Predicting Earnings Management using XBRL though Machine Learning Models

Master Thesis in Business Analytics and Management

Rotterdam School of Management
Erasmus University Rotterdam

Apoorv Sunny Bhatia
590913

Supervisors:
Coach: dr. Iuliana Sandu
Co-reader: dr. Simon Zehnder

Date: May 24, 2023

Preface

Executive summary

Earnings Management (EM) refers to earnings-manipulative practices that are within the general accepted accounting principles that can mislead investors, regulators, and other stakeholders. Its detection and prevention is important to ensure that financial statement users can make their decisions with correct

This study will evaluate several machine learning models to analyze whether financial statement information can be used to predict earnings management. This will be achieved using a fairly new approach in this field to gather the financial statements, which is the use of eXtensible Business Reporting Language (XBRL). This language allows stakeholders to process, analyze, and validate financial statement with greater speed and accuracy than using databases.

This study found that..

As this study focused on an investor's point of view, it attempted to create a study that can be replicated by investores. Therefore, all the code that is used in this study can be found via the following GitHub repository: `https://github.com/sunny1999123/MasterThesis`

Contents

List of Tables

List of Figures

1 Introduction

1.1 Problem Background

All publicly listed companies are obliged to publish financial statements that provide an accurate representation of their financial status. Nevertheless, there have been instances where firms lacked in doing so. Enron, a large U.S. energy company, deliberately submitted inaccurate financial statements which even led to its bankruptcy (Campa, 2019). This was mainly due to the fact that Enron engaged in earnings management (EM).

EM refers to the manipulation of financial statements using accounting practices in order to achieve desired outcomes (Beneish, 2001). Firms' management may decide to engage in EM for various reasons, such as meeting earning targets or influencing the share price. However, its primary objective is to deceive shareholders regarding the company's financial situation (Almahrog and Lasyoud, 2021). While EM and Fraud have a common goal, its main difference is that EM is within the bound of the General Accepted Accounting Principles (GAAP), whereas fraud is not (Perols and Lougee, 2011). Since investors use financial statements in their investment choices, EM could lead to incorrect investment decisions due to the misleading financial reports (Chen et al., 2019).

According to Dbouk (2017), the current state-of-the-art regarding the detection of EM, which includes manual verification of general ledger accounts, the use of simple ratios, and inventory counts, are imprecise and outdated. The recent technological development in business demands the use of advanced technology for EM detection (Sanad, 2021). In recent years, several studies attempted to incorporate machine learning techniques in the detection of EM. Chen, Chi, and Wang (2015) studied whether a set of 25 financial statement items could predict the discretionary accruals of a company, which are used as a proxy for EM. A more recent research, conducted by Hammami and Zadeh (2022), used various feature selection techniques and combined this with a Support Vector Machine to assess its performance in predicting EM. Still, Almaqtari et al. (2021) argue that there is limited research on predicting EM using machine learning, whilst the field seems promising.

One of the components in the field that is yet to be explored when it comes to EM prediction is how investors could assess EM themselves. As mentioned before, Chen et al. (2019) argue that investors rely their investment decisions on financial statements. Therefore, they should have reasonable assurance that financial statements represent a firm's actual performance. Even though this responsibility belongs to the auditor, investors can have personal risk preferences and should be able to do their research. Therefore, this study will focus on the perspective of investors.

A drawback of the current research on the use of machine learning in EM detection is that it is conducted using databases (Hammami and Zadeh, 2022 Chen, Chi, and Wang, 2015). However, this method is undesirable for retail investors. Databases, like COMPUSTAT, de-

mand a subscription, which reduces the accessibility for investors (Palas, 2019). Furthermore, prior research by Tallapally, Luehlfing, and Motha (2011) have shown that data from COMPUSTAT can differ from the original financial data. So, a different method for EM detection could enable investors to perform their own analysis.

This study uses a new approach in the research of machine learning in EM detection, which is eXtensible Business Reporting Language (XBRL). This software is used to exchange business information, such as financial statements, and is XML language based. This software is free and accessible to retail investors. In the setting of this study, XBRL could help investors to analyse financial statements for EM detection. Prior research has used XBRL for several purposes, such as its general adaption (Zhang, Guan, and Kim, 2019), its extensions (Johnston, 2020), and to access financial data for earnings predictions (Rao and Guo, 2022). Nevertheless, there is no literature on its usefulness in detecting EM, even though it could be a great solution for retail investors to overcome the problem of not having access to databases. This study attempts to fill this gap by exploring whether XBRL can be used to predict EM. Therefore, the goal of this research is to predict EM using XBRL though various machine learning models.

1.2 Problem Statement and Research Questions

The problem statement for this research is:

The current state-of-the-art regarding earnings management (EM) detection is not usable for retail investors, since it is often based on subscription-based databases. This study will evaluate various machine learning models to find which model performs best in detecting EM using XBRL, an open-access system.

This problem statement can be split into 4 research questions:

- How can EM be proxied using financial statements.
- Which set of financial statement items contains relevant information that can be used in EM detection?
- Which machine learning model achieves the highest accuracy and lowest false positive rate in detecting EM?
- Can XBRL be used to detect EM?

1.3 Managerial Relevance

EM detection could help investors in their investment decisions, since it will inform them as to which companies have likely engaged in EM. As mentioned before, current accounting practices that examine EM are limited to manual verification and the use of simple ratios (Dbouk, 2017). These methods are regarded as slow and inaccurate. Hence, it is important to integrate current technological developments into the EM literature. This study attempts to

achieve this by applying machine learning models on financial statements that are retrieved using XBRL, with the goal of building a model that could accurately detect EM. As XBRL is freely accessible, and an extra elaboration on how investors van use the results for their own analysis will be discussed in section 5, it will enable investors to analyse financial statements to detect EM. This makes this study relevant for small retail investors, but also larger institutions who want to analyse their portfolios. Therefore, the results of this study should enable investors to make better investment decisions.

1.4 Academic Relevance

This study relates to the accounting and fraud detection literature stream. It builds on several studies in the field of EM, machine learning, and XBRL, and attempts to fill certain gaps in the literature. First of all, this research adds to prior research by Chen, Chi, and Wang (2015) and Hammami and Zadeh (2022), who conducted a similar study in the field om EM detection. However, instead of using databases or collecting data manually, this study uses XBRL. This makes the results replicable and allows retail investors to analyse a certain set of companies of their interest.

Furthermore, in contrast to other EM detection studies, this study uses a larger set of features. It uses a set that has been used to predict earnings (Palas, 2019)[1]. In this context, since the set of features seems to be important determinants in earnings prediction, it could also potentially mean that they are important for EM prediction. This could lead to new insights as to which features are important in EM prediction. Another gap-filling addition of this research is that it will incorporate all its findings in a GitHub repository and will explain how investors can use this for their own analysis. This makes the results of this study replicable by stakeholders who want to analyse a different set of firms.

1.5 Thesis Overview

The subsequent sections of this thesis has the following structure. In Chapter 2, a review of the existing literature concerning XBRL and EM is presented. Chapter 3 presents the data and methodology used in the current research, including descriptive statistics. Chapter 4 is devoted to the results generated by the research methodology. Lastly, Chapter 5 provides the concluding remarks of this study, including a discussion of the implications of the findings, as well as recommendations for future research.

[1]The list that (Palas, 2019) is slightly modified, as some variables had too many missing values

2 Theoretical Background

2.1 Earnings Management

This chapter provides a review of the existing literature on Earnings Management (EM) and its implications. Firstly, EM will be defined and distinguished from fraud, and its motives will be outlined. Secondly, the different forms of EM detection, both in literature as well as in practice, are examined in detail. Finally, the potential consequences of EM are discussed, emphasizing the importance of detecting and preventing such practices.

2.1.1 Introduction to Earnings Management

Financial statements serve as an important source of information for various stakeholders, including borrowers, analysts, and investors (Almaqtari et al., 2021). These statements are expected to adhere to the accounting standards set by regulatory bodies. However, there is a risk that firms may not comply with these standards, leading to the manipulation of its earnings. Such practices can potentially mislead stakeholders and reduce the trust they have in financial statements. Therefore, detecting and preventing earnings management has become a crucial area of research.

The concept of earnings management (EM) refers to a set of practices that intentionally conceal a firm's actual performance, thereby compromising the reliability and credibility of financial reporting (Campa, 2019). One of the first researcher that explored the notion of EM was Messod D. Beneish, a prominent researcher in the field, who defined it as a deliberate process that operates within the confines of generally accepted accounting principles (GAAP) to achieve a desired level of reported earnings (Beneish, 2001). Prior literature has showed mixed results when it comes to whether the practice of EM is ethical or not. While some papers argue that EM should be considered as ethical (Jiraporn et al. (2008), Diana and Madalina (2007), and Bajra and Cadez (2018)), as it is line with the GAAP, others argue that EM is unethical (Beneish (2001) and Healy and Palepu (2003)), since it can mislead stakeholders. This study follows the second literature stream, and advocates that EM is unethical and should be prevented. It is important to notice the difference between EM and fraud.While fraud has the same objective as EM, its most prominent difference is that its practice is not within the GAAP, whereas EM's is (Perols and Lougee, 2011). This also entails that it is easier to label fraud than EM (Kassem, 2012), as fraud can be tested against the GAAP. Nevertheless, this study focuses on the detection of EM using the discretionary accruals, which is a widely-used proxy (Hammami and Zadeh, 2022; Chen and Howard, 2016; Dbouk, 2017)

2.1.2 Motives for Earnings Management

Firms have several motives to engage in EM. Management of firms might decide to engage in EM to meet certain debt covenants (Franz, HassabElnaby, and Lobo, 2014).The violation of these covenants could push a firm in a negative spiral of bad publicity and negative stock behavior. Therefore, firms might adapt EM to ensure all covenants are met. Another motive

to engage in EM is related to the bonus scheme of managers (Iatridis and Kadorinis, 2009). This is often related to the reported earnings in financial statements. Therefore, managers might feel the incentive to manage their earnings to meet certain bonus target. Both these types of EM can mislead stakeholders that use financial statements in their decision-making. Therefore, its detection is crucial, and is further explored in the next subsection

2.1.3 Detection of Earnings Management

Auditors use several tools to attempt to detect EM. These methods include manual verification of general ledger accounts and financial ratio analysis(Sanad, 2021). However, these methods are outdated and not in line with the current digital developments. Literature has attempted to use a more analytical approach to extract the discretionary accruals from the total accruals (Tsai and Chiou, 2009; Rahul, Seth, and Kumar, 2018; Hammami and Zadeh, 2022). The discretionary accruals are used as a proxy for EM. While many studies used this method to research EM, there has also been several studies that argue that other methods should be considered (Höglund, 2012; Dechow et al., 2012). This literature stream advocates that discretionary accruals are difficult to isolate from the total accruals, and that its drivers are often unknown. This is also acknowledged by Hammami and Zadeh (2022), who emphasize the importance of EM detection, but argue that there are only a few studies who have developed a model to do so.

