Predicting Earnings Management using XBRL through Machine Learning Models

Master Thesis in Business Analytics & Management

Rotterdam School of Management
Erasmus University Rotterdam

Apoorv Sunny Bhatia
590913

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

Date: June 15, 2023

Preface

Executive summary

Earnings Management (EM) refers to earnings-manipulative practices within the accepted accounting principles, which can mislead investors, regulators, and other stakeholders. Its detection and prevention are essential to ensure that financial statement users can make decisions based on correct information.

This study will evaluate several machine learning models to analyse whether financial statement information can be used to detect earnings management. This will be achieved using a 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, analyse, and validate financial statements faster and more accurately than using databases.

This study has two main findings. First, it has shown that financial statement information has good predictive power in detecting EM. The random forest model achieved the best performance across all used models. Furthermore, this study has shown that XBRL can be a good source for financial statements in terms of correctness and accessibility. However, it also has its disadvantages when it comes to the technical skills that it needs.

As this master thesis focuses on an investor's perspective, it attempts to perform a study that investors can replicate. This is achieved in two different ways. First, all the code used in this study can be accessed via the following GitHub repository: `https://github.com/sunny1999123/MasterThesis`. Furthermore, an EM detection tool is developed (`https://sunny1999.shinyapps.io/EMDetectionTool/`) that allows stakeholders to analyse a firm of their interest.

Contents

List of Tables

List of Figures

1 Introduction

1.1 Problem Background

All publicly listed companies must publish financial statements which accurately represent 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, leading to its bankruptcy (Campa, 2019). The bankruptcy was mainly because Enron engaged in earnings management (EM).

EM refers to the manipulation of financial statements using accounting practices to achieve desired outcomes (Beneish, 2001). Firms' management may engage in EM for various reasons, such as to meet earnings targets or to influence the share price. 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, their main difference is that EM is within the bounds of the General Accepted Accounting Principles (GAAP), whereas fraud is not (Perols and Lougee, 2011). Since investors rely on financial statements for their investment choices (Chen et al., 2019), firms that engage in EM could mislead investors.

According to Dbouk (2017), the current state-of-the-art regarding the detection of EM in practice, which includes manual verification of general ledger accounts, the use of simple ratios, and inventory counts, need to be revised and updated. Furthermore, prior research on EM mainly focused on the association between certain factors and EM (Chen, Chi, and Wang, 2015). The recent technological development in business enables using advanced technologies for EM detection (Sanad, 2021). Several recent studies attempted to incorporate machine learning techniques in detecting EM. Chen, Chi, and Wang (2015) studied whether a set of financial statement items could predict the discretionary accruals of a company, which are used as a proxy for EM. More recent research conducted by Hammami and Zadeh (2022) used various feature selection techniques in combination with a Support Vector Machine to assess its performance in predicting EM. Still, Almaqtari et al. (2021) argues that there is limited research on predicting EM using machine learning, while the field seems promising.

One of the components in the field that is yet to be explored when it comes to EM detection is how investors could assess EM. As mentioned, 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. Still, investors lack the professional skills to assess a firm's earnings quality (Sun, Wang, and Xiao, 2022). Even though the auditor is responsible for this task, investors can have personal risk preferences. Therefore, this study will focus on how investors can assess EM.

A drawback for (retail) investors regarding the current research on EM detection is that it is conducted using databases (Hammami and Zadeh, 2022; Chen, Chi, and Wang, 2015). However, databases as COMPUSTAT demand a subscription, which reduces the accessibility

for investors (Palas, 2019). Furthermore, prior research by Tallapally, Luehlfing, and Motha (2011) has shown that financial statement information retrieved from COMPUSTAT can differ from the original information. So, a different data retrieval method in EM detection is desirable for investors.

This study focuses on a new approach in EM detection research, which is the use of eXtensible Business Reporting Language (XBRL). This free software is XML language based and is used to exchange business information, such as financial statements. In the setting of this study, XBRL could enable investors to analyse financial statements for EM detection, as these can be retrieved via XBRL without needing a subscription to a database. Prior research has studied XBRL in several settings, 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 solve the challenges regarding data accessibility for investors. This study attempts to fill this gap by exploring whether XBRL can be used to detect EM. Therefore, this research aims to detect EM using XBRL through 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 desirable 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 four research questions:

- How can EM be proxied using XBRL?
- Which financial statement features contain relevant information that can be used in EM detection?
- Which machine learning model achieves the highest sensitivity while maintaining good accuracy, specificity, and ROC-AUC 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 which companies have likely engaged in EM. Since there is a positive relationship between fraud and ex-ante EM (Perols and Lougee, 2011), the detection of EM is crucial for investors. Current accounting practices that examine EM are limited to outdated methods such as manual verification of accounts and the calculation of simple ratios (Dbouk, 2017). Hence, it is essential to integrate current technological developments into the EM literature (Almaqtari

et al., 2021). This study attempts to achieve this by applying machine learning models on financial statements, which are retrieved using XBRL, to detect EM accurately. As XBRL is freely accessible, and an elaboration on how investors can use the results of this study for their analysis will be discussed in Chapter 5, this study will enable investors to use XBRL in EM detection. This makes this study relevant for small retail investors and larger institutions who want to analyse their portfolios. Therefore, the results of this study should enable stakeholders 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 on EM, machine learning, and XBRL and attempts to fill specific gaps in the literature. First, this research adds to prior research by Chen, Chi, and Wang (2015) and Hammami and Zadeh (2022), who conducted a similar study on EM detection. However, instead of using databases or collecting data manually, this study will use XBRL to retrieve the data. This could lead to new insights into whether XBRL can be used as an alternative to traditional data retrieval methods.

Furthermore, unlike other EM detection studies, this study uses features that are important determinants in earnings movement (Palas, 2019)[1]. Since the set of features seems to be essential determinants in earnings movement prediction, it could potentially mean that they are essential for earnings management prediction. This could lead to new insights into which features are important in EM prediction. Another gap-filling addition of this research is that it will incorporate its findings in a GitHub repository and Shiny WebApp, which are further elaborated in Chapter 5. This makes the results of this study replicable by stakeholders who want to replicate this study using a different set of firms or want to analyse a specific firm.

1.5 Thesis Overview

The subsequent sections of this thesis have the following structure. Chapter 2 presents a review of the existing literature concerning XBRL and EM. 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. Chapter 5 discusses how investors and other stakeholders could use this research by providing insights into the GitHub repository and Shiny WebApp. Lastly, Chapter 6 provides the concluding remarks of this study, including a discussion of the main findings of this study, its limitations, and recommendations for future research.

[1]The list that (Palas, 2019) used is slightly modified, as some financial statement items were too industry-specific

2 Theoretical Background

2.1 Earnings Management

This section discusses the existing literature on earnings management (EM). Firstly, EM will be defined and distinguished from fraud. Next, the types of EM are outlined. Finally, prior literature on EM detection is examined.

2.1.1 Introduction to Earnings Management

Financial statements serve as an essential 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 their earnings. Such practices could mislead stakeholders and reduce their trust in financial statements. Therefore, detecting and preventing these practices, referred to as 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 most prominent researchers in the field of EM is Messod D. Beneish, who defined EM as a deliberate process within the generally accepted accounting principles (GAAP) to achieve a desired level of reported earnings (Beneish, 2001). Prior literature has shown mixed results regarding whether the practice of EM is ethical. While some papers argue that EM should be considered ethical (Jiraporn et al., 2008; Diana and Madalina, 2007; Bajra and Cadez, 2018), as it is in line with the GAAP, others argue that EM is unethical (Beneish, 2001; Healy and Palepu, 2003) since it can mislead stakeholders. This study follows the second literature stream, which advocates that EM is unethical.

It is important to notice the difference between EM and fraud. While fraud and EM have the same objective, their most essential difference is that fraud is not within the GAAP, whereas EM is within the bounds of the GAAP (Perols and Lougee, 2011). This difference entails that it is easier to label fraud than EM (Kassem, 2012), as fraud can be tested against the GAAP. Still, the detection and prevention of EM are crucial to ensure investors are not misled (Kliestik et al., 2021). On top of that, Perols and Lougee (2011) argue that companies that have had significant frauds engaged in EM practices in the years before the fraud.

Firms can 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). Violating these covenants could push a firm into 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). These schemes could be related to the reported earnings in financial statements. Therefore, managers might feel incentivized to manage their earnings to meet specific bonus

targets. Despite its motive, EM practices should be prevented to ensure investors can make well-informed decisions. The following subsection elaborates on the various types of EM.

2.1.2 Types of Earnings Management

The literature has categorized EM into two types; accrual-based earnings management (AEM) and real earnings management (REM) (Cohen and Zarowin, 2010). AEM is related to activities that are carried out using accounting practices. An example of AEM is the change of pricing of the inventory or depreciation method of the fixed assets (Huang and Sun, 2017). REM is related to practices influencing the company's cash flow (Darmawan, Sutrisno, and Mardiati, 2019). An example of REM is the change of advertising costs to meet specific targets (Cohen and Zarowin, 2010). Despite the type of EM, firms engaging in either can mislead investors, making its detection crucial. Since this study will retrieve financial statements, which are a product of accounting practices, it will solely focus on AEM.

