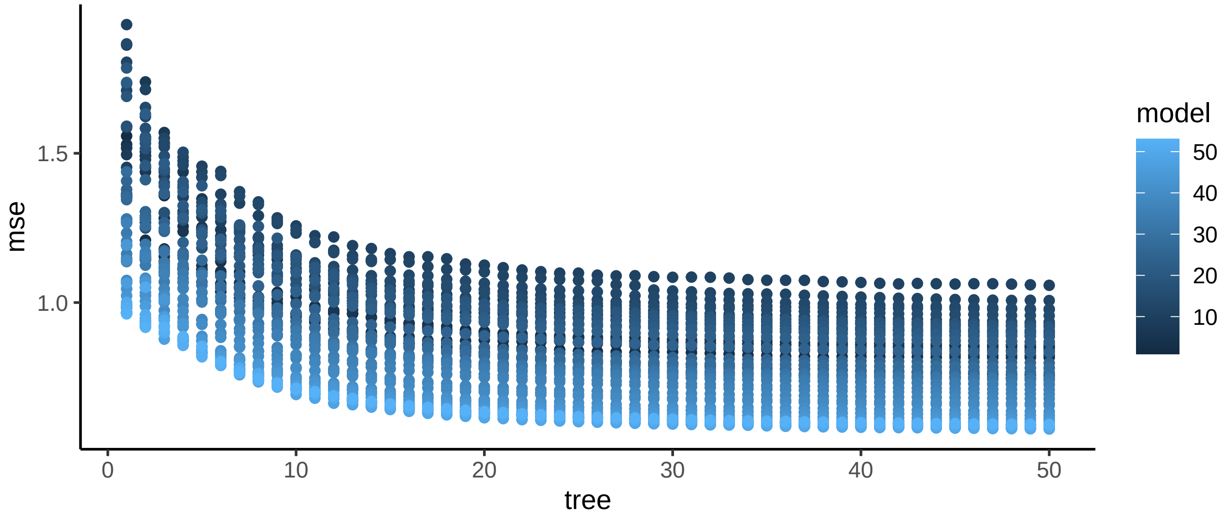

Figure A1: Marginal decrease in MSE

Table A1: Fama-regression for Institutional investors

	Dependent variable: Excess return - transaction cost		
	XGBoost	Random Forest	Market
	(1)	(2)	(3)
MKT	−0.005	−0.025	−0.012
	(0.018)	(0.017)	(0.008)
SMB	0.046	0.022	−0.004
	(0.036)	(0.034)	(0.016)
HML	0.014	−0.004	−0.004
	(0.030)	(0.028)	(0.009)
TERM	0.455	1.138***	1.625***
	(0.342)	(0.316)	(0.236)
DEF	0.594*	1.234***	1.507***
	(0.327)	(0.302)	(0.235)
Amihud	−0.275	−0.146	0.023
	(0.195)	(0.180)	(0.118)
Alpha	1.590***	0.925*	0.430
	(0.542)	(0.501)	(0.328)
Observations	53	53	53
Adjusted R^2	0.068	0.245	0.525

Note: *p<0.1; **p<0.05; ***p<0.01

Table A2: Fama-regression for Retail investors

	Dependent variable: Excess return - transaction cost		
	XGBoost	Random Forest	Market
	(1)	(2)	(3)
MKT	−0.005	−0.025	−0.012
	(0.018)	(0.017)	(0.008)
SMB	0.046	0.022	−0.004
	(0.036)	(0.034)	(0.016)
HML	0.014	−0.004	−0.004
	(0.030)	(0.028)	(0.009)
TERM	0.455	1.138***	1.625***
	(0.342)	(0.316)	(0.236)
DEF	0.594*	1.234***	1.507***
	(0.327)	(0.302)	(0.235)
Amihud	−0.275	−0.146	0.023
	(0.195)	(0.180)	(0.118)
Alpha	1.271**	0.606	0.430
	(0.542)	(0.501)	(0.328)
Observations	53	53	53
Adjusted R^2	0.068	0.245	0.525
Note:			*p<0.1; **p<0.05; ***p<0.01

Table A3: Sharpe ratio scenarios

This table presents the Sharpe ratio for all scenarios with and without transaction cost.