Hammami and Zadeh (2022) used several ensemble classifiers to predict EM. They used a set of financial statement items and ratios as features to predict the discretionary accruals of a sample of companies. One of their ensemble models resulted in an out-of-sample accuracy of 90.4%. This result could be used by investors for their decision-making, and auditors for their monitoring. This study is one of the few papers that incorporated machine learning in EM detection, using financial statement items as features. Furthermore, it is in line with Sanad (2021), who argues that EM research should shift to using machine learning models in its detection. Still, the paper of Hammami and Zadeh (2022) has several drawbacks. Firstly, the authors only focus on a predetermined set of 20 features, and apply feature selection methods on this set. However, prior literature by Palas (2019) used a set of 60 financial statement features to predict earnings. This study uses a similar approach and will use a larger set of financial statement features. The second drawback of the paper of Hammami and Zadeh (2022) is related to their method of data retrieval. They retrieved their data from the COMPUSTAT database, which requires a license that most retail investors do not have. At the same time, they argue the usefulness of their findings for investor. However, most investors will not be able to replicate the results of the study, as they will not be able to access the data themselves. This study attempts to solve this problem by incorporating a new method in the use of machine learning in EM detection, which is eXtenisve Business Reporting Language (XBRL). Next section will discuss what this method entails and how it could solve the aforementioned problem of data availability for retail investors

2.2 XBRL

This section covers prior literature on the topic of XBRL. First, a short introduction on XBRL and its architecture is covered. Next, its benefits and challenges in research are outlined. Finally, prior studies that combined EM and XBRL are discussed

2.2.1 XBRL Introduction and Architecture

XBRL is a business and reporting technology that facilitates the sharing of information (Palas, 2019). It allows companies to share information in a standardized way, whilst financial statement users can use the information for their needs. XBRL reduces information processing costs, as it allows users to accurately compare accounting information across firms (Janvrin, Pinsker, and Mascha, 2013). This is because the use of XBRL ensures that financial statements are machine-readable (Rao and Guo, 2022)

The U.S. Securities and Exchange commission (SEC) obliged public companies in 2009 to adopt XBRL in their financial reporting (Zhang, Guan, and Kim, 2019). XBRL for financial statements is composed on several components, including XBRL taxonomy, XBRL linkbases and XBRL Instance Documents.

The XBRL taxonomy is a dictionary that consists of XBRL tags that company can use to tag specific financial statements. A tag consists of two elements. The first element refers to the regulations where the specific financial statement item refers to, whereas the second element consists of the actual label that corresponds with the financial statement item. For example, the cost of goods sold that a company reports with the U.S. GAAP into account is labeled as *us.gaap:CostOfGoodsAndServicesSold*. The taxonomy helps companies that report their financial statements using XBRL on which labels they can use. Similarly, it can guide investors in what a specific financial statement item entails, and can help them to compare similar items across companies (Perdana, Robb, and Rohde, 2015). Still, there is some critique in the literature on the latter statement due to the errors that companies make using the tags. Du, Vasarhelyi, and Zheng (2013) analyzed 4,532 financial statements and found a total of 4,260 errors, which entails 0.940 errors per statement. Furthermore, In 2013, the U.S. GAAP taxonomy consisted of around 19,000 tags that companies were allowed to use in their reporting (Palas, 2019), indicating the size of the taxonomy that companies need to choose from. The challenges of XBRL is further outlined in 2.2.3

The XBRL linkbases are additional information to the XBRL taxonomy and consists of metadata for each XBRL tag. It consists of several components; Reference, Calculation, Definition, Label and Presentation, (Yaghoobirafi and Nazemi, 2019). They all fulfill their own objective in making XBRL documents more readible. For example, the presentation linkbase defines the structure of the tables in the XBRL documents.

The XBRL instance document is the file that contains all the financial statement information. The SEC has its own platform called the Electronic Data Gathering Analysis and retrieval (EDGAR), where company's instance documents are stored.

2.2.2 XBRL Benefits

As mentioned in the previous paragraph, the use of XBRL for financial statements has several benefits, especially for financial statement stakeholders. The use of XBRL allows it users to process, validate and analyze financial statements in a time-efficient manner (Liu et al., 2017). As the reports are machine-readable and consist of standardized tags that are part of the XBRL taxonomy, the issue of comparability of financial statements seems to be resolved. In fact, the SEC (SEC, 2009) stated that: *If interactive data serves to lower the data aggregation costs as expected, then it is further expected that smaller investors will have greater access to financial data than before.".* This should lead to lower level of information asymmetry between companies and investors, which is shown by (Liu et al., 2017).

Another benefit of XBRL is that it is an exact copy of the actual financial statement, unlike financial databases like COMPUSTAT which may show different amounts than the actual financial statement (Chychyla and Kogan, 2015). This also entails that XBRL includes information that are stored in the notes of financial statements, which are often not accessible via third-party databases (Hoitash, Hoitash, and Morris, 2021). On top of that, financial statements are immediately accessible after its publication, unlike databases that are not real-time updated, improving the information efficiency in capital markets (Efendi, Park, and Smith, 2014). Still, whilst the benefits of XBRL seem promising, it has also led to several challenges for both researchers as well as its users. Next subsection discusses the challenges of XBRL.

2.2.3 XBRL Challenges

To fulfill its purpose of easy information-gathering, it is important that the data quality of XBRL follows high standards. Nevertheless, some early evidence on the data quality of XBRL has shown otherwise. Debreceny et al. (2010) performed research on the first year that firms were obliged to submit their financial statements in XBRL format by the SEC. They found that financial statements had 1.8 errors on average, which is associated with a median error of $9.1 m. They argue that most of these errors were related to the lack of knowledge on the use of XBRL by fillers. Even though the error rate in XBRL instance documents has decreased over time due to companies' experimental learning (Bartley, Chen, and Taylor, 2011 & Perdana, Robb, and Rohde, 2019), the use of XBRL by investors is still limited (Janvrin, Pinsker, and Mascha, 2013). Therefore, whilst Efendi, Park, and Smith (2014) argue that XBRL should lead to better data processing efficiency, Rao and Guo (2022) tested this statement and found that XBRL has a non-significant effect on data efficiency.

Whilst Liu et al. (2017) found a decrease in information asymmetry between companies and investors due to the adoption of XBRL, Blankespoor, Miller, and White (2014) has shown that the information asymmetry within the group of investors has actually increased. They found that investors have different levels of technical capabilities, which may have worsen the information asymmetry between small retail investors and large institutional investors. This in contrast with the SEC's goal for XBRL, which was to *"If interactive data serves to lower the data aggregation costs as expected, then it is further expected that smaller investors will have greater access to financial data than before"* (SEC, 2009). Guo and Yu (2022) tested

the use of XBRL by small retail investors and found that this group does not use XBRL as much as was expected by the SEC due to the aforementioned issues. One of the reason that the authors give is that the use of XBRL demands a certain level of technical expertise. Still, it has its advantages in comparison with third-party databases like COMPUSTAT when it comes to correctness, completeness, and timeliness. This study attempts to test whether XBRL can be used to detect EM. Next subsection will discuss some prior research in the field of XBRL and EM.

2.2.4 XBRL and EM

There have been several studies that combined both the XBRL and EM literature. Kim, Kim, and Lim (2019) used a different perspective for the two fields and analyzed whether the adoption of XBRL would lead to lower EM. They found that the discretionary accruals, which served as a proxy for EM, were significantly lower in the post-XBRL period compared to the pre-XBRL period. Whilst this is a different application of the two fields in research, its implication for this study is that it EM detected will likely be of a smaller magnitude than years before, as this study only focuses on the post-XBRL period. This statement is supported by other studies by Peng, Shon, and Tan (2011) and Mayapada, Afdhal, and Syafitri (2020), who argue that the lower EM in the post-XBRL period is due to the less opportunistic behavior in financial reporting

Most papers in this field focused on the adoption of XBRL and its effect of the size of EM, rather than using XBRL for EM detection. Still, the purpose of the adoption of XBRL by the SEC was to a *'create new ways for investors, analysts, and others to retrieve and use financial information in documents filed with us.'* Therefore, XBRL should enable investors and other stakeholders to analyze financial statements for various purposes, like EM detection. Still, to my knowledge, there have not been any papers that attempted to predict EM using XBRL data. This could be due to the high error-rate that researchers found in XBRL data, which lowers their interest to use XBRL data. In fact, Ahmi and Mohd Nasir (2019) researched the trend of XBRL studies in this century and found that there are less academic papers written on XBRL in the period 2011 and 2019 compared to the period 2001 and 2011, which could be a consequence of the low level of interest in XBRL in recent years. Still, XBRL has practical advantages over other methods like COMPUSTAT in terms of time and efficiency. This study fills the gap of EM detection using XBRL and attempts to assess the use-fullness of this method.

3 Data and Methodology

This chapter discusses the data and methodology of this study. Firstly, the data sources and how the data is handled, will be discussed. Next, the methodology to answer the research question is outlined, including an overview of all the machine learning models applied. Finally, some descriptive statistics are plotted to show how the data looks like.

3.1 Data

The sample used in this research includes all firms listed on the S&P 500, excluding financial institutions, between 2018 and 2022. The reason why this index is used is twofold. First of all, prior literature has shown that big companies are more likely to engage in EM (Liu et al., 2017). Furthermore, the companies on this index make up for 79 % of the total U.S. equity market capitalization Hammami and Zadeh (2022), making it interesting for retail investors. Financial institutions are excluded due to their specific reporting style. Furthermore, data from 2018 is used since this was the first year that the SEC obliged firms to publish their financial statements in "inline XBRL" format. By not using data before this in the analysis part, this study attempts to increase the data quality. For all firms, all numeric financial statement information that is stored in XBRL format is extracted using a self-built scraper in R. This scraper attempts to access the data via the SEC's Electronic Data Gathering and Retrieval (EDGAR) system. As the composition of the S&P 500 changes over time, the composition per December of each year is used as the composition of that respective year. Appendix A shows all the features that are used in this research, including how they are calculated. Since some variables are constructed by calculating the change in a specific year, data of 2017 is also retrieved to calculate the changes in 2018. After calculating all variables, they are winsorized on the 1% and 99% level to deal with outliers, and then normalized such that the mean is equal to zero and the standard deviation is equal to one. This last step is used to control for the difference in scales between the absolute and relative features.

As mentioned before, one of the biggest drawbacks of using XBRL is related to its quality of data. The labeling of the financial statement items is inconsistent. Appendix B shows a list of all the specific financial statement items that are used for the calculations, and the labels that are considered to be part of that specific variable. For each firm and for each variable, I looped through the list in the order as specified in Appendix B, and assigned the firm the first label that it had. The order is based on how often the label occurred in the total data set. Next section discusses the methodology of this study.

3.2 Methodology

This section discusses the methodology of this research. Firstly, the way EM is calculated in this research is outlined. Next, the feature selection method that is used to reduce the dimensions in the set of features, is discussed. Finally, the machine learning algorithms used in this study are motivated.

3.2.1 Earnings Management Proxy

As mentioned before, this study will use the discretionary accruals as a proxy for EM. This study will use a similar approach as proposed by Kothari, Leone, and Wasley (2005) and Jackson (2018), which are both based on the performance-adjusted Jones model;

$$\frac{TA_{i,t}}{AT_{i,t-1}} = \beta_0 + \beta_1 \frac{1}{AT_{i,t-1}} + \beta_2 \frac{(\Delta REV_{i,t} - \Delta REC_{i,t})}{AT_{i,t-1}} + \beta_3 \frac{PPE_{i,t}}{AT_{i,t-1}} + \beta_4 ROA_{i,t} + \epsilon_{i,t} \quad (1)$$

Where TA is total accruals defined as total current accruals less depreciation, total current accruals is the change in current assets less the change in current liabilities less the change in cash plus the change in the debt in current liabilities;ΔREV is the change in sales; ΔREC is the change in accounts; PPE is gross property, plant, and; ROA is the return on assets calculated as earnings divided by lagged total assets. All variables (excluding ROA) are deflated by the lagged total assets (AT).

The residual ϵ_{i_t} of (1), which will be estimated using simple OLS, is the discretionary accruals and will be used as a proxy for EM for company i in year t. Since lagged variables are used in this model, information from one year before the period of interest is retrieved to estimate the model for the whole period.This should result in a numeric proxy for EM per company per year. Instead of using this number in further analysis, this study will discretize this variable. Even though this will lead to a loss in model performance, since information is removed, it is still essential for this study. By discretizing the EM proxy, this study aims to achieve a well-working algorithm that can recommend investors to keep/drop a certain company in their investment decision. This makes the model more interpretable for retail investors compared to when continuous numbers were used. To discretize the EM proxy, a similar approach as Hammami and Zadeh (2022) will be used, by defining a ceiling and a floor. Ceiling is defined as the average value of EM plus one standard deviation, and floor is defined as the average value of EM minus one standard deviation. All EM proxy values between the floor and ceiling are assigned the value zero, whilst values outside of this are assigned the value one[2]. This approach also overcomes an often occurring problem of imbalanced data in fraud detection research, since both classes will now be of sufficient size to apply machine learning algorithms. The table below summarizes how the proxy for EM will be refered to in the remaining of this thesis;

Table 1: EM Proxy

This table shows how the EM proxy is constructed and labeled.