One of the most used models in prior studies on AEM is the Jones Model (Jones, 1991), which several researchers have later adjusted. Examples are the modified Jones Model (Dechow, Sloan, and Sweeney, 1995) and the performance-adjusted Jones Model (Kothari, Leone, and Wasley, 2005). Still, all models are based on the same principle of discretionary accruals. According to Jones (1991), the total accruals of a firm, defined as the change in current assets minus the change in current liabilities, can be split into two categories; non-discretionary accruals and discretionary accruals. Non-discretionary accruals are those accruals that are a consequence of economic developments, whereas discretionary accruals are due to managers' actions. Therefore, the discretionary accruals extracted from the Jones Model are often used as a proxy for earnings management. Still, there is also critique on the use of (a form of) the Jones Model. Jackson (2018) argues that accrual-based earnings management models are unrelated to ex-post known manipulation cases. However, Larson, Sloan, and Zha Giedt (2018) found a relationship between discretionary accruals and Accounting and Auditing Enforcement Releases (AAER), which are known fraud cases.

Furthermore, McNichols and Stubben (2018) argue that discretionary accruals are a noisy proxy for EM, which entails that it does not provide an adequate basis for inferences regarding EM. This is also motivated by Höglund (2012), who argues that most accrual-based EM studies use linear models, while EM is a non-linear phenomenon. This study evaluates several machine learning models that can capture non-linear relationships. Therefore, this study argues that discretionary accruals can be used as a proxy for earnings management. The specific method used in this research to calculate the discretionary accruals is outlined in 3.2.1. The following subsection discusses prior research on the detection of EM.

2.1.3 Detection of Earnings Management

Prior research in Em has focused on the correlation between earnings management and other factors (Chen, Chi, and Wang, 2015). For example, Ruwanti, Chandrarin, and Assih (2019) found a positive relationship between Corporate Social Responsibility (CSR) and earnings management. Still, only a few studies have performed research on EM detection. Furthermore, these studies often used conventional models, e.g., regression models and logit analysis

(Chen, Chi, and Wang, 2015). However, these models require the data to follow specific properties, such as normality and linearity, which are often invalid with financial data. It is important to shift the focus to using non-linear machine learning models for EM detection (Almaqtari et al., 2021).

A recent study that incorporated machine learning in EM detection was conducted by Hammami and Zadeh (2022). They used several feature selection methods in combination with a support vector machine to predict the discretionary accruals. They used a set of financial statement ratios as features. One of their 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, as it shows that financial statement features can be used to detect EM. This study is one of the few papers incorporating machine learning in EM detection, using financial statement items as features. 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 a drawback regarding its data retrieval method. 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 investors. However, most investors will not be able to replicate the results of the study as they cannot access the data themselves. This study attempts to solve this challenge by incorporating a new method in the use of machine learning in EM detection, which is eXtensible Business Reporting Language (XBRL). The following section will discuss what this method entails and how it could solve the data availability problem for retail investors in EM detection.

2.2 XBRL

This section covers prior literature on XBRL. First, XBRL is introduced. Next, its benefits and challenges in research are outlined. Finally, prior studies that combined XBRL and Earnings Management are discussed.

2.2.1 Introduction to XBRL

XBRL is a business and reporting technology that facilitates information sharing (Palas, 2019). It allows companies to share information in a standardized way. For example, firms can publish their financial statement using XBRL, which investors can access for their needs. XBRL reduces information processing costs, allowing users to accurately compare accounting information across firms (Janvrin, Pinsker, and Mascha, 2013). This is because using 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's architecture comprises several components, including XBRL taxonomies, XBRL linkbases, and XBRL Instance Documents.

The XBRL taxonomy is a dictionary that consists of XBRL tags that a company can use to tag specific financial statements. A tag consists of two elements. The first element refers

to the regulations of the specific financial statement item. The second element consists of the label corresponding to the financial statement item. For example, the cost of goods sold that a company reports with the U.S. GAAP is labeled as *us_gaap:CostOfGoodsAndServicesSold*. The taxonomy helps companies that report their financial statements using XBRL on which labels they can use. At the same time, it can guide investors in what a specific financial statement item entails and help them compare similar items across firms (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) analysed 4,532 financial statements and found 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. The challenges of XBRL are further outlined in 2.2.3

The XBRL linkbases are additional information to the XBRL taxonomy and consist of metadata for each XBRL tag. It has several components; Reference, Calculation, Definition, Label, and Presentation (Yaghoobirafi and Nazemi, 2019). They all fulfill their objective of making XBRL documents more readable.

The XBRL instance document is the file that contains all the financial statement information. The SEC manages a platform named the Electronic Data Gathering Analysis and Retrieval (EDGAR), where companies' instance documents are stored. In 2018, the SEC introduced inline XBRL (iXBRL), which entails that the XBRL data is now embedded in the original instance document. This results in one document that is both machine-readable and human-readable, which makes iXBRL more transparent than XBRL (Basoglu and White, 2015).

2.2.2 Benefits of XBRL

As mentioned in the previous subsection, using XBRL for financial statements has several benefits, especially for its stakeholders. XBRL allows users to process, validate and analyse financial statements time-efficiently (Liu et al., 2017). As the reports are machine-readable and consist of standardized tags that are part of the XBRL taxonomy, the comparability issue of financial statements seems to be resolved. 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 lower the information asymmetry between companies and investors (Liu et al., 2017).

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

2.2.3 Challenges of XBRL

XBRL should follow high standards regarding its data quality to fulfill its purpose of easy information-gathering. 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 by the SEC to submit their financial statements in XBRL format. 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 fillers' lack of knowledge of XBRL. 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, while 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, which is likely due to the lack of use of XBRL.

In contrast, Liu et al. (2017) found a decrease in information asymmetry between companies and investors due to the adoption of XBRL, as it is easier for investors to access financial statements. Still, Blankespoor, Miller, and White (2014) showed that the information asymmetry within the group of investors has increased. They found that the technical capabilities within the group of investors differ, which may have worsened the information asymmetry between small retail and large institutional investors. This is in contrast with the SEC's goal for XBRL; *"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 retail investors and found that this group does not use XBRL as much as was expected by the SEC due to the issue mentioned above regarding technical expertise. Still, XBRL has its advantages in comparison with third-party databases as COMPUSTAT when it comes to correctness, completeness, and timeliness. This study attempts to test whether XBRL can be used to detect EM. The following subsection will discuss some prior research on XBRL and EM.

2.2.4 XBRL and Earnings Management

There have been several studies that combined XBRL and EM literature. Kim, Kim, and Lim (2019) used a different perspective for the two fields and analysed whether the adoption of XBRL would lead to lower EM. They found that the discretionary accruals, which served as the proxy for EM, were significantly lower in the post-XBRL period compared to the pre-XBRL period. Even though this is a different application of the two fields in research, its implication for this study is that the EM detected will likely be smaller than 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 that firms have because of the obligation to report their financial statements using XBRL.

Most papers in this field focused on adopting XBRL and its effect on 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.'*(SEC, 2009). Therefore, XBRL should enable investors and other stakeholders to analyse financial statements for various purposes, as EM detection. Still, there have not been any papers that attempted to predict EM using data that is retrieved using XBRL. This could be due to the high error rate researchers found in XBRL data, which lowers their interest in using XBRL data. In fact, Ahmi and Mohd Nasir (2019) researched the trend of XBRL studies in this century and found that there are fewer 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 as COMPUSTAT regarding time and efficiency. This study fills the gap of EM detection using XBRL and attempts to assess its usefulness for investors.

3 Data and Methodology

This chapter discusses the data and methodology used in this study. First, the data source and the handling of the data 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 shown.

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, 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 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 and onwards is used since this was the first year that the SEC obliged firms to publish their financial statements in iXBRL, which drastically increased the data quality. All firms' numeric financial statement information 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 used in this research, including how they are calculated. Since some features are constructed by calculating the change in a specific year, data from 2017 is also retrieved to calculate the changes in 2018. After calculating all features, they are winsorized on the 1% and 99% level to deal with outliers and then normalized such that the mean equals zero and the standard deviation equals 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 most significant drawbacks of using XBRL is related to its data quality. The XBRl tags used for similar financial statement items across firms are inconsistent. This leads to challenges regarding aggregation. Appendix B shows a list of all the financial statement items used to calculate the features and the XBRL tags that are part of that individual item. The tags that are part of each specific financial statement item are chosen by the author. The first matched label per financial statement item, as specified in Appendix B, is assigned to each firm. The order in Appendix B is based on how often the respective tag occurred in the whole data set. The next section discusses the methodology of this study.

3.2 Methodology

This section discusses the methodology of this research. First, the model that is used to proxy EM is explained. Next, the feature selection method 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

This study uses the performance-adjusted Jones model to proxy EM (Kothari, Leone, and Wasley, 2005);

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

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 receivable; PPE is the gross property, plant, and; ROA is the return on assets calculated as earnings divided by total assets. All variables (excluding ROA) are deflated by the lagged total assets (AT).

The model will be estimated using OLS. According to Kothari, Leone, and Wasley (2005), total accruals can be split into non-discretionary and discretionary accruals. The latter serves as a proxy for earnings management and is equal to the residual of (1). The application of (1) will lead to an EM proxy for each ticker in each year. This study will discretize this variable instead of using this number in further analysis. Even though this modification will lead to a loss in model performance, since explainability 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 particular company in their investment decisions. This makes the model more interpretable for retail investors compared to if continuous numbers were used. A similar approach as Hammami and Zadeh (2022) is used to discretize the EM proxy by defining a ceiling and a floor. The Ceiling is defined as the average value of EM plus one standard deviation, and the floor is defined as the average value of EM minus one standard deviation. All EM proxy values between the floor and the ceiling are assigned the value zero, while values outside of these bounds are assigned the value one. This approach has some differences compared to Hammami and Zadeh (2022). They created buckets ranging from -2 and 2 instead of making a binary outcome variable. This study's method ensures that both high and low levels of EM are assigned the same label. This is in line with the definition of EM, which relates to management's intentional change of earnings. This could refer to both income-decreasing as well as income-increasing practices. Discretizing the EM proxy 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. A summary of how the proxy for EM will be referred to in the remaining sections of this study is shown in table 1.