Portfolio	**0 bp**	**52.1 bp**	**84 bp**
XGBoost	1.369	1.247	1.172
Random Forest	1.243	1.106	1.023

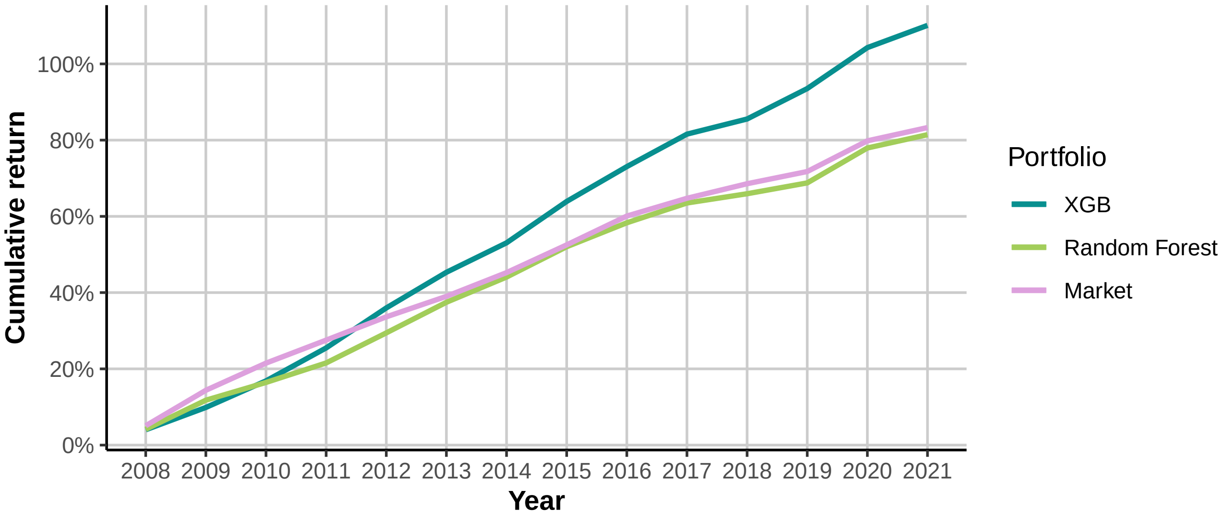

Figure A2: Cumulative return for Institutional investors

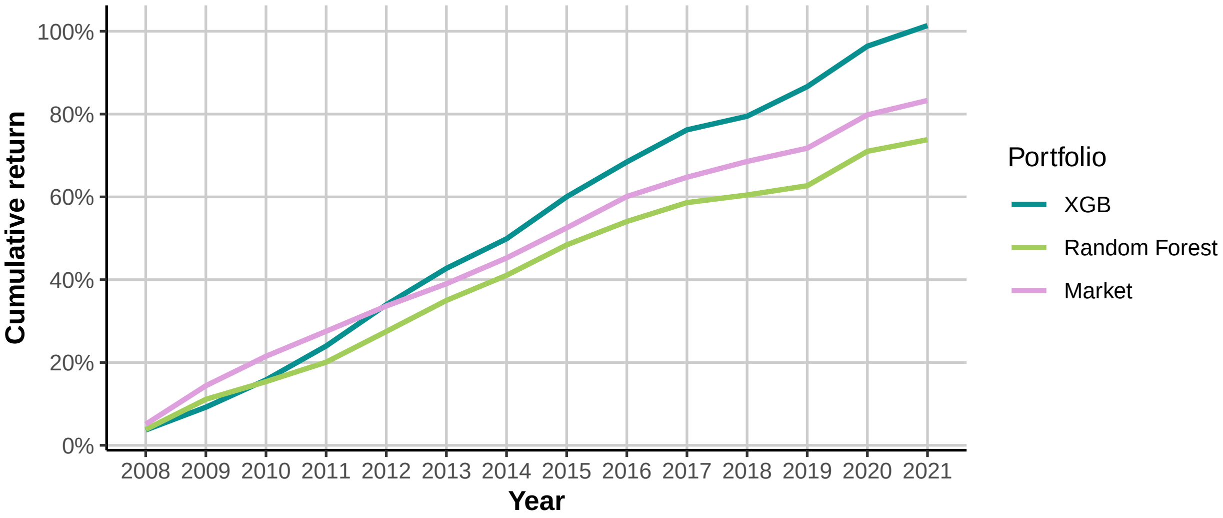

Figure A3: Cumulative return for Retail investors