Label	Name	Abbreviation	Interval
0	Moderately Upwards/Downwards Proxy for EM	Moderate EM	$[\bar{x} - \sigma, \bar{x} + \sigma]$
1	Extremely Upwards/Downwards Proxy for EM	Extreme EM	$(-\infty, \bar{x} - \sigma) \cup (\bar{x} + \sigma, \infty)$

[2]Hammami and Zadeh, 2022 did not create a binary outcome, but made discrete buckets with values ranging from -2 and 2.

3.2.2 Lasso Regularization

Before various machine learning models are evaluated, this study will consider regularization to control for the number of features used. The reason why regularization is used in this research is two fold Gareth et al. (2013). First of all, by reducing the number of features used, the variance in the estimated coefficients will decrease since redundant variables are removed. This can consequentially lead to a higher accuracy. Secondly, the model will be easier to interpret, as irrelevant variables can be removed.

There are different ways to control for the number of features or dimensions. This study will focus on lasso regularization. This approach is based on fitting all features. However, the estimated coefficients are shrunken towards zero. In lasso regularization, some features can be shrunken to exactly zero, which makes this method useful for variable selection.

Lasso regression is based on a OLS regression model. The OLS fitting procedure to calculate $\beta_0, \beta_1 .... \beta_p$ is done by minimizing the residual sum of squares:

$$_{min} RSS = \sum_{i=1}^{n} (y_i - \sum_{j=1}^{p} \beta_j x_{i,j})^2 \tag{2}$$

Lasso regression has a similar formula, but has an extra term that is used to shrink the size of estimated parameters towards zero. This is done by minimizing the following formula, where λ, which is the penalty parameter, is tuned:

$$_{min} RSS + \lambda \sum_{j=1}^{p} \|\beta_j| \tag{3}$$

λ will be tuned. The model chosen will be chosen based on the one-standard-error rule for the accuracy. This rule entails that the most parsimonious model is chosen that is within one standard-error of the best model. Based on this model, all the coefficients are estimated. All estimated coefficients that are shrunken towards exactly zero will be disregarded in the machine learning models.

3.2.3 Random Forest

A random forest is an ensemble method that combines multiple decision trees to make more accurate predictions than any individual tree (Gareth et al., 2013). This algorithm attempts to build several trees on randomly selected subsets of the data and features. By only using a subset of the features for each tree, it will solve any correlation among trees that might occur is there is one strong feature. This is because otherwise, each tree would use this strong feature to split the data, which will result in highly similar trees. For prediction purposes, the algorithm will aggregate the predictions of all trees to come up with a final prediction.

Two parameters will be tuned using a grid search. The first parameter is *mtry*, which refers to the number of features used at each split. The second tuning parameter is the number of trees to build. This tuning grid is set between 50 en 2000, with steps of 50. 10-fold cross validation is applied to increase the model performance on new data

3.2.4 Gradient Boosting

The second algorithm is also an ensemble method which attempts to improve the performance of a single tree. The goal of gradient boosting is to learn *sequentially*; this entails that each tree is grown using the information of the previous trees. It attempts to correct the errors that the previous trees made. This approach allows the algorithm to focus on the examples that are most difficult to predict, and produce accurate predictions even when the data is noisy or contains outliers. Gradient boosting has the following procedure:

Algorithm 1 Gradient Boosting Algorithm

0: Set $\hat{f}(X) = 0$ and $r_i = y_i$ for all $i = 1, \ldots, n$
0: **for** $b = 1, \ldots, B$ **do**
0: Fit tree $\hat{f}_b$ with d splits to the data (X, r)
0: Update $\hat{f}$ by adding in a shrunken version of the new tree: $\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}_b(x)$
0: Update the residuals: $r_i \leftarrow r_i - \lambda \hat{f}_b(x_i)$
0: **end for**
0: Output the boosted model: $\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}_b(x)$ =0

Several parameters are tuned. The first tuning parameter is the number of trees B. The second parameter is the shrinkage parameter λ. Finally, the number of splits in each tree will also be tuned

3.2.5 Support Vector Machine

The support vector machine (SVM) is a algorithm that is widely used for classification tasks. The goal of a SVM is to construct a hyperplane with *(p-1)* dimensions that can separate the data (Gareth et al., 2013). This hyperplane is selected in such a way that its Euclidean distance with the nearest data points is maximized. These data points are called support vectors. SVM is useful for high-dimensional data and to model non-linearity. This is due to the kernel function that is used with a SVM. The kernel function transforms the data into a higher-dimensions feature space to control for non-linearity. This can be useful if the input data shows non-linearity, which entails that a linear hyperplane will have a poor performance. Therefore, the data will be transformed into a higher-dimensional space to deal with the non-linearity. The kernel used in this research will be the Radial Basis Function (RBF) kernel. It computes the similarity of two points $X_1 \& X_2$ using high-dimensional transformations.

The SVM is an optimization problem and is given by:

Algorithm 2 Optimization problem for SVM with RBF kernel

0: **Maximize:** M
0: **Subject to:**
0: $\sum_{i=1}^{n} \beta_i - \sum_{i=1}^{n} \beta_i y_i K(\mathbf{x}i, \mathbf{x}j) = 0, \quad \forall j \in 1, 2, ..., n$
0: $0 \leq \beta_i \leq C, \quad \forall i \in 1, 2, ..., n$
0: $0 \leq \epsilon_i \leq C, \quad \forall i \in 1, 2, ..., n$
0: $\sum i = 1^n \epsilon_i \leq C$
0: $y_i \left(\sum j = 1^n \beta_j y_j K(\mathbf{x}_i, \mathbf{x}_j) + \beta_0 \right) \geq M(1 - \xi_i), \quad \forall i \in 1, 2, ..., n$
0: $\xi_i \geq 0, \quad \forall i \in 1, 2, ..., n$ =0

β_i are the Lagrange multipliers. ϵ_i and ξ_i are slack variables. y_i is the target variable of the i-th training instance. $K(\mathbf{x}_i, \mathbf{x}_j)$ is the RBF kernel function, defined as $K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma||\mathbf{x}_i - \mathbf{x}_j||^2\right)$, where γ is a parameter that controls the width of the kernel. C is a regularization parameter that controls the trade-off between maximizing the margin and minimizing the classification error on the training data.

Two parameters will be tuned. The first parameter is the Cost (C), which controls for the degree of misclassification of the data. The second parameter that will be tuned is the γ, which determines the width of the kernel function.

3.3 Descriptive Statistics

This subsection covers some descriptive statistics that can help to better understand the data. The way the final data set is constructed is described in table 6. First, all the ticker symbols from the S&P 500 per December of each year are retrieved. Due to ongoing changes to the index, the number of ticker symbols per year differs from 500. Next, all financial ticker symbols are removed from the data set. This is because these firms have a different reporting style and would bias the results. Afterwards, a scraped is applied to the remaining data set. A total of 55 ticker symbols did not provide a response on the API, and are therefore removed. Next, 460 tickers are removed because they have at least one missing value in the variables of interest. This can be due to many reasons, but is often due to the specific reporting style that the company uses. What can be observed from the table is that the number of ticker symbols removed is decreasing over the sample period. This seems to indicate that the data quality and comparability over time is increasing. This is line with prior research by (Perdana, Robb, and Rohde (2019), who argue that the quality of XBRL reporting increases over time due to experimental learning and easier-to-understand instructions for companies. Finally, another 216 ticker symbols are removed. This is because some variables of interest are based on changes with the previous year, which leads to the first time the firm occurring in the data set, the respective delta being zero. To ensure that this will not bias the results, these ticker symbols are removed from the dataset. Data of 2017 is also retrieved and used to lower the possible number of missing values. All these modifications lead to a final data set with 1238 tickers that are used for further analysis.

Table 2: Firm Selection

The table reports the steps that are taken to which led to the final Dataframe. The data sample consists of all the ticker symbols on the S&P 500 between 2018 and 2022, which leads to a set of 2467 ticker symbols. The final Dataframe consist of 1238 tickers. This dataset will be used for the analysis/.

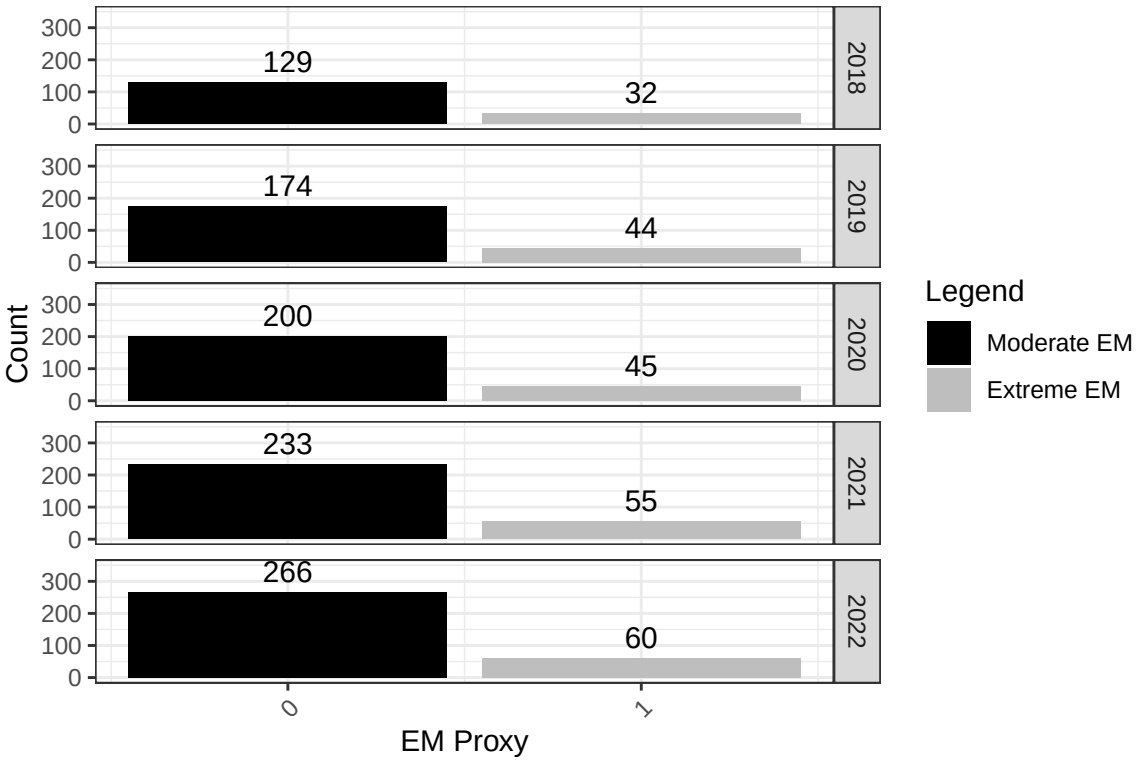
	2018	2019	2020	2021	2022	Total
Total Number of tickers on the S&P 500	483	490	494	497	503	2467
Number of financial tickers	100	101	98	97	102	498
Total number of non-financial tickers	383	389	396	400	401	1969
Number of tickers with no data from scraper	14	6	20	11	4	55
Total number of tickers with data	369	383	376	389	397	1914
Number of tickers with missing data	147	126	79	59	49	460
Total number of tickers with no missing data	222	257	297	330	348	1454
Number of new tickers that occur for the first time	61	39	52	42	22	216
Final Dataframe	**161**	**218**	**245**	**288**	**326**	**1238**

Histograms of all the features used in this research are added to figure 9 in subsection 7.3. Most distributions have fat tails, which is a consequence of the winsorizing. Furthermore, all values are normalized with a mean of zero and a standard deviation of 1, to control for the difference units of measurement across the features.

figure 1 illustrates how the AEM proxy is distributed. The figure shows the number of moderately upwards/downwards (0) and extremely upwards/downwards (1) observations per year. Both buckets increase over time, indicating that the number of complete observations increase over time, which could also be seen in table 6.