3.2.2 Lasso Regularization

Before various machine learning models are evaluated, this study will attempt 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, reducing the number of features used will decrease the variance in the estimated coefficients since redundant variables are removed. This can lead

Table 1: Earnings Management Proxy

This table summarizes how the EM proxy is constructed and labeled.Each label's name, abbreviation, and corresponding interval are shown.

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	$(-\infty, \bar{x} - \sigma) \cup (\bar{x} + \sigma, \infty)$

to higher accuracy. Second, the model will be easier to interpret, as irrelevant variables can be removed.

There are different ways to control the number of features or dimensions. This study will use lasso regularization. This method is based on an Ordinary Least Squares (OLS) regression. However, compared to the OLS regression, the estimated coefficients in the Lasso regularization are shrunken toward zero. In fact, some features can be shrunken to precisely zero in lasso regularization. This makes it useful for variable selection.

Lasso regression is based on an OLS regression model. The OLS fitting procedure to calculate $\beta_0, \beta_1 .... \beta_p$ is done by minimizing the residual sum of squares as shown in (2):

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

Lasso regression has a similar approach but has an additional term that is used to shrink the size of estimated parameters, β_j, towards zero. This is done by minimizing the model as shown in (3). The λ, which is the penalty parameter, is tuned using a grid-search.

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

Different values of λ will be used to evaluate its performance on the training set. This process, called tuning, will allow this study to choose an appropriate value for lambda. Based on this value, all the coefficients are estimated using (3). The machine learning models will disregard all estimated coefficients shrunken towards zero.

Prior research by Li et al. (2021) also used lasso regularization in EM research and found a significant effect of real earnings management on financial distress.

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. Only using a subset of the features for each tree will solve any correlation among trees that might occur if there is one strong feature. This is because otherwise, each tree would use this strong feature to split the data, resulting in highly similar trees. For prediction purposes, the algorithm will aggregate the predictions of all trees to come up with a final prediction. The algorithm follows the procedure as described in algorithm 1.

Algorithm 1 Random Forest Algorithm

0: Set $\hat{f}(X) = 0$
0: **for** $b = 1, \ldots, B$ **do**
0: Randomly sample m training samples with replacement from the original dataset (X, y)
0: Randomly select d features from the total p features
0: Fit a decision tree $\hat{f}_b$ to the sampled data using the selected features
0: Update the ensemble by adding the new tree: $\hat{f}(x) \leftarrow \hat{f}(x) + \frac{1}{B}\hat{f}b(x)$
0: **end for**
0: Output the random forest model: $\hat{f}(x) = \frac{1}{B}\sum b = 1^B \hat{f}_b(x)$ =0

Two parameters will be tuned using a grid search. The first parameter, d, refers to the number of features used at each split. The second tuning parameter, m, is the number of trees to build.

Random forests have been widely used in accounting research, such as in fraud detection (Liu et al., 2015), failure prediction (Rustam and Saragih, 2018), and credit spread approximation (Mercadier and Lardy, 2019). Furthermore, it is also used in EM detection research (Chen and Howard, 2016; Xue and Ding, 2022).

3.2.4 Gradient Boosting

The second algorithm is also an ensemble method that attempts to improve the performance of a single decision 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 most challenging examples to predict and produce accurate predictions even when the data is noisy or contains outliers. The procedure of the gradient boosting algorithm is shown in algorithm 2.

Algorithm 2 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, d , will also be tuned

Gradient Boosting has shown good predictive performance in prior accounting studies, such as in risk management (Acharya et al., 2021) and credit card fraud detection (Rushin et al., 2017).

3.2.5 Support Vector Machine

The support vector machine (SVM) is widely used for regression and classification tasks. The goal of an SVM is to construct a hyperplane with *(p-1)* dimensions that can separate the data (Gareth et al., 2013). This hyperplane is selected to maximize its Euclidean distance with the nearest data points. These data points are called support vectors. SVM is useful for high-dimensional data and for modeling non-linearity. This is due to the kernel function that is used with an SVM. The kernel function transforms the data into a higher-dimensional feature space to control for non-linearity. This can be useful if the input data shows 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 3:

Algorithm 3 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. The two parameters are tuned using a grid search.

Several studies have used support vector machines for various purposes, such as financial distress prediction (Sun et al., 2021) and fraud detection (Sowah et al., 2019). Hammami and Zadeh (2022) used a support vector machine in their research on EM detection and found an accuracy of 90.4 %. This indicates that support vector machines can be a highly valuable model for EM detection. This finding was already motivated by Höglund (2012), who argued that the accrual process is non-linear and that research should focus on non-linear models for EM detection.

3.3 Descriptive Statistics

This section covers some descriptive statistics that can help improve understanding the data. The modifications applied to construct the final dataset are outlined in table 2. 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. Next, the data is retrieved from the SEC's API using a self-built scraper. A total of 55 ticker symbols did not respond to the API were 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 companies use or the labels that companies use. For example, firms might not mention their pre-tax income separately in their income statement or have no inventory. This will lead to missing values for their respective financial statement item. This issue fits the consensus on inconsistencies and errors researchers face when using XBRL (Hoitash, Hoitash, and Morris, 2021). The number of ticker symbols removed due to missing data decreases over the sample period. This indicates that the data quality and comparability over time are increasing. This aligns 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 features are based on changes with the previous year, which leads to the first time the firm occurs in the data set, the respective delta being zero. These ticker symbols are removed from the dataset to ensure this will not bias the results. All these modifications lead to a final data set with 1238 tickers that are used for further analysis.

Table 2: Ticker Selection

The table reports the steps that are taken, which led to the final dataset. The data sample consists of all the ticker symbols on the S&P 500 between 2018 and 2022, which equals a set of 2467 ticker symbols. First, financial ticker symbols are removed. Next, several ticker symbols are removed due to several data quality-related challenges. This leads to a final data frame of 1238 tickers. This dataset will be used in further analysis.

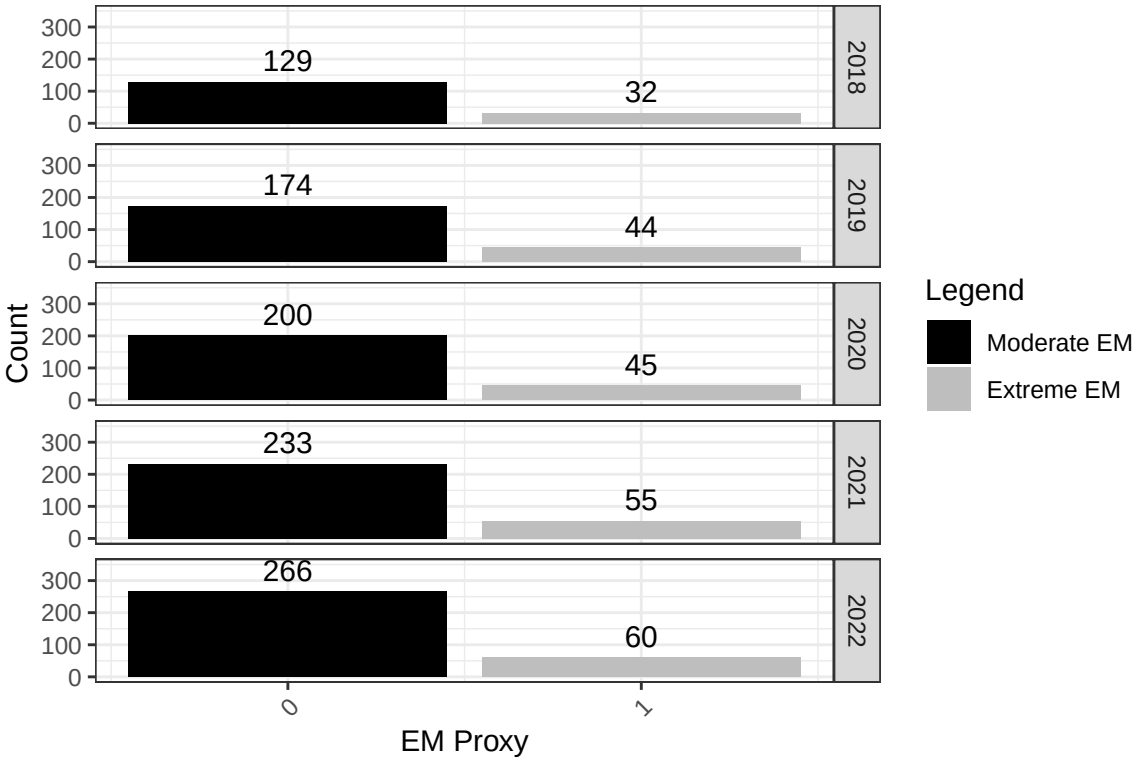
	2018	2019	2020	2021	2022	Total
Total Number of tickers on the S&P 500 per December	483	490	494	497	503	2467
Number of financial tickers	100	101	98	97	102	498
Total number of non-financial tickers	383	389	396	400	401	1969
Number of tickers with no data from scraper	14	6	20	11	4	55
Total number of tickers 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 Dataset	**161**	**218**	**245**	**288**	**326**	**1238**

Frequency histograms of all the features used in this research are added to Appendix C. 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 different units of measurement across the features.

EM is proxied using the performance-adjusted Jones Model (Kothari, Leone, and Wasley, 2005). Its distribution is plotted in figure 1. The figure shows the number of observations per label per year. The frequency of both labels increases over time, indicating that the number of complete observations increases. This could also be noticed from table 2, where the number of tickers in the final dataset increased over time.

Figure 1: Earnings Management Distribution

The graph shows the label distribution of the earnings management proxy per year.