Figure 1: Earnings Management Distribution

The graph shows the count distribution of the Accrual-based earnings management proxy per year.

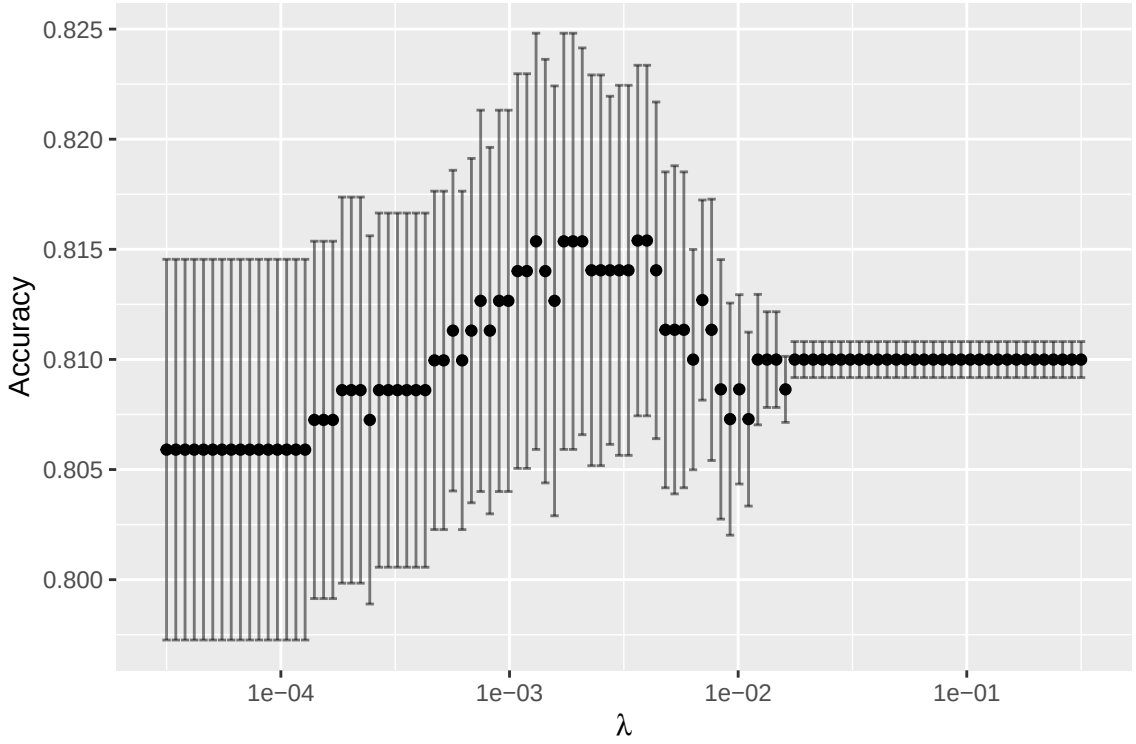


4 Results

4.1 Lasso Regularization

This subsection discusses the results of the lasso regularization. The goal of this step is to control for the number of features that are used in this model. All variables that have estimated coefficients that are shrunken to zero will not be used in any further analysis. The penalty parameter λ is tuned based on 100 different values ranging from $10^{-4.5}$ and $10^{-0.5}$. The model is than evaluated based on its accuracy. figure 2 shows the accuracy across all penalty parameters. The accuracy seems to fluctuate around 81%. The penalty parameter that is eventually chosen is based on the one-standard-error-rule, i.e. the most parsimonious model within one standard error of the best model. The value of λ is 0.03635851.

Figure 2: Lasso regularization
The graph shows the accuracy across a range of penalties.



Based on this penalty parameter, the coefficients $\beta_0, \beta_1 .... \beta_p$ are estimated using the training set and the results are evaluated based on the test set. The metric results on the test set are shown in table 3. The accuracy is around 80%. This seems to indicate a relatively good out-of-sample performance,.

The estimated coefficients, based on the aforementioned penalty parameter, are shown in table 8. All estimated coefficients are shrunken towards zero in comparison to its OLS estimate. Five variables have estimated coefficients equal to zero and seem to be redundant in detecting earnings management. These variables are: *"EquityOverFixedAssets", "PreTaxIncomeToSales", "ChangeInAssets", "ChangeInRevenues", "ChangeInDaysSalesinAccountingReceivable"* Including these variables will lead to less price estimates of the other variables. Therefore, they are not used in further analysis.

Table 3: Metrics
The table shows the out-of-sample metrics of the lasso regularization method.

Metric	Estimator	Estimate
accuracy	binary	0.792
f_meas	binary	0.883
kap	binary	0.0293
bal_accuracy	binary	0.510

4.2 Machine learning models

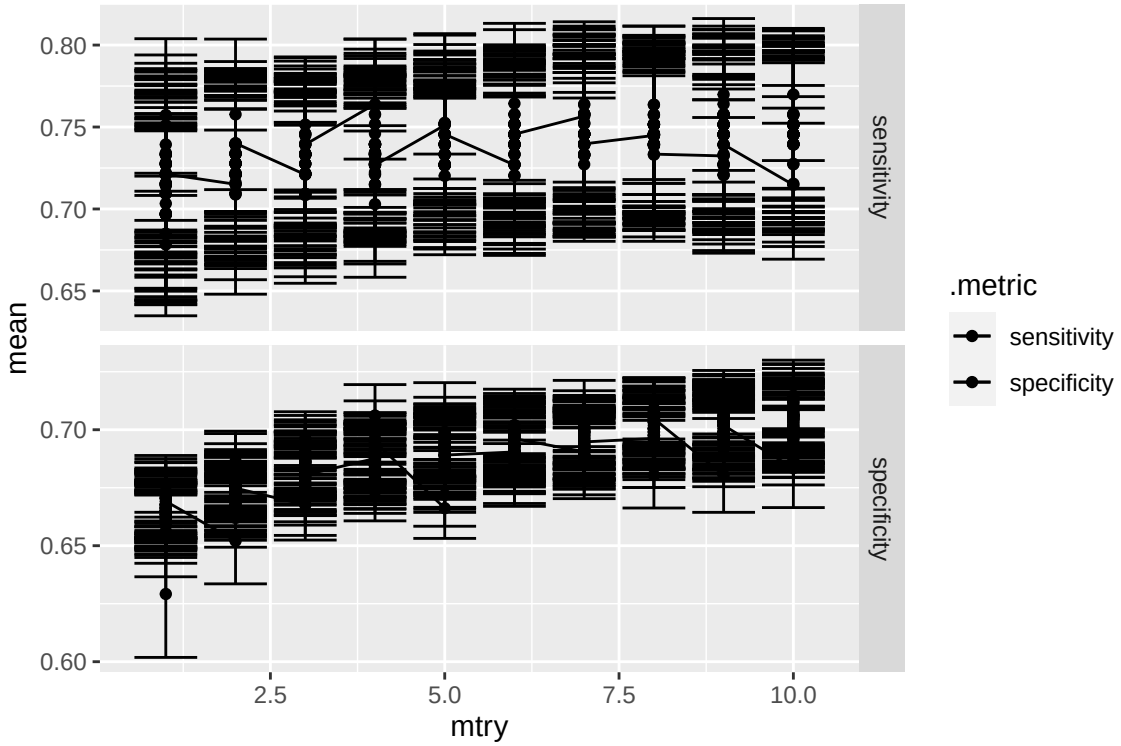
This section discusses the application of the machine learning models as discussed in subsection 3.2. All the models have a fairly similar set-up. The data is split into 70%/ 30% for training and testing. The split is stratified on the dependent variable, to ensure that the distribution of the dependent variable is the same for both the training and testing set. Since there is an unbalance between the two classes of the dependent variable, the larger class is sampled to ensure that the sizes of the two classes are (close to) equal. 10-fold cross-validation is applied to ensure that the model is trained on different subsets of the data. The metrics that are used to evaluate the results are accuracy, sensitivity, specificity and ROC-AUC. The goal is to maximize sensitivity, whilst maintaining a good level of specificity. Sensitivity relates to the number of false negatives, which are all the tickers that had an extremely upwards/downwards proxy for EM, but were predicted to have a moderately upwards/downwards proxy for EM. From an investor perspective, it is important to keep the number of false positives as low as possible, as these firms should have been avoided. The specificity is less important, as it focuses on the false positives, which are all the tickers for which the observed proxy for EM is extremely upwards/downwards, but the predicted proxy is moderately upwards/downwards.

All models are tuned to choose appropriate hyperparameters, which will be applied to the test set. The first model that is tuned is the Random forest. The number of features used at each split at each tree and the number of trees are both tuned. This will prevent over fitting, reduce model complexity and improve model accuracy. The two figures (figure 3 and figure 4) show the results of the model tuning. As can be observed from the graphs, the accuracy seems to be relatively constant across all tuned models, but the specificity and sensitivity does differ. The model chosen is the model with the highest level of sensitivity, as it corresponds with a good level of specificity and accuracy. Next, three hyperparameters of the gradient boosting model are tuned. The first tuning parameter is the number of trees, and is set between 500 and 10000 with steps of 500. The second tuning parameter is the learning rate, which is set on either 0.1 or 0.10. The third and final tuning parameter is the tree depth, and is set between 1 and 3. The metric results of the model tuning can be found in table figure 5. The graph shows the metrics results across different different number of trees for the six different groups of learning rate and three depth. For a learning rate of 0.1, the model seems to perform better with a higher tree depth for the same number of trees. For a learning rate of 0.01, the figures remain relatively stable. The model chosen is based on the model with the highest sensitivity, which is the model corresponding with learn rate of 0.01, tree depth of 3, and the number of trees being equal to 1500. The third

and final model that is tuned is the Support Vector Machine (SVM). SVM is an algorithm that attempts to construct a hyperplane that could split the data. The dimensionality is transformed using the Radial Basis Function (RBF). The recipe used is similar to the recipe used in the ensemble method. Two parameters are tuned. The first parameter is the cost C, which controls for the degree of missclasification and is set between 10 and 100 with steps of 10. The second parameter is the λ, which control the width of the RBF. This parameter is set between 0.00001 and 0.1. The results of this tuning against the value of λ are plotted in figure 6. It is important to choose the hyperparameters carefully to ensure that the model can have a good out-of-sample performance. Based on figure 6, the hyperparameters chosen are 10 for the C and 0.01 for the λ. The model is trained using these parameters and fitted on the test set.

Figure 3: Sensitivity and Specificity RF tuning

The graph shows the sensitivity and specificity of the random forest on the training set



As all models are tuned and the hyperparameters are chosen, the model can be again trained using these hyperparameters on the training set and applied on the test set. The metric results of the application of the three machine learning models are shown in table 4

Table 4: Algorithm Metric Performance

This table shows the test results of the metrics for the Random Forest (RF), Gradient Boosting (GB), and Support Vector Machine (SVM) algorithms.

Metric	RF	GB	SVM
Accuracy	0.659	0.737	0.739
Sensitivity	0.803	0.704	0.577
Specificity	0.625	0.744	0.777
ROC AUC	0.793	0.785	0.739

Figure 4: Accuracy and ROC AUC RF tuning

The graph shows the accuracy and ROC AUC of the random forest on the training set

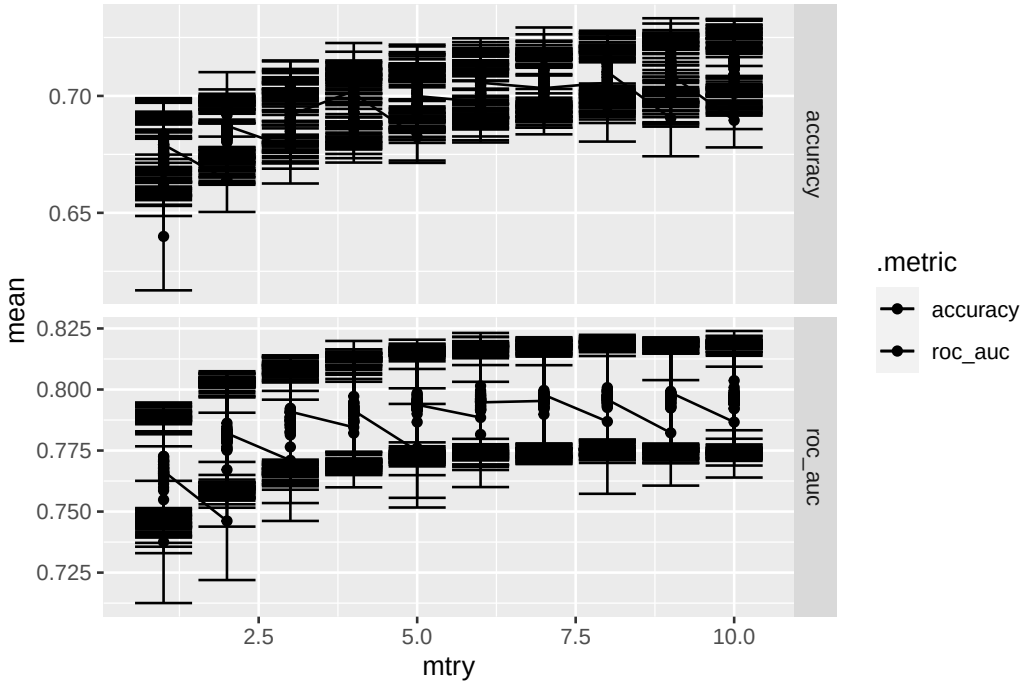


Figure 5: Metrics Gradient Boosting Tuning

the graph shows the tuning results of the gradient boosting algorithm

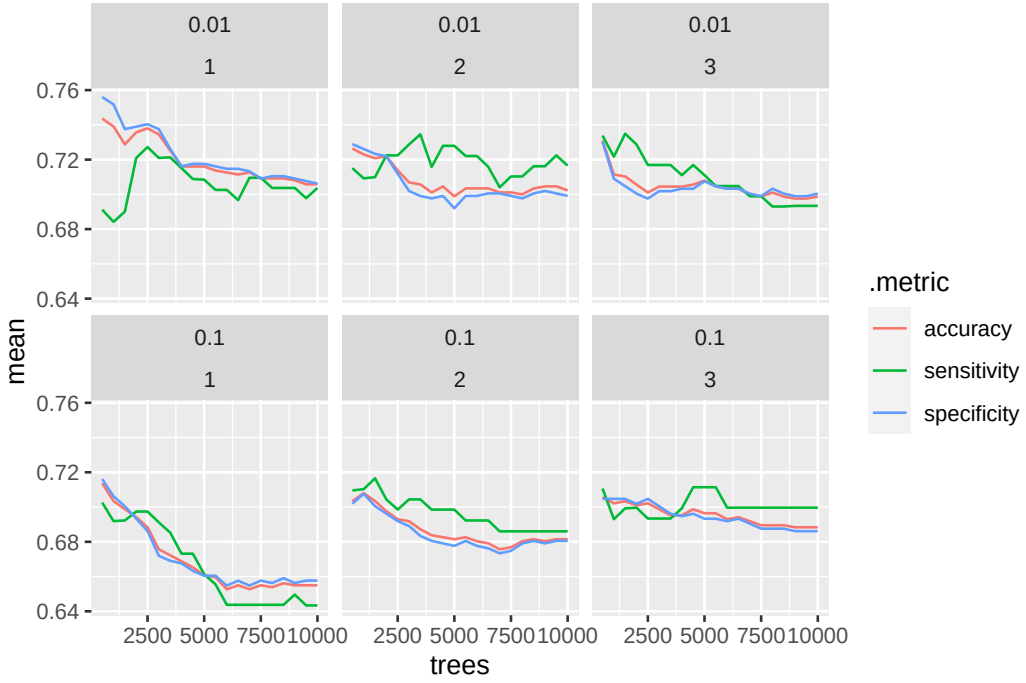
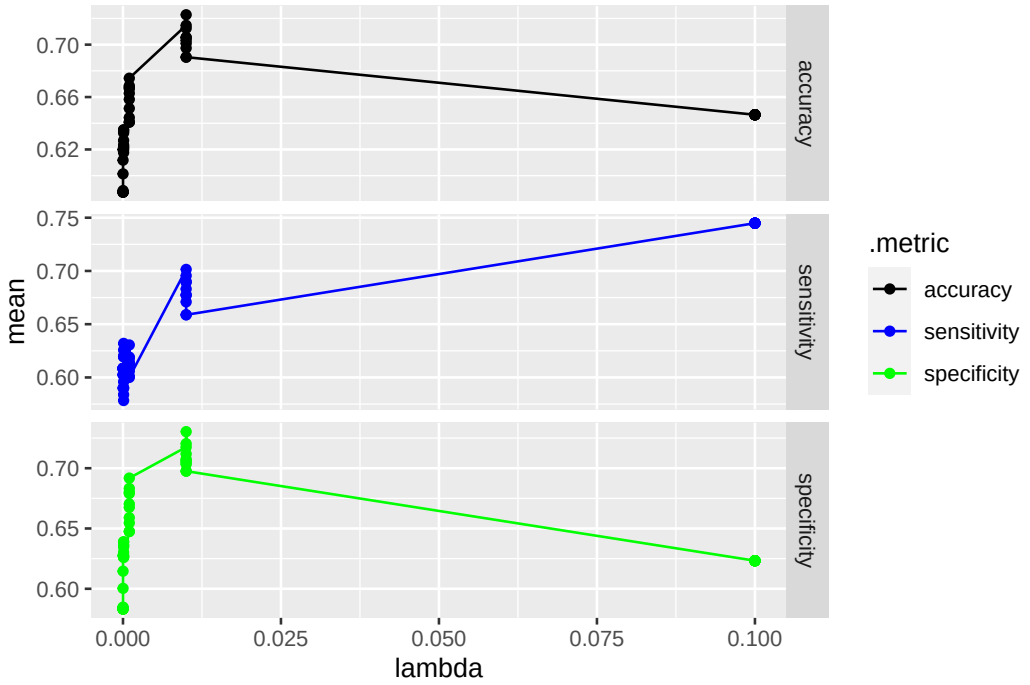


Figure 6: Metrics Support Vector Machine Tuning
the graph shows the tuning results of the SVM algorithm



Next, a confusion matrix corresponding to the results of the application of the machine learning models on the test set are shown in table 5

Table 5: Combined Confusion Matrix
This table shows the combined confusion matrices for the Random Forest (RF), Gradient Boosting (GB), and Support Vector Machine (SVM) models.

Actual	RF		GB		SVM	
	Moderate EM	Extreme EM	Moderate EM	Extreme EM	Moderate EM	Extre
Moderate EM	188	113	224	77	234	
Extreme EM	14	57	21	50	30	

The goal was to have a relatively high sensitivity whilst maintaining a good level of specificity and accuracy. This seems to be achieved. The test-set sensitivity is just over 80%, with a accuracy of close to 70%. There are only 14 ticker symbols that had en extremely upwards/downwards proxy for EM, whilst the model predicted it to have a moderately upwards/downwards proxy for EM. The goal was to keep this number as low as possible.

The results of applying the gradient boosting algorithm on the test set can be found in **??**. Next, a confusion matrix of the test set results is displayed in **??**. The metric results show that the specificity on the test set is slightly above 70 %, whilst the other metrics are a couple percentage points higher. The confusion matrix indicates that 21 ticker symbols were predicted to have a moderately upwards/downwards proxy for EM, whilst it was actually an extremely upwards/downwards proxy for EM. The model did a fairly good job to keep this number as low as possible, which resulted in the aforementioned specificity metric.

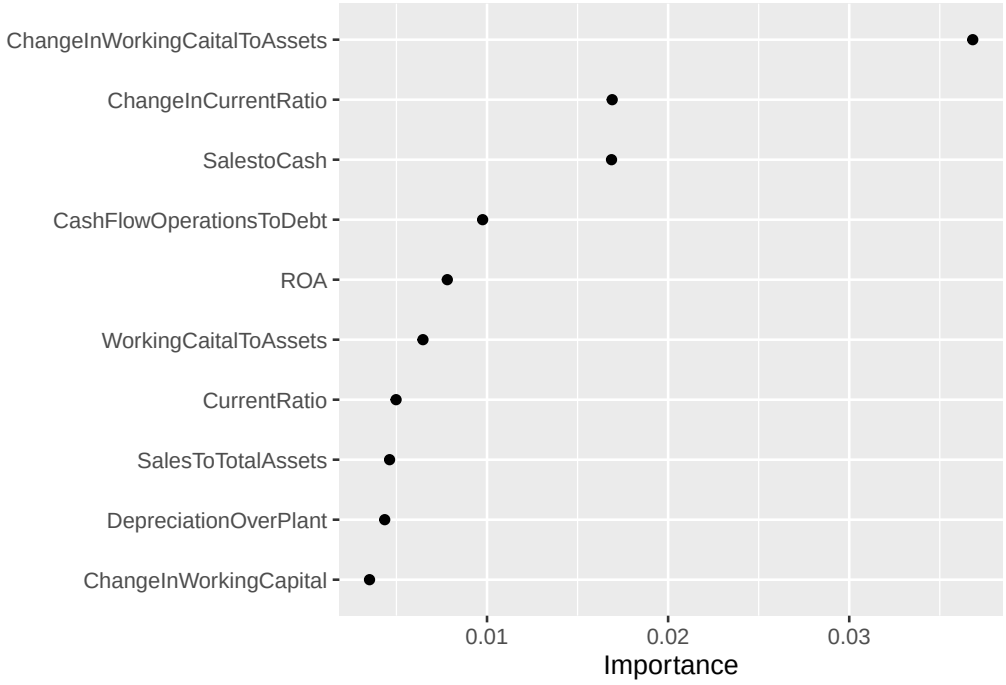
The test-set results are shown in **??**. As mentioned before, sensitivity is an important metric, as this metric indicates how well the model performed in terms of predicting the

firms that had an extremely upwards/downwards proxy for EM. However, the out-of-sample performance for this metric is relatively low at 57.7 %. Therefore, the model seems to perform poorly when it comes to predicting the firms that had an extremely upwards/downwards proxy for EM. In fact, as can be derived from **??**, from the 71 ticker symbols that have an extremely upwards/downwards proxy for EM in the test set, the model was only able to identify 41 correct. Still, the other three metrics seem to have a better out-of-sample performance.

Since random forest algorithms are based on multiple decision trees, it is not possible to evaluate in which direction certain features are split. Nevertheless, it is possible to evaluate the importance of the features, regardless of the way they are used in the decision trees. figure 7 shows the top 10 most important features, including their importance. The graph shows that the change in working capital to assets seems to be the most important feature in the random forest. This is in line with prior research by Aduda, Ongoro, et al. (2020), who argued that earnings management can be a consequence of working capital management. This study confirms this finding and argues that modifications in working capital to assets is one of the main drivers of EM.

Figure 7: Variable Importance RF

The graph shows the top 10 most important variables in the random forest model



5 Investor's application

5.1 Investor's Focus

This section will further elaborate on the aforementioned results, but than from the investor's perspective. As mentioned before, this study focuses on how investors can use the methods and results in their own EM detection analysis. The first step to achieve this was to include all relevant files and code in a GitHub repository (`https://github.com/sunny1999123/MasterThesis`). This chapter will elaborate on how investors could use this repository for their own analysis, and how the results should be interpreted. This enables investors with the necessary tools and knowledge to perform their own EM detection analysis[3].

5.2 Use of GitHub

All the relevant code and files are stored in a GitHub repository[4]. This platform enables version control and collaboration on projects. All the files and code can be collected via this repository and stored in a R project. To do so, one should open Rstudio, open a New Project, click on version control, and paste the aforementioned link to create a new project. The project, including all its files and code, will now be stored in a directory. This allows anyone now to replicate this study. Next paragraph elaborates on all the R code files that are used in this study.