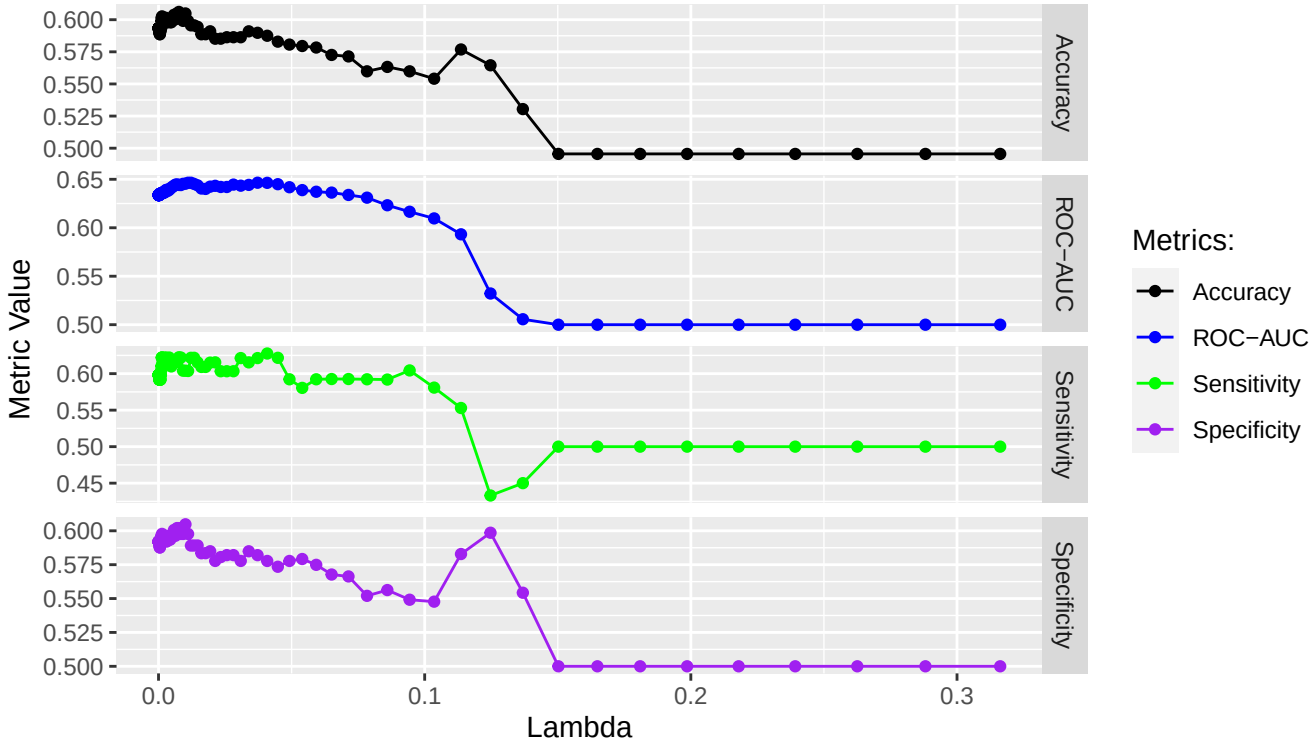
4 Results

4.1 Lasso Regularization

This section 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 study. This type of regression shrinks the values of all estimates towards zero compared to its OLS estimates. All variables with estimated coefficients shrunken to exactly zero will not be used in the machine learning models. This will lead to more parsimonious models because unnecessary features will be removed from the dataset. The data is split into 70%/30% for training and testing. The penalty parameter λ, which controls for the shrinkage, is tuned based on 100 different values ranging from $10^{-4.5}$ and $10^{-0.5}$. The model is then evaluated using its tuning results based on the following metrics; accuracy, sensitivity, specificity, and ROC-AUC. After applying the lasso regularization on the training set, the tuning results across all penalty parameters are shown in figure 2. The figure shows that all metrics range between approximately 50 % and 60 %. Based on the tuning, the model chosen is the one that achieved the highest accuracy. This is because this model also has a decent level for the other metrics. The value of λ is 0.03635851.

Figure 2: Tuning Results Lasso Regularization

The graph shows the tuning results of the lasso regularization across a range of values for Lambda.



Based on this penalty parameter, the coefficients $\beta_0, \beta_1 .... \beta_p$ are estimated using the training set and tested on the test set. The metric results of the application of the model on the test set are shown in table 3. The model seems to perform poorly, with an accuracy of just 55.1%. Still, the sensitivity is relatively good at 70.4 %

Table 3: Metrics Results Lasso Regularization
The table shows the metric performance metrics of the lasso regularization.

Metric	Estimate
Accuracy	0.551
Sensitivity	0.704
Specificity	0.515
ROC-AUC	0.652

As mentioned before, the goal of lasso regularization in this study is to use it as a feature selection method. The estimated coefficients of the lasso regression, based on the aforementioned penalty parameter, are shown in Appendix D. All estimated coefficients are shrunken towards zero in comparison to its OLS estimate. Five variables have estimated coefficients equal to zero and seem redundant in detecting earnings management. These variables are: *"EquityOverFixedAssets", "PreTaxIncomeToSales", "ChangeInAssets", "ChangeInRevenues", "ChangeInDaysSalesinAccountingReceivable"* Including these features will lead to less price estimates for the other features. Therefore, they are not used in subsequent analyses.

The next section discusses the results of the application of the three machine learning models.

4.2 Machine Learning Models

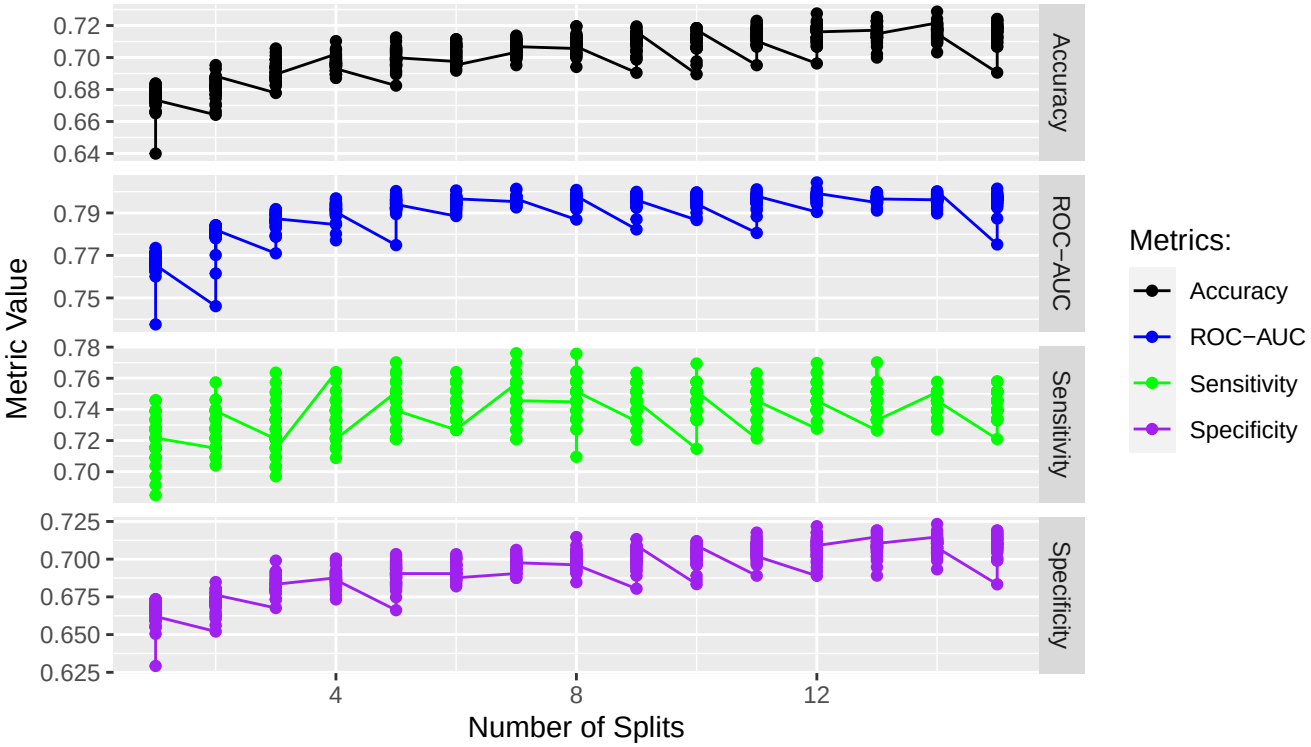
This section discusses the application of the machine learning models as discussed in subsection 3.2. All the models have a similar setup. The data is split into 70%/30% for training and testing. The split is stratified on the dependent variable to ensure that the distribution of the dependent variable 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 used to evaluate the results are accuracy, sensitivity, specificity, and ROC-AUC. The goal is to maximize sensitivity while maintaining a good level of specificity, accuracy, and ROC-AUC. 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's perspective, keeping the number of false negatives as low as possible is important since an investor should have avoided these firms in their investment decisions. 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. 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 tuned. The number of features is set between 1 and 15, and the number of trees is set between 50 and 2000 with steps of 50. Tuning will prevent overfitting, reduce model complexity and improve model accuracy. figure 3 shows the results of the model tuning. The number of splits is plotted against the metric value, where each dot represents a different value for the second tuning parameter, the number of trees. As can be observed from the

graph, for a given number of splits, the dispersion across the number of trees is relatively low. Furthermore, the metric values marginally differ across the number of splits. The model that is chosen is the model with the highest level of sensitivity, as it corresponds with a good level of specificity, accuracy, and ROC-AUC. The number of splits is 7, whereas the number of trees is 200. The model is trained using these hyperparameters and tested on the test set.