5.3 Data retrieval & Cleaning

This study consists of several steps that are used to perform the analysis. All the R files start with a number and are stored in numeric order. The first file *1.Ticker Symbols.r* is used to create a table of ticker symbols of interest. The desirable outcome should be a dataframe with two columns; the first column should consist of the ticker symbols and the second column should consist of the year of interest for that respective ticker symbol. If an investor would like to replicate this study using a different set of ticker symbols, this R file can be skipped and one could start with the second R file. This file should take a dataframe with tickers and years as input. Next, in *2.Scraper.R*, all the data is scraped using an API, and stored in a dataframe. The third file, *3.Cleaning Results* takes all the data that is retrieved from the scraper as its input and tries to clean the results, such that only the relevant variables are kept and all other variables are deleted. Next, the features and EM proxy are calculated in *4.Feature Calculation*. This file provides a dataframe with all the relevant data that is needed for the machine learning models. Before these models can be applied, this study first focuses on selecting an appropriate set of features by applying lasso regularization. This is done using *5.Feature Selection*. Next, the three machine learning models applied in this research are stored in three different R files (*6.Random Forest, 7.Gradient Boosting and 8.SupportVectorMachine*). All these R files have a similar lay-out. The first step is to use the results from the regularization as input. Next, the respective model is trained, the

[3] A basic understanding of the R programming language, as well as data analysis in general is required to perform the EM detection

[4] https://github.com/sunny1999123/MasterThesis.

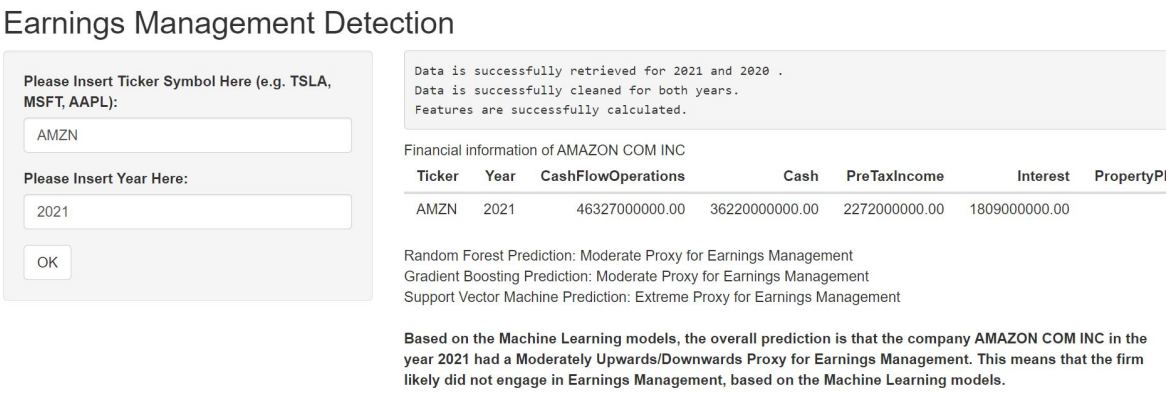
hyperparameters are tuned, the appropriate model is chosen, and the model is applied to the test set after which the metrics are collected.

5.4 New firm analysis

Whilst the previous paragraph provides an summarized overview of all the relevant code, it does not enable to investors to analyze a specific firm of their interest. Therefore, using ...., investors can also analyze whether the models used in this research would predict a moderate or extreme EM proxy for their firm of interest. This can help investors in their decision making whether or not a firm should be included in their investment decisions. The goal of this tool is to increase the practical usefulness of this study for retail investors.

The tool can be accessed via `https://sunny1999.shinyapps.io/EMDetection/`. When the tool is opened, an investor will see two input fields. The first input field serves for the ticker symbol of the firm of interest for the investor [5]. The second input should consist of the year that the investor is interested in [6]. After the ticker symbol and year are inserted, an investor should click on the "OK" button to start the tool. This will trigger several steps that will lead to the final recommendation. First, based on the input, the 10-K financial statement will be retrieved. Next, the data will be cleaned and the features will be calculated. After this, the machine learning models are trained on the full dataset that is used in this research with the same hyperparameters. Finally, a prediction of all three machine learning models of the input will be shown. This includes a final prediction whether or not, based on this study, a firm should (not) be included in a investor's portfolio. An screenshot on how the output would look for the input AMZN and 2021 is shown below.

Figure 8: Earnings Management Tool
The figure shows an example of the output of the EM detection tool.



[5] If the ticker symbol of the company of interest is unknown, an investor could use `https://www.sec.gov/edgar/searchedgar/companysearch` to find the corresponding ticker symbol

[6] As the XBRL data quality has drastically increased since the introduction of inline XBRL in 2018, there is low probability that financial statements before this year can be retrieved without any errors.

6 Discussion

6.1 Main Findings

6.2 Discussion

6.3 Limitation

6.4 Limitations

6.5 Further Research

6.6 Conclusion

7 Appendices

7.1 Features

Table 6: All Features

The table shows all the features that are used to predict the AEM proxy, and how they are calculated.

	Feature	Calculation
1	Accounts Receivable Turnover	$\frac{\text{Revenues}}{\text{Accounts Receivable}}$
2	Current Ratio	$\frac{\text{Current Assets}}{\text{Current Liabilities}}$
3	Debt-to-Equity	$\frac{\text{Debt}}{\text{Equity}}$
4	Return on Assets (ROA)	$\frac{\text{Net Income}}{\text{Assets}}$
5	Return on Equity (ROE)	$\frac{\text{Net Income}}{\text{Equity}}$
6	Days Sales in Accounting Receivable	$\frac{1}{360} * \frac{\text{Revenue}}{\text{Accounts Receivable}}$
7	Depreciation over Plant	$\frac{\text{Depreciation}}{\text{Plant}}$
8	Equity over Fixed Assets	$\frac{\text{Equity}}{\text{Fixed Assets}}$
9	Times Interest Earned	$\frac{\text{EBITDA}}{\text{Interest}}$
10	Sales to Total Assets	$\frac{\text{Revenues}}{\text{Assets}}$
11	Pre-Tax Income to Sales	$\frac{\text{Pre-Tax Income}}{\text{Sales}}$
12	Net Income/Loss to Sales	$\frac{\text{Net Income}}{\text{Sales}}$
13	Sales to Cash	$\frac{\text{Sales}}{\text{Cash}}$
14	Sales to Working Capital	$\frac{\text{Sales}}{\text{Current Assets- Current Liabilities}}$
15	Sales to Fixed Assets	$\frac{\text{Sales}}{\text{Fixed Assets}}$
16	Working Capital to Assets	$\frac{\text{Current Assets- Current Liabilities}}{\text{Assets}}$
17	EBITDA Margin Ratio	$\frac{\text{EBITDA}}{\text{Revenues}}$
18	Cash Flow Operations to Debt	$\frac{\text{Cash Flow from Operations}}{\text{Debt}}$
19	Net Income/Loss to Cash Flow	$\frac{\text{Net Income}}{\text{Cash Flow from Operations}}$
20	Change in Depreciation	$\text{Depreciation}_t - \text{Depreciation}_{t-1}$
21	Change in Assets	$\text{Assets}_t - \text{Assets}_{t-1}$
22	Change in Revenues	$\text{Revenues}_t - \text{Revenues}_{t-1}$
23	Change in Current Ratio	$\text{Current Ratio}_t - \text{Current Ratio}_{t-1}$
24	Change in Debt-to-Equity	$\text{Debt-to-Equity}_t - \text{Debt-to-Equity}_{t-1}$
25	Change in Working Capital	$\text{Working Capital}_t - \text{Working Capital}_{t-1}$
26	Change in Days Sales in Accounting Receivable	$\text{Days Sales in Accounting Receivable}_t - \text{Days Sales in Accounting Receivable}_{t-1}$
27	Change in Depreciation over Plant	$\text{Depreciation over Plant}_t - \text{Depreciation over Plant}_{t-1}$
28	Change in Equity over Fixed Assets	$\text{Equity over Fixed Assets}_t - \text{Equity over Fixed Assets}_{t-1}$
29	Change in Times Interest Earned	$\text{Times Interest Earned}_t - \text{Times Interest Earned}_{t-1}$
30	Change in Sales to Total Assets	$\text{Sales to Total Assets}_t - \text{Sales to Total Assets}_{t-1}$
31	Change in Pre-Tax Income to Sales	$\text{Pre-Tax Income to Sales}_t - \text{Pre-Tax Income to Sales}_{t-1}$
32	Change in Net Income to Sales	$\text{Net Income to Sales}_t - \text{Net Income to Sales}_{t-1}$
33	Change in Sales to Working Capital	$\text{Sales to Working Capital }_t - \text{Sales to Working Capital }_{t-1}$
34	Change in Working Capital to Assets	$\text{Working Capital to Assets}_t - \text{Working Capital to Assets}_{t-1}$
35	Change in EBITDA Margin Ratio	$\text{EBITDA Margin Ratio }_t - \text{EBITDA Margin Ratio }_{t-1}$
36	Change in Debt	$\text{Debt}_t - \text{Debt}_{t-1}$

Table 7: Variable Identifiers

This table shows all the labels that are part of each financial statement item

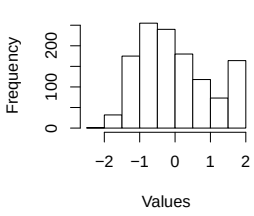
Financial statement Item	Label
Revenue	*us-gaap:RevenueFromContractWithCustomerExcludingAssessedTax*
	us-gaap:ContractWithCustomerLiabilityRevenueRecognized
	us-gaap:DeferredRevenueCurrent
	us-gaap:SalesRevenueNet
Accounts Receivable	*us-gaap:AccountsReceivableNetCurrent*
	us-gaap:ReceivablesNetCurrent
	us-gaap:AccountsAndOtherReceivablesNetCurrent
Current Assets	*us-gaap:AssetsCurrent*
	us-gaap:OtherAssetsCurrent
Current Liabilities	*us-gaap:LiabilitiesCurrent*
	us-gaap:OtherLiabilitiesCurrent
	us-gaap:OtherAccruedLiabilitiesCurrent
	us-gaap:ContractWithCustomerLiabilityCurrent
Inventory	*us-gaap:InventoryNet*
	us-gaap:InventoryFinishedGoodsNetOfReserves
	us-gaap:InventoryFinishedGoods
	us-gaap:InventoryGross
	us-gaap:LIFOInventoryAmount
	us-gaap:InventorySuppliesNetOfReserves
Equity	*us-gaap:StockholdersEquity*
	us-gaap:StockholdersEquityIncludingPortionAttributableToNoncontrollingInterest
	us-gaap:StockholdersEquityOther
NetIncome	*us-gaap:NetIncomeLoss*
	us-gaap:ComprehensiveIncomeNetOfTax
	us-gaap:OperatingIncomeLoss
	us-gaap:IncomeLossFromContinuingOperationsBeforeIncomeTaxesForeign
	us-gaap:IncomeLossFromContinuingOperationsBeforeIncomeTaxesDomestic
	us-gaap:IncomeLossFromContinuingOperationsBeforeIncomeTaxesExtraordinaryItemsNoncontro
Assets	*us-gaap:Assets*
Cost of Goods Sold	*us-gaap:CostOfGoodsAndServicesSold*
	us-gaap:CostOfRevenue
Depreciation	*us-gaap:DepreciationDepletionAndAmortization*
	us-gaap:Depreciation
	us-gaap:DepreciationAndAmortization
	us-gaap:DepreciationAmortizationAndAccretionNet
	us-gaap:DepreciationNonproduction
Plant	*us-gaap:PropertyPlantAndEquipmentNet*
	us-gaap:PropertyPlantAndEquipmentGross
	us-gaap:PropertyPlantAndEquipmentAdditions
	us-gaap:PropertyPlantAndEquipmentOther
	us-gaap:PropertyPlantAndEquipmentDisposals
	us-gaap:PropertyPlantAndEquipmentAndFinanceLeaseRightOfUseAssetAfterAccumulatedDeprec
	us-gaap:PropertyPlantAndEquipmentAndFinanceLeaseRightOfUseAssetAfterAccumulatedDeprec
Long-term Debt	*us-gaap:LongTermDebtNoncurrent*
	us-gaap:LongTermDebt
	us-gaap:LongTermDebtAndCapitalLeaseObligations
	us-gaap:LongTermDebtFairValue
	us-gaap:OtherLongTermDebt
	us-gaap:UnsecuredLongTermDebt
Fixed Assets	*us-gaap:NoncurrentAssets*
	us-gaap:AssetsNoncurrent
Interest	*us-gaap:InterestExpense*
	us-gaap:InterestPaidNet
	us-gaap:InterestPaid
	us-gaap:InterestExpenseOther
Pre-tax Income	*us-gaap:IncomeLossFromContinuingOperationsBeforeIncomeTaxesForeign*
	us-gaap:IncomeLossFromContinuingOperationsBeforeIncomeTaxesMinorityInterestAndIncomeL
	us-gaap:IncomeLossFromContinuingOperationsBeforeIncomeTaxesExtraordinaryItemsNoncontro
Cash	*us-gaap:CashAndCashEquivalentsAtCarryingValue*
	us-gaap:CashCashEquivalentsRestrictedCashAndRestrictedCashEquivalentsPeriodIncreaseDecrea
	us-gaap:CashCashEquivalentsRestrictedCashAndRestrictedCashEquivalents
	CashEquivalentsAtCarryingValue
Cash flow from Operations	*us-gaap:NetCashProvidedByUsedInOperatingActivities*
	us-gaap:NetCashProvidedByUsedInOperatingActivitiesContinuingOperations
	us-gaap:CashProvidedByUsedInOperatingActivitiesDiscontinuedOperations

7.2 Variable Identifiers

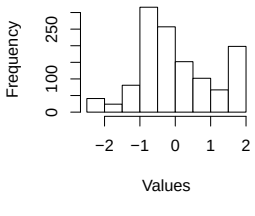
7.3 Feature Distribution

Figure 9: Feature Distribution

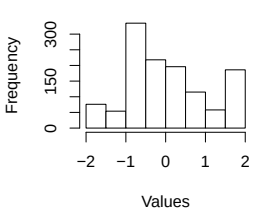
The figure shows the distribution of all the features that are used in this analysis. All the features are winsorized and normalized.



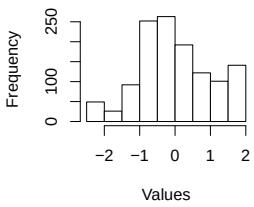
(i) Current Ratio



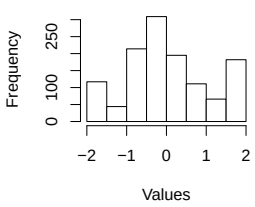
(ii) Accounts Receivable Turnover



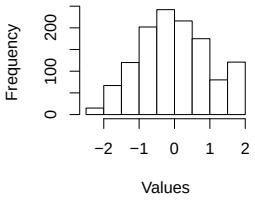
(iii) DebtToEquity



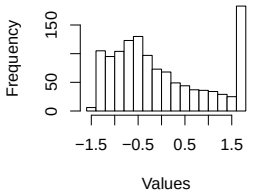
(iv) ROA



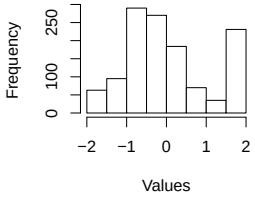
(v) ROE



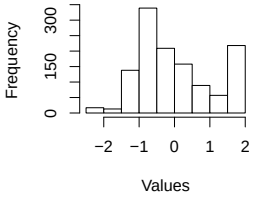
(vi) DaysSalesinAccountingReceivable



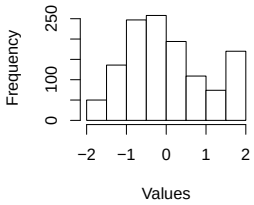
(vii) DepreciationOverPlant



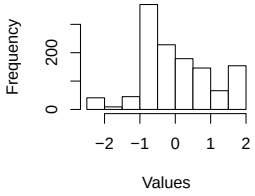
(viii) EquityOverFixedAssets



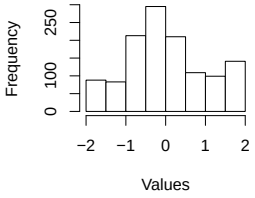
(ix) TimesInterestEarned



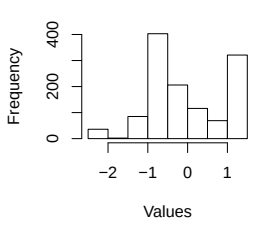
(x) SalesToTotalAssets



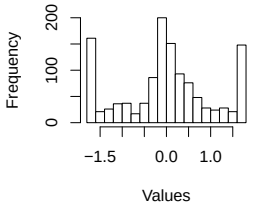
(xi) PreTaxIncomeToSales



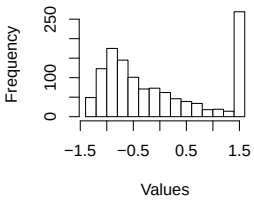
(xii) NetIncomeLossToSales



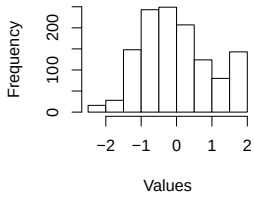
(xiii) SalestoCash



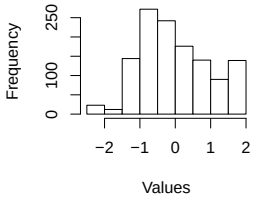
(xiv) SalestoWorking-Capital



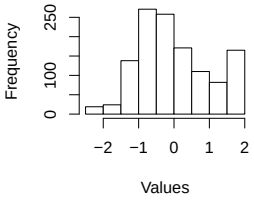
(xv) SalesToFixedAssets



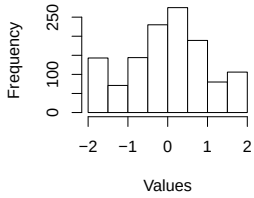
(xvi) WorkingCaital-ToAssets



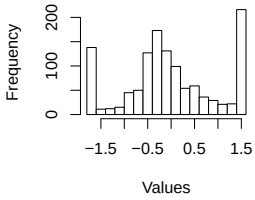
(xvii) EBITDAMargin-Ratio



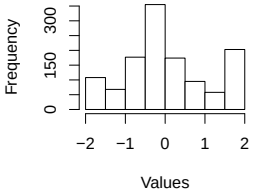
(xviii) CashFlowOperationsToDebt



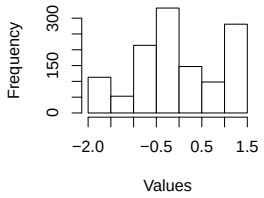
(xix) NetIncomeLossTo-Cashflow



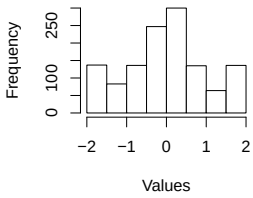
(xx) ChangeInDepreciation



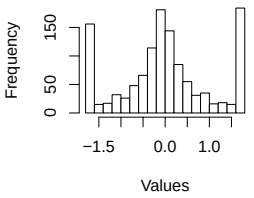
(xxi) ChangeInAssets



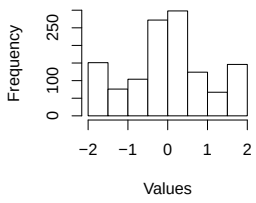
(xxii) ChangeInRevenues



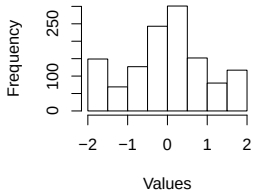
(xxiii) ChangeInCurrentRatio



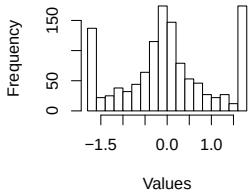
(xxiv) ChangeInDebt-ToEquity



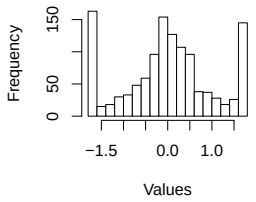
(xxv) ChangeInWorkingCapital



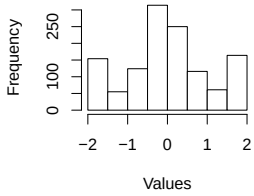
(xxvi) ChangeInDaysSalesinAccountingReceivable



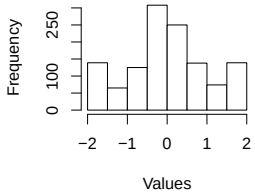
(xxvii) ChangeInDepreciationOverPlant



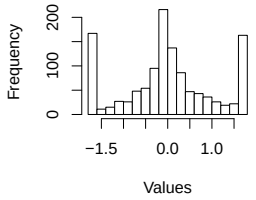
(xxviii) ChangeInEquityOverFixedAssets



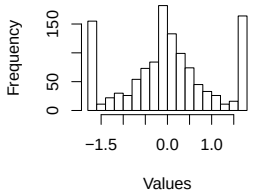
(xxix) ChangeInTimesInterestEarned



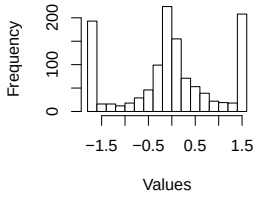
(xxx) ChangeInSalesToTotalAssets



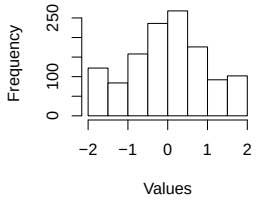
(xxxi) ChangeInPreTaxIncomeToSales



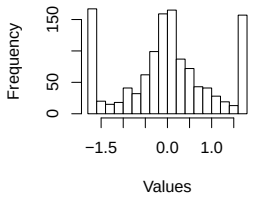
(xxxii) ChangeInNetIncomeLossToSales



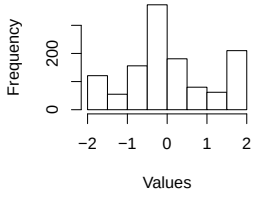
(xxxiii) ChangeInSalestoWorkingCapital



(xxxiv) ChangeInWorkingCaitalToAssets



(xxxv) ChangeInEBITDAMarginRatio



(xxxvi) ChangeInDebt

7.4 Estimates Lasso regularization

Table 8: Estimates Lasso Regularization

The table shows the estimates based on the lasso regression. Six variables are shrunk to zero, and will not be used in the machine learning models

Term	Penalty	Estimate
Intercept	0.003635851	-1.712
WorkingCapitalToAssets	0.003635851	0.684
CashFlowOperationsToDebt	0.003635851	0.634
ChangeInWorkingCapitalToAssets	0.003635851	-0.494
ChangeInSalesToWorkingCapital	0.003635851	0.355
ChangeInCurrentRatio	0.003635851	0.309
SalesToWorkingCapital	0.003635851	-0.275
NetIncomeLossToSales	0.003635851	-0.273
SalestoCash	0.003635851	-0.254
ChangeInPreTaxIncomeToSales	0.003635851	0.230
CurrentRatio	0.003635851	-0.212
ChangeInDepreciationOverPlant	0.003635851	-0.181
TimesInterestEarned	0.003635851	-0.170
DepreciationOverPlant	0.003635851	0.168
ChangeInWorkingCapital	0.003635851	0.156
AccountsReceivableTurnover	0.003635851	0.150
DebtToEquity	0.003635851	0.134
SalesToFixedAssets	0.003635851	-0.129
SalesToTotalAssets	0.003635851	-0.091
ChangeInTimesInterestEarned	0.003635851	-0.070
ChangeInDepreciation	0.003635851	0.061
ChangeInNetIncomeLossToSales	0.003635851	0.056
ChangeInDebt	0.003635851	0.052
ROA	0.003635851	0.050
ChangeInEquityOverFixedAssets	0.003635851	-0.048
DaysSalesinAccountingReceivable	0.003635851	-0.041
ChangeInSalesToTotalAssets	0.003635851	0.029
EBITDAMarginRatio	0.003635851	-0.028
ROE	0.003635851	0.024
ChangeInEBITDAMarginRatio	0.003635851	0.004
ChangeInDebtToEquity	0.003635851	0.001
EquityOverFixedAssets	0.003635851	0.000
PreTaxIncomeToSales	0.003635851	0.000
PreTaxIncomeToSales	0.003635851	0.000
ChangeInAssets	0.003635851	0.000
ChangeInRevenues	0.003635851	0.000
ChangeInDaysSalesinAccountingReceivable	0.003635851	0.000

References

Aduda, Josiah, Morgan Ongoro, et al. (2020). "Working capital and earnings management among manufacturing firms: A review of literature". In: *J. Financ. Invest. Anal* 9, pp. 71–79.