Figure 3: Tuning Results Random Forest

The graph shows the tuning results of the random forest model on the training set against the number of splits. Every dot represents a specific number of trees.

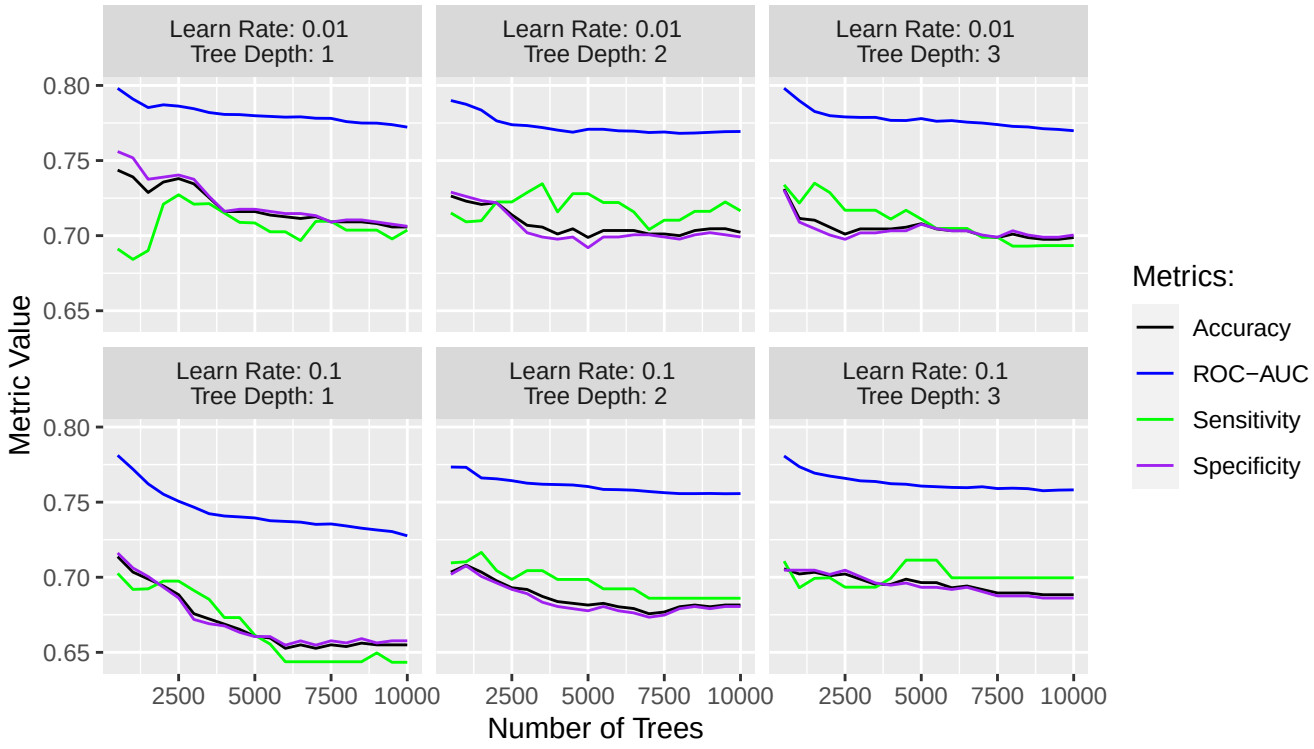


Next, three hyperparameters of the gradient boosting model are tuned. The first tuning parameter is the number of trees and is set between 500 and 10000 with steps of 500. The second tuning parameter is the learning rate, which is set to either 0.01 or 0.10. The third and final tuning parameter is the tree depth, which is set between 1 and 3. The metric results of the model tuning can be found in table figure 4. The graph shows the metrics results across different number of trees for the six different groups of learning rate and three depth. For a learning rate of 0.1, the model performs better with a higher tree depth for the same number of trees. For a learning rate of 0.01, the figures remain relatively stable across the various tree depth values. The model chosen is based on the model with the highest sensitivity, which corresponds with a learning rate of 0.01, tree depth of 3, and number of trees of 1500.

The third and final model that is tuned is the Support Vector Machine (SVM). SVM is a machine learning algorithm that attempts to construct a hyperplane that could split the data. The dimensionality is transformed using the Radial Basis Function (RBF). Two parameters are tuned. The first parameter is the cost C, which controls the degree of misclassification and is set between 10 and 100 with steps of 10. The second parameter is the λ, which controls

Figure 4: Tuning Results Gradient Boosting

The graph shows the tuning results of the Gradient Boosting Algorithm across the number of trees for each unique combination of learning rate and tree depth.



the width of the RBF. This parameter is set between 0.00001 and 0.01. The results of the tuning process against the cost for the various values for λ are plotted in figure 5. It is essential to choose the hyperparameters carefully to ensure that the model can have a good out-of-sample performance. The graph shows a better tuning performance for higher levels of lambda. A higher value of lambda indicates that each data point only has a substantial value on the points very close to it. Furthermore, the metrics remain relatively stable across the various cost levels. Based on figure 5, the hyperparameters chosen are 10 for cost and 0.01 for λ. This model has the highest tuning sensitivity and performs well across the other three metrics. The model is trained using these parameters and fitted on the test set.

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

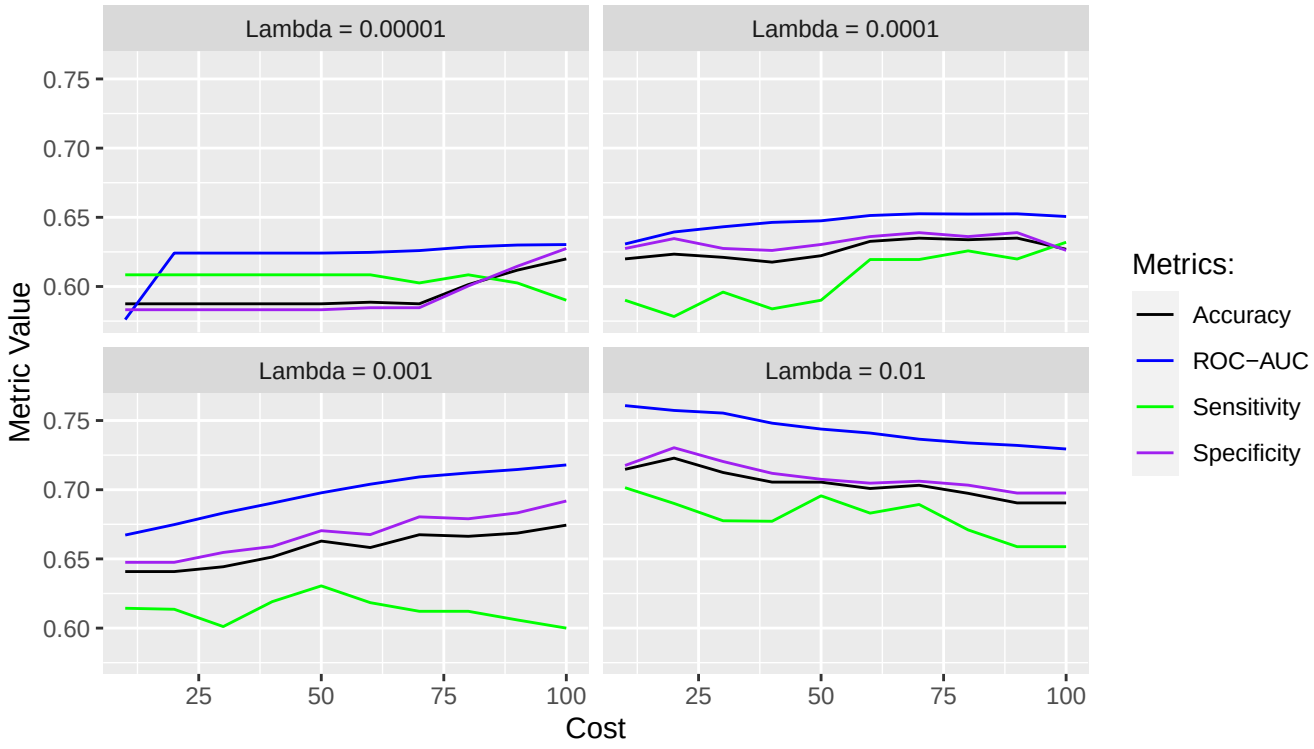
Table 4: Metrics Results Earnings Management Detection

This table shows the metric performance for the Random Forest (RF), Gradient Boosting (GB), and Support Vector Machine (SVM) models

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 5: Tuning Results Support Vector Machine

The graph shows the tuning results of the Support Vector Machine across a range of values for the cost, where each graph represents a different value of lambda.



The goal of the machine learning models was to achieve a high sensitivity while maintaining a good level of specificity and accuracy. table 4 indicates the out-of-sample performance of the four metrics across the three different machine learning models. The highest accuracy is achieved with the Support Vector Machine, which is just 0.2 percentage points higher than the accuracy of the Gradient Boosting algorithm. The random forest prediction was the least accurate, with a metric value of 65.9 %. Nevertheless, the sensitivity of the random forest model is the highest metric overall at 80.3 %. This indicates that the random forest model did an excellent job of keeping the number of false negatives as low as possible, even though the model is least accurate overall. The sensitivity of the support vector machine is the lowest overall metric, at just 57.7 %. Therefore, in contrast to the random forest model, the support vector machine did a poor job keeping the number of false negatives as low as possible while still having the best accuracy. The specificity of the random forest model sits at just 62.5 %, which is the lowest specificity across all three models. Nevertheless, as mentioned before, specificity is less important as it considers the number of false positives. To conclude, all three machine learning models have a relatively good ROC-AUC test set performance, ranging from 73.9% to 79.3%

Hammami and Zadeh (2022) also used a support vector machine in their research on EM detection and found accuracies between 80 % and 90%, which is higher than what this study found. Hammami and Zadeh (2022) used different features that have also been used in other EM detection studies (Dechow et al., 2012; Chen, Chi, and Wang, 2015; Höglund, 2012).

This study focused on a different set that was used in earnings movement prediction (Palas, 2019). While there are some common features between this study and the study of Hammami and Zadeh (2022), there are also several differences. This indicates that the set of features used by Palas (2019) in their research on earnings movement prediction is less valuable in EM detection than what's used in prior EM detection studies.

Next, a confusion matrix is set up to analyse the performance of the classification. The matrix is shown in table 5.

Table 5: Confusion Matrix Earnings Management Detection
This table shows the confusion matrix for the Random Forest (RF), Gradient Boosting (GB), and Support Vector Machine (SVM) models.

	Prediction					
	RF		**GB**		**SVM**	
	Moderate EM	**Extreme EM**	**Moderate EM**	**Extreme EM**	**Moderate EM**	**Extreme EM**
Actual Moderate EM	188	113	224	77	234	67
Actual Extreme EM	14	57	21	50	30	41

As mentioned, this study's goal is to recommend whether or not investors should keep a firm in their investment portfolio based on its earnings management proxy. This is because earnings management can mislead investors regarding a company's performance. When evaluating the test set performance, it is curcial to consider the firms labeled to have a moderate proxy for EM, even though they had an extreme proxy for EM. This miss-classification is crucial to minimize, as investors should avoid firms with an extreme proxy for EM. The other miss-classification, which considers the firms that had an actual moderate proxy for EM but are predicted to have an extreme proxy for EM, is less important from an investor's point of view.

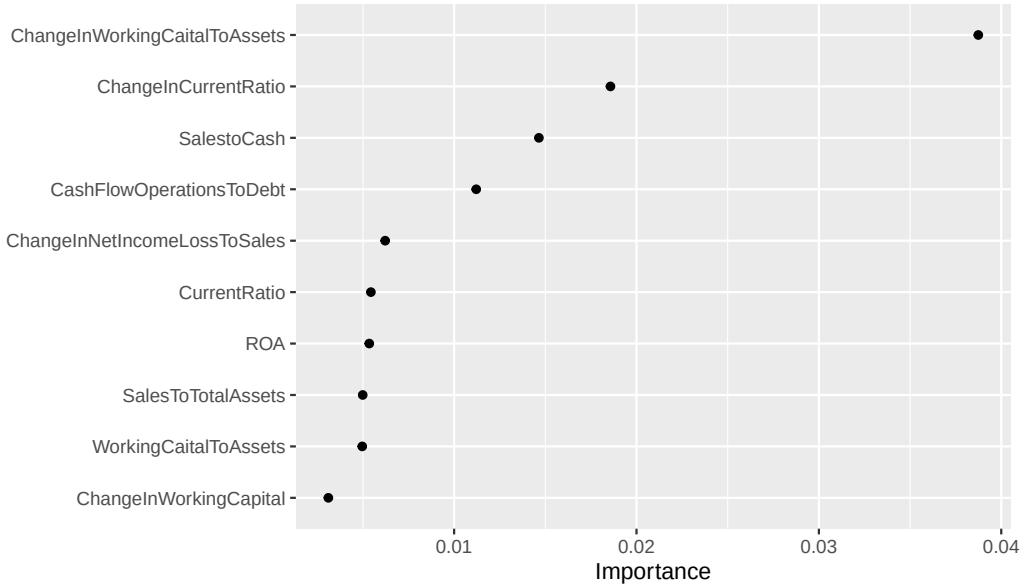
The confusion matrix in table 5 shows that in the random forest model, 14 ticker symbols were labeled to have a moderate proxy for EM, while their actual proxy for EM was extreme. This is the lowest miss-classification across the three machine learning models, with the highest being the support vector machine. This finding corresponds with the high sensitivity in the random forest model and the low sensitivity for the support vector machine. When considering the other miss-classification, table 5 shows that the random forest gave 113 out of the 301 ticker symbols with moderate proxy for EM a wrong prediction. This ratio is lower for the gradient boosting and support vector machine. In this last model, only 67 out of 301 ticker symbols with a moderate proxy for EM were misclassified. This finding corresponds with the low/high specificity in the random forest model/support vector machine.

Since random forest algorithms are based on multiple decision trees, it is impossible 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 decision trees. figure 6 shows the top 10 most essential 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 aligns with prior research by Aduda, Ongoro, et al. (2020), who argued that earnings management could be a consequence of working capital management. This study confirms this finding and argues that modifications in working capital to assets are strongly associated with EM. Still, this could mean that there is only a correlation between

the respective feature and the EM proxy. As mentioned before, Tsai and Chiou (2009) argues that most studies on EM have focused on the correlation between a specific factor and EM. However, this does not give any guidance in the detection of EM. Therefore, while figure 6 provides valuable insights into the most important features in the RF model, more emphasis should be placed on the out-of-sample metrics results as described in table 4.

Figure 6: Variable Importance Random Forest

The graph shows the ten most important features in the random forest model.



5 Investor's application

5.1 Focus on Investor

As mentioned before, this study focuses on the usefulness of XBRL from an investor's perspective. Prior research has shown that investors use financial statements in their investment choices (Hoffmann and Shefrin, 2014; Chen et al., 2019). XBRL, which can be used to access companies' financial statements, has great advantages compared to databases as COMPUSTAT regarding efficiency and accessibility (Guo and Yu, 2022). The SEC stated: *"Interactive data can create new ways for investors, analysts, and others to retrieve and use financial information in documents filed with us."* (SEC, 2009). Furthermore, in the same file, the SEC (2009) also stated that: *".... is helping make interactive data increasingly useful to both institutional and retail investors, as well as to other participants in the U.S. and global capital markets."*. While the idea of XBRL improving the data availability for (retail) investors sounds promising, its actual outcome has been different than expected. Guo and Yu (2022) analysed the use of XBRL by retail investors by studying the SEC's download log. They found that small investors do not use XBRL. Even though the authors do not explain its reason, they argue that it could be due to a lack of technical capabilities. In fact, Blankespoor, Miller, and White (2014) argue that XBRL has increased the information asymmetry between small and large investors since small investors lack the knowledge to use XBRL. Therefore, this study attempts to facilitate (retail) investors in two different ways in which they can use XBRL for EM detection.

First, all relevant files and code are stored in a GitHub repository (`https://github.com/sunny1999123/MasterThesis`). This allows investors to replicate this study on a whole different set of firms. The following paragraph elaborates on all the R files used in this research. This enables investors with the necessary tools and knowledge to perform EM detection analysis themselves[2]. On top of that, an EM detection tool is developed (`https://sunny1999.shinyapps.io/EMDetectionTool/`). This tool allows investors to analyse a firm of their interest. The mechanics of this tool is outlined in subsection 5.3

5.2 Use of GitHub

All the relevant code and files are stored in a GitHub repository[3]. This platform enables version control and collaboration on projects. All the files and code can be collected via this repository and stored in an R project. To do so, one should open Rstudio, open a New Project, click on version control, and paste the link mentioned above to create a new project. The project, including all its files and code, will now be stored in a directory of the user's computer. This allows anyone to replicate this study. This study consists of several steps that are used to perform the analysis. All R files 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 data frame 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

[2] A basic understanding of the R programming language, Git, and data analysis is required to use the GitHub directory

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

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 data frame with tickers and years as input. Next, in *2.Scraper.R*, all the data is scraped using an API and stored in a data frame. The third file, *3.Cleaning Results.R* takes all the data 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.R*. This file outputs a data frame with all the data needed for the machine learning models. Before these models can be applied, this study focuses on selecting an appropriate set of features by applying lasso regularization. This is done using *5.Feature Selection.R*. Next, the three machine learning models applied in this research are stored in three different R files (*6.Random Forest.R, 7.Gradient Boosting.R, and 8.SupportVectorMachine.R*). These R files have a similar layout. First, the results of the lasso regularization serve as an input for the machine learning model. Next, the models are all split into a training and testing set, the parameters are tuned, the appropriate model is chosen, and the model is applied to the test set, after which the metrics are collected. These metrics can be used to evaluate the performance of the EM detection algorithm. Still, running all code does not recommend whether a specific firm should be included in an investor's portfolio. To achieve this, an EM detection tool is developed and outlined in subsection 5.3.

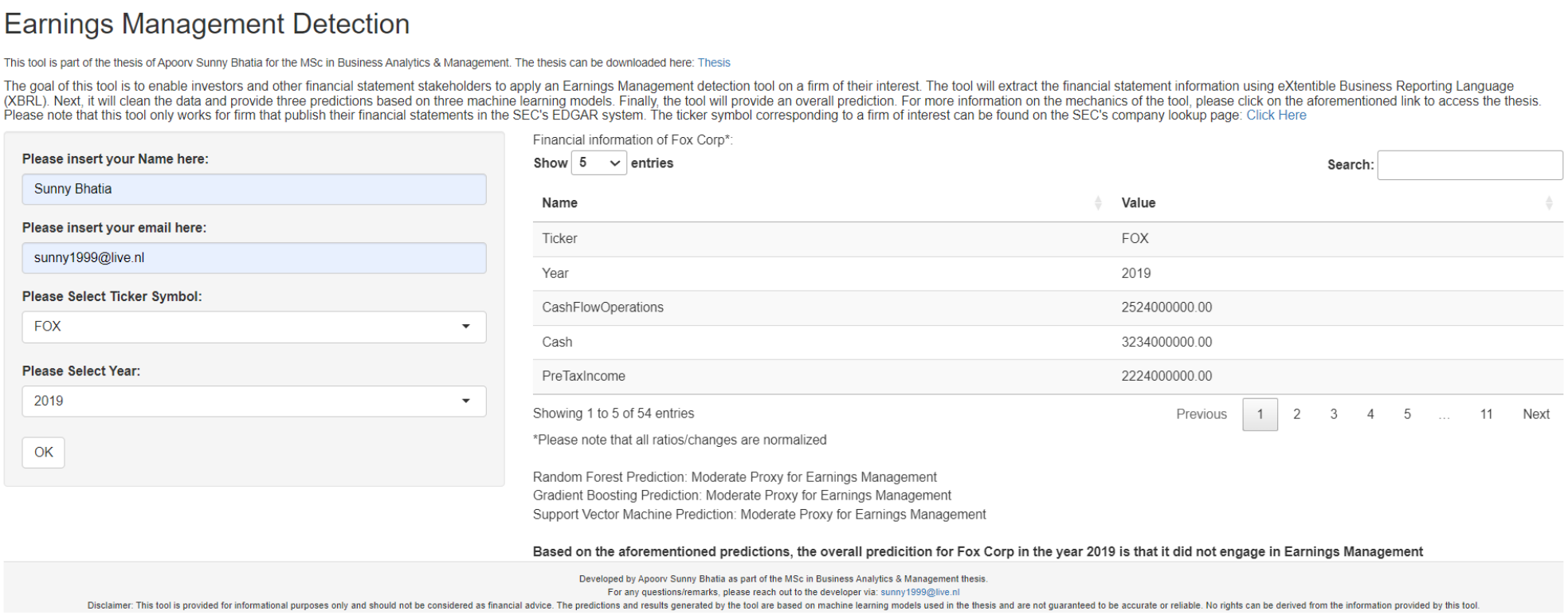
5.3 Earnings Management Detection Tool

While the previous section provides a summarized overview of all the relevant code files, it does not enable investors to analyse a specific firm of their interest. Therefore, using an EM detection tool, investors can also analyse whether the models used in this research would predict a moderate or extreme EM proxy for their firm of interest. This can help investors decide whether or not a firm should be included in their investment decisions. This tool aims to increase the practical usefulness of this study for retail investors.

The tool can be accessed via `https://sunny1999.shinyapps.io/EMDetectiontool/`. When the tool is opened, a user will see four input fields. The first and second inputs require the name and email of the user. This information is needed to set the user-agent to retrieve the data via an API from the SEC's EDGAR database. The third input field requires the ticker symbol of the firm of interest for the investor. The fourth input requires the year of interest and can be chosen from the list of years. Its lower limit is 2018, as the data quality before this year is too low. After the name and email are inserted and the ticker symbol and year are selected, an investor should click 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 using XBRL. Next, the data will be cleaned, and the features will be calculated. After this, the machine learning models are trained on the full dataset used in this research with the same hyperparameters. Finally, a prediction of all three machine learning models based on the input will be shown. Per model, the tool will show a "Moderate Proxy for Earnings Management" or an "Extreme Proxy for Earnings Managment" as the label. At last, a final prediction is shown, which is based on the most occurring label. An example of the input FOX and 2019 is shown in figure 7.

Figure 7: Earnings Management Detection Tool

The figure shows an example of the output of the EM detection tool. The tool can be accessed via `https://sunny1999.shinyapps.io/EMDetectiontool/`.



6 Discussion

6.1 Main Findings

This study aims to analyse whether earnings management (EM) can be detected using eXtensible Business Reporting Language (XBRL) through machine learning models. The motivation of this study is to assess the usefulness of XBRL data for its stakeholders. XBRL has many advantages over databases in terms of time and cost. Still, its usefulness in predicting EM is yet to be studied.

The problem statement is split into four research questions. The first research question relates to how EM can be proxied using XBRL. While there is much debate in the accounting field on how this measure can be proxied, this study followed the widely used Jones Model that extracts the discretionary accruals and uses that as a proxy for EM. This model effectively detects EM (Islam, Ali, Ahmad, et al., 2011). Still, in recent years, there has been an upcoming debate on the actual performance of the Jones Model (McNichols and Stubben, 2018). The critique mainly centers around the fact that many studies do not consider the underlying drivers of the Jones Model. This study attempts to solve this by analyzing which financial statement features drive discretionary accruals. To conclude, in response to the research question, this study has shown that XBRL can be used to proxy EM.

The second research question is: *Which set of financial statement features contains relevant information that can be used in EM detection?*. This study used a similar set of features as Palas (2019), which used XBRL in earnings movement prediction and found an excellent out-of-sample performance in predicting earnings movement. Using this set will allow this study to determine whether features that perform well in predicting earnings movement can perform well in EM detection. This study identifies the relevant features that can be used in the machine learning models by applying a lasso regularization algorithm. All the estimated coefficients shrunken to zero are removed from further analysis to reduce model complexity and to create a more parsimonious model. From the original set of features as specified in Appendix A, five features are removed from the dataset, which are: *"EquityOverFixedAssets", "PreTaxIncomeToSales", "ChangeInAssets", "ChangeInRevenues" and "ChangeInDaysSalesinAccountingReceivable"*. All other features have estimates coefficients that are not shrunken to precisely zero. Thus, this set contains relevant information for EM detection based on the lasso model and is used in further analysis.

The third research question relates to the application of the machine learning models. The three models that are used are the Random Forest (RF), Gradient Boosting (GB), and Support Vector Machine (SVM). This study evaluates whether the set of features can be used in EM detection. The models are evaluated based on several metrics. As outlined in the research question, the goal is to find a model with a high sensitivity level while maintaining a good level for the other metrics. This study focuses on sensitivity because this metric is most important from an investor's perspective, as it relates to firms with a high proxy for EM but is wrongly classified. Based on the performance metrics, the best model is the RF. This

model achieved the highest sensitivity while maintaining a good level for the other metrics. Also, its ROC-AUC metric is the highest across all three models. Still, the performance of the machine learning models in this study is lower than what has been found in other EM detection-related studies (Hammami and Zadeh, 2022; Chen, Chi, and Wang, 2015), which could be due to the set of features used in this study.

The fourth and final research question relates to the use of XBRL for EM detection. As mentioned before, XBRL has many advantages in terms of efficiency compared to databases (e.g., COMPUSTAT). This study aims to analyse whether XBRL can be used in EM detection. This question can be split into several parts. In terms of the data quality of XBRL, this study concludes that XBRL is becoming a better source for financial statement information. This study found that the data quality of XBRL increased over time. This can be due to the experimental learning of firms and the improvements in monitoring by the SEC. However, there are still some issues regarding the data quality of XBRL due to differences in the usage of XBRL tags across firms. Still, as XBRL allows financial statement users to retrieve the data directly from the source, this study argues that its advantages out-weight its disadvantages and that XBRL is a good alternative for databases as Compustat. The second part of this research question is related to the actual use of XBRL. As mentioned in Chapter 5, this study focuses on whether investors can use XBRL. Prior research has shown that the usage of XBRL by retail investors is small (Guo and Yu, 2022), which is likely due to the lack of technical knowledge. This study tries to solve this in two different ways. First, for investors with a basic understanding of the R programming language, all files are stored in a GitHub repository. Furthermore, an EM detection tool is developed that allows stakeholders to perform their own EM detection analysis on a firm of interest. So, most (retail) investors will not be able to use XBRL due to the technical capabilities that it demands. Still, the EM detection tool allows investors to implicitly use XBRL without the need for any (advanced) technical skills. The third part of this research question relates to the use of XBRL, specifically for EM detection. This study evaluated several machine learning models using only XBRL data and found a decent out-of-sample performance. To conclude, this study argues that XBRL can be used in EM detection.

6.2 Contribution

This study has several contributions to the existing literature. First of all, this study has shown that it is possible to calculate an EM proxy using XBRL. The only paper that has done this before was the paper by Henselmann, Ditter, and Scherr (2015). Still, they proxied EM using Benford's Law, which differs from the adjusted Jones Model used in this study. Second, this study has shown that the set of features used by Palas (2019) to predict earnings also has valuable information regarding earnings management. This set of features has not been used in EM research yet. Furthermore, this study found that the Random Forest model had a good out-of-sample performance, whereas the Support Vector Machine did a poor job in this study. In contrast, Hammami and Zadeh (2022) used a Support Vector Machine to detect EM and found a good performance. Finally, the main contribution of this research

is related to the use of XBRL in EM detection. Prior research by Guo and Yu (2022) has shown that investors do not use XBRL, which could be due to the needed technical skills. Still, this study provides two ways investors could use XBRL for EM detection. First, the code and relevant files are stored in a GitHub repository, allowing investors to replicate this study on different firms. Furthermore, this study developed an EM detection tool. Therefore, this study has shown that XBRL can be used to detect EM in terms of data availability and machine learning results and has enabled investors and other stakeholders to use XBRL for EM detection in two different ways.

6.3 Limitation

The limitations in this study can be split into three categories; XBRL limitations, EM limitations, and GitHub/Shiny-related limitations.

As mentioned before, one of the main concerns in research regarding XBRL is its data quality. This is mainly related to the XBRL tags. Firms use different tags for similar financial statement items, which makes aggregation complex. This study looped through a list of pre-determined tags for each financial statement item to assign a value to a specific item. However, this method could have led to wrong tags for certain financial statement items, which is a limitation of this study. Furthermore, due to different types of reporting styles and business activity, firms might not report certain financial statement items (explicitly). These limitations are shown in table 2, where the final dataset is approximately half the size of the original dataset.

This study used discretionary accruals as a proxy for EM. While this method is often used in EM research, there is also some critique by (Jackson, 2018), who argue that there is limited research on the comparison between discretionary accruals and confirmed cases of earnings management.

Finally, there are several limitations to using the GitHub repository and the Shiny WebApp. Users of the GitHub repository should have a basic understanding of the R programming language to replicate this study. As many retail investors would need more programming knowledge to replicated this study, its actual use by this group will be limited. The Shiny WebApp is more accessible and easier to use. Still, it only works when retrieving all the financial statement items used for the features and the dependent variable is possible. The tool will not work for ticker symbols that suffer from data quality issues related to XBRL. Still, as the data quality of XBRL has increased over the sample period, the tool will become increasingly helpful over time.

6.4 Further Research

This section will provide several recommendations for further research. The first recommendation is to replicate this study in 5-10 years using the new data generated in these years. The data quality of XBRL has drastically increased since the launch of iXBRL by the SEC in 2018. Replicating this study in a few years will allow to use more firms to train the models, leading to better predictions. The data quality overall will also increase, leading to fewer missing values. Next, related to the data quality of XBRL, one could verify the XBRL data

with other sources as COMPUSTAT to analyse any deviations. This could lead to more insights into the data quality of XBRL, as it can be compared to a different source. Furthermore, as XBRL is shown to be a valuable source for financial statement information, further research could consider using this for other accounting topics where financial statements are needed, such as fraud detection and failure prediction.

This study used the discretionary accruals, calculated using the Jones Model, as a proxy for EM. Further research should consider other proxies for EM, such as Beneish's M-score, Benford's Law, or real earnings management. This could lead to different insights regarding the detection of EM using XBRL.

The features used in this research are based on prior research by Palas (2019) on earning movement prediction using XBRL. Still, financial statements contain more information that could be used in further research. For example, new research could apply NLP to the financial statement text retrieved via XBRL and analyse whether its sentiment can be used as a feature in EM detection. This would add to prior research by Li, Wang, and Luo (2022), who found that the tone in financial statements is related to EM.

Further research could consider other feature selection methods than the one used in this research, which could lead to a more parsimonious model. At last, further research could focus on different machine learning models, as neural networks, that can lead to a better-performing model.

6.5 Conclusion

The study has shown the potential of using XBRL in detecting EM. By retrieving data using XBRL and identifying relevant features, this study has demonstrated that XBRL can be a valuable tool in detecting EM. The Random Forest model has shown the best out-of-sample performance in EM detection. This study has contributed to the practical usefulness of XBRL for retail investors by collecting all the code in a GitHub repository and developing an EM detection tool. This enables stakeholders to use XBRL for EM detection analysis. To conclude, this study highlights the usefulness of XBRL in EM detection despite its limitations and suggests further research in this field.

A Features

Table 6: Features

The table shows all the features that are used to predict the EM proxy and their calculation.

	Feature	Calculation
1	Current Ratio	$\frac{\text{Current Assets}}{\text{Current Liabilities}}$
2	Accounts Receivable Turnover	$\frac{\text{Revenues}}{\text{Accounts Receivable}}$
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	Sales to Working Capital $_t -$ Sales to Working Capital $_{t-1}$
34	Change in Working Capital to Assets	Working Capital to Assets$_t -$ Working Capital to Assets$_{t-1}$
35	Change in EBITDA Margin Ratio	EBITDA Margin Ratio $_t -$ EBITDA Margin Ratio $_{t-1}$
36	Change in Debt	$\text{Debt}_t - \text{Debt}_{t-1}$

B Financial Statement Items and XBRL Tags

Table 7: Financial Statement Items and XBRL Tags

The table shows all the financial statement items that are and their respective tags.

Financial statement Item	Tags
Revenue	*us_gaap:RevenueFromContractWithCustomer-ExcludingAssessedTax*
	us_gaap:ContractWithCustomer-LiabilityRevenueRecognized
	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:StockholdersEquityIncludingPortionAttributable-ToNoncontrollingInterest
	us_gaap:StockholdersEquityOther
NetIncome	*us_gaap:NetIncomeLoss*
	us_gaap:ComprehensiveIncomeNetOfTax
	us_gaap:OperatingIncomeLoss
	us_gaap:IncomeLossFromContinuingOperationsBefore-IncomeTaxesForeign
	us_gaap:IncomeLossFromContinuingOperationsBefore-IncomeTaxesDomestic
	us_gaap:IncomeLossFromContinuingOperationsBefore-IncomeTaxesExtraordinaryItemsNoncontrollingInterest
Assets	*us_gaap:Assets*

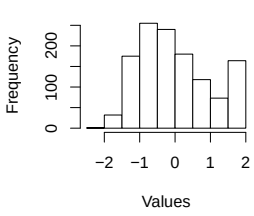
Cost of Goods Sold	*us_gaap:CostOfGoodsAndServicesSold*
	us_gaap:CostOfRevenue
Depreciation	*us_gaap:DepreciationDepletionAndAmortization*
	us_gaap:Depreciation
	us_gaap:DepreciationAndAmortization
	us_gaap:DepreciationAmortizationAndAccretionNet
	us_gaap:DepreciationNonproduction
PPE	*us_gaap:PropertyPlantAndEquipmentNet*
	us_gaap:PropertyPlantAndEquipmentGross
	us_gaap:PropertyPlantAndEquipmentAdditions
	us_gaap:PropertyPlantAndEquipmentOther
	us_gaap:PropertyPlantAndEquipmentDisposals
	us_gaap:PropertyPlantAndEquipmentAndFinanceLease-RightOfUseAssetAfterAccumulatedDepreciation-AndAmortization
	us_gaap:PropertyPlantAndEquipmentAndFinanceLease-RightOfUseAssetAfterAccumulatedDepreciation-AndAmortization
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:IncomeLossFromContinuingOperationsBefore-IncomeTaxes*
	us_gaap:IncomeLossFromContinuingOperationsBefore-IncomeTaxesMinorityInterestAndIncomeLossFromEquity-MethodInvestments
	us_gaap:IncomeLossFromContinuingOperationsBefore-IncomeTaxesExtraordinaryItemsNoncontrollingInterest
Cash	*us_gaap:CashAndCashEquivalentsAtCarryingValue*
	us_gaap:CashCashEquivalentsRestrictedCashAndRestricted-CashEquivalents
	CashEquivalentsAtCarryingValue
Cash flow from Operations	*us_gaap:NetCashProvidedByUsedInOperatingActivities*

	us_gaap:NetCashProvidedByUsedInOperatingActivities-ContinuingOperations *us_gaap:CashProvidedByUsedInOperatingActivities-DiscontinuedOperations*

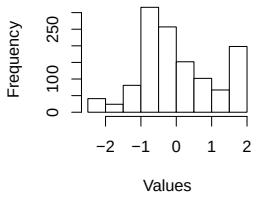
C Feature Distribution

Figure 8: Feature Distribution

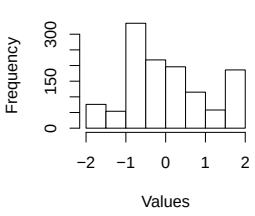
These figures show the distribution of all the features that are used in this analysis. All the features are winsorized and normalized.



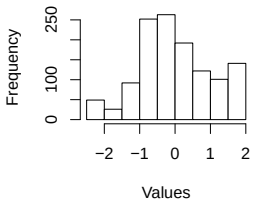
(i) Current Ratio



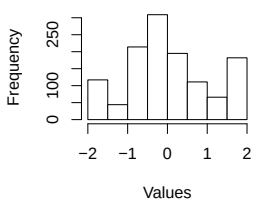
(ii) Accounts Receivable Turnover



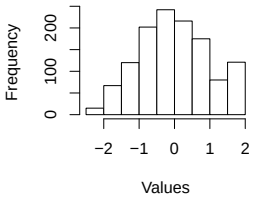
(iii) DebtToEquity



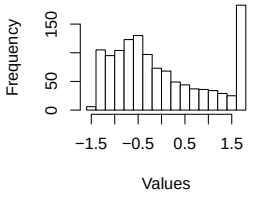
(iv) ROA



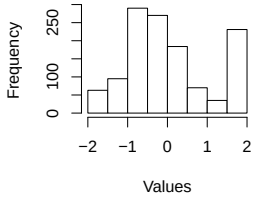
(v) ROE



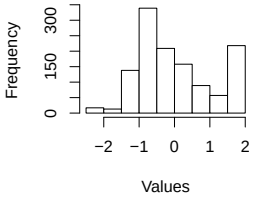
(vi) DaysSalesinAccountingReceivable



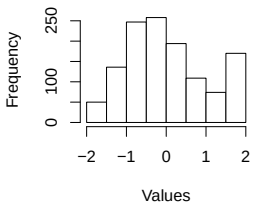
(vii) DepreciationOverPlant



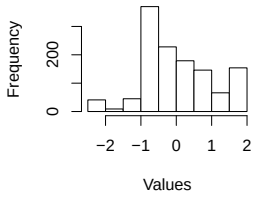
(viii) EquityOverFixedAssets



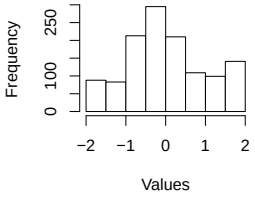
(ix) TimesInterestEarned



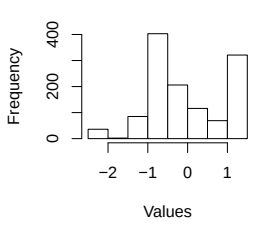
(x) SalesToTotalAssets



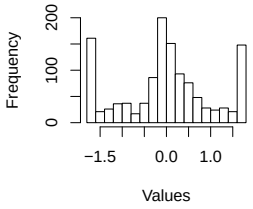
(xi) PreTaxIncomeToSales