Ahmi, Aidi and Mohd Herry Mohd Nasir (2019). "Examining the trend of the research on extensible business reporting language (XBRL): A bibliometric review". In: *International journal of innovation, creativity and change* 5.2, pp. 1145–1167.

Almahrog, Yousf Ebrahem and Alhashmi Aboubaker Lasyoud (2021). "An Overview of Earnings Management Detection Approaches". In: *Journal of critical reviews* 8.02, pp. 92–101.

Almaqtari, Faozi A. et al. (2021). "Earning management estimation and prediction using machine learning: A systematic review of processing methods and synthesis for future research". In: Institute of Electrical and Electronics Engineers Inc., pp. 291–298. ISBN: 9781665420877. DOI: 10.1109/ICTAI53825.2021.9673157.

Bajra, Ujkan and Simon Cadez (2018). "The impact of corporate governance quality on earnings management: Evidence from European companies cross-listed in the US". In: *Australian Accounting Review* 28.2, pp. 152–166.

Bartley, Jon, Al Y S Chen, and Eileen Z Taylor (2011). "A comparison of XBRL filings to corporate 10-Ks—Evidence from the voluntary filing program". In: *Accounting Horizons* 25.2, pp. 227–245.

Beneish, Messod D (2001). "Earnings management: A perspective". In: *Managerial finance* 27.12, pp. 3–17.

Blankespoor, Elizabeth, Brian P Miller, and Hal D White (2014). "Initial evidence on the market impact of the XBRL mandate". In: *Review of Accounting Studies* 19, pp. 1468–1503.

Campa, Domenico (Dec. 2019). "Earnings management strategies during financial difficulties: A comparison between listed and unlisted French companies". In: *Research in International Business and Finance* 50, pp. 457–471. ISSN: 02755319. DOI: 10.1016/j.ribaf.2019.07.001.

Chen, Fu Hsiang, Der Jang Chi, and Yi Cheng Wang (Apr. 2015). "Detecting biotechnology industry's earnings management using Bayesian network, principal component analysis, back propagation neural network, and decision tree". In: *Economic Modelling* 46, pp. 1–10. ISSN: 02649993. DOI: 10.1016/j.econmod.2014.12.035.

Chen, Fu Hsiang and Hu Howard (May 2016). "An alternative model for the analysis of detecting electronic industries earnings management using stepwise regression, random forest, and decision tree". In: *Soft Computing* 20 (5), pp. 1945–1960. ISSN: 14337479. DOI: 10.1007/s00500-015-1616-6.

Chen, Yuh Jen et al. (Mar. 2019). "Fraud detection for financial statements of business groups". In: *International Journal of Accounting Information Systems* 32, pp. 1–23. ISSN: 14670895. DOI: 10.1016/j.accinf.2018.11.004.

Chychyla, Roman and Alexander Kogan (2015). "Using XBRL to conduct a large-scale study of discrepancies between the accounting numbers in Compustat and SEC 10-K filings". In: *Journal of Information Systems* 29.1, pp. 37–72.

Dbouk, B. (2017). "Financial Statements Earnings Manipulation Detection Using a Layer of Machine Learning". In: *International Journal of Innovation, Management and Technology*, pp. 172–179. ISSN: 20100248. DOI: 10.18178/ijimt.2017.8.3.723.

Debreceny, Roger et al. (2010). "Does it add up? Early evidence on the data quality of XBRL filings to the SEC". In: *Journal of Accounting and Public Policy* 29.3, pp. 296–306.

Dechow, Patricia M et al. (2012). "Detecting earnings management: A new approach". In: *Journal of accounting research* 50.2, pp. 275–334.

Diana, Balaciu and Pop Cosmina Madalina (2007). "Is creative accounting a form of manipulation". In: *Economic Science Series, Annals of the University of Oradea* 17.3, pp. 935–940.

Du, Hui, Miklos A. Vasarhelyi, and Xiaochuan Zheng (June 2013). "XBRL Mandate: Thousands of Filing Errors and So What?" In: *Journal of Information Systems* 27.1, pp. 61–78. ISSN: 0888-7985. DOI: 10.2308/isys-50399. eprint: https://publications.aaahq.org/jis/article-pdf/27/1/61/11031/isys-50399.pdf. URL: https://doi.org/10.2308/isys-50399.

Efendi, Jap, Jin Dong Park, and L Murphy Smith (2014). "Do XBRL filings enhance informational efficiency? Early evidence from post-earnings announcement drift". In: *Journal of Business Research* 67.6, pp. 1099–1105.

Franz, Diana R, Hassan R HassabElnaby, and Gerald J Lobo (2014). "Impact of proximity to debt covenant violation on earnings management". In: *Review of Accounting Studies* 19, pp. 473–505.

Gareth, James et al. (2013). *An introduction to statistical learning: with applications in R*. Spinger.

Guo, Ken H and Xiaoxiao Yu (2022). "Retail Investors Use XBRL Structured Data? Evidence from the SEC's Server Log". In: *Journal of Behavioral Finance* 23.2, pp. 166–174.

Hammami, Ahmad and Mohammad Hendijani Zadeh (Dec. 2022). "Predicting earnings management through machine learning ensemble classifiers". In: *Journal of Forecasting* 41 (8), pp. 1639–1660. ISSN: 1099131X. DOI: 10.1002/for.2885.

Healy, Paul M. and Krishna G. Palepu (2003). "The Fall of Enron". In: *Journal of Economic Perspectives* 17.2, pp. 3–26. DOI: 10.1257/089533003765888403. URL: https://www.aeaweb.org/articles?id=10.1257/089533003765888403.

Höglund, Henrik (2012). "Detecting earnings management with neural networks". In: *Expert systems with applications* 39.10, pp. 9564–9570.

Hoitash, Rani, Udi Hoitash, and Landi Morris (2021). "eXtensible Business Reporting Language (XBRL): a review and implications for future research". In: *Auditing: A Journal of Practice & Theory* 40.2, pp. 107–132.

Iatridis, George and George Kadorinis (2009). "Earnings management and firm financial motives: A financial investigation of UK listed firms". In: *International Review of Financial Analysis* 18.4, pp. 164–173.

Jackson, Andrew B. (June 2018). "Discretionary Accruals: Earnings Management.. or Not?" In: *Abacus* 54 (2), pp. 136–153. ISSN: 14676281. DOI: 10.1111/abac.12117.

Janvrin, Diane J, Robert E Pinsker, and Maureen Francis Mascha (2013). "XBRL-enabled, spreadsheet, or PDF? Factors influencing exclusive user choice of reporting technology". In: *Journal of information systems* 27.2, pp. 35–49.

Jiraporn, Pornsit et al. (2008). "Is earnings management opportunistic or beneficial? An agency theory perspective". In: *International Review of Financial Analysis* 17.3, pp. 622–634.

Johnston, Joseph (2020). "Extended xbrl tags and financial analysts' forecast error and dispersion". In: *Journal of Information Systems* 34 (3), pp. 105–131. ISSN: 15587959. DOI: 10.2308/ISYS-16-013.

Kassem, Rasha (2012). "Earnings management and financial reporting fraud: can external auditors spot the difference?" In: *American Journal of Business and management* 1.1, pp. 30–33.

Kim, Jeong-Bon, Joung W Kim, and Jee-Hae Lim (2019). "Does XBRL adoption constrain earnings management? Early evidence from mandated US filers". In: *Contemporary Accounting Research* 36.4, pp. 2610–2634.

Kothari, S. P., Andrew J. Leone, and Charles E. Wasley (Feb. 2005). "Performance matched discretionary accrual measures". In: *Journal of Accounting and Economics* 39 (1), pp. 163–197. ISSN: 01654101. DOI: 10.1016/j.jacceco.2004.11.002.

Liu, Mingzhi et al. (2017). "Does family involvement explain why corporate social responsibility affects earnings management?" In: *Journal of business research* 75, pp. 8–16.

Mayapada, Arung Gihna, Muhammad Afdhal, and Rahmi Syafitri (2020). "Earnings management in the pre and post eXtensible business reporting language period in Indonesia". In: *The Indonesian Journal of Accounting Research* 23.1, pp. 29–48.

Palas, A (2019). *Earning movement prediction using machine learning-Support Vector Machines (SVM)*, pp. 36–53.

Peng, Emma Yan, John Shon, and Christine Tan (2011). "XBRL and accruals: Empirical evidence from China". In: *Accounting Perspectives* 10.2, pp. 109–138.

Perdana, Arif, Alastair Robb, and Fiona Rohde (Mar. 2015). "An Integrative Review and Synthesis of XBRL Research in Academic Journals". In: *Journal of Information Systems* 29.1, pp. 115–153. ISSN: 0888-7985. DOI: 10 .2308 / isys -50884. eprint: https : //publications.aaahq.org/jis/article-pdf/29/1/115/8685/isys-50884.pdf. URL: https://doi.org/10.2308/isys-50884.

– (2019). "Textual and contextual analysis of professionals' discourses on XBRL data and information quality". In: *International Journal of Accounting & Information Management* 27.3, pp. 492–511.

Perols, Johan L and Barbara A Lougee (2011). "The relation between earnings management and financial statement fraud". In: *Advances in Accounting* 27.1, pp. 39–53.

Rahul, Kumar, Nandini Seth, and U. Dinesh Kumar (2018). "Spotting earnings manipulation: Using machine learning for financial fraud detection". In: vol. 11311 LNAI. Springer Verlag, pp. 343–356. ISBN: 9783030041908. DOI: 10.1007/978-3-030-04191-5_29.

Rao, Yanchao and Ken Huijin Guo (Feb. 2022). "Does XBRL help improve data processing efficiency?" In: *International Journal of Accounting and Information Management* 30 (1), pp. 47–60. ISSN: 17589037. DOI: 10.1108/IJAIM-07-2021-0155.

Sanad, Zakeya (May 2021). "Machine Learning and Earnings Management Detection". In: pp. 77–83. ISBN: 978-3-030-73056-7. DOI: 10.1007/978-3-030-73057-4_6.

SEC (2009). *"Interactive data to improve financial reporting. U.S. Securities and exchange commission.* https://www.sec.gov/rules/final/2009/33-9002.pdf.

Tallapally, Prabhakar, Michael S. Luehlfing, and Marianne Motha (2011). "The Partnership Of EDGAR Online And XBRL - Should Compustat Care?" In: *Review of Business Information Systems (RBIS)* 15.4, pp. 39–46. DOI: 10.19030/rbis.v15i4.6011.

Tsai, Chih Fong and Yen Jiun Chiou (2009). "Earnings management prediction: A pilot study of combining neural networks and decision trees". In: *Expert Systems with Applications* 36 (3 PART 2), pp. 7183–7191. ISSN: 09574174. DOI: 10.1016/j.eswa.2008.09.025.

Yaghoobirafi, Kamaleddin and Eslam Nazemi (2019). "An approach to XBRL interoperability based on Ant Colony Optimization algorithm". In: *Knowledge-Based Systems* 163, pp. 342–357.

Zhang, Yanan, Yuyan Guan, and Jeong Bon Kim (Jan. 2019). "XBRL adoption and expected crash risk". In: *Journal of Accounting and Public Policy* 38 (1), pp. 31–52. ISSN: 18732070. DOI: 10.1016/j.jaccpubpol.2019.01.003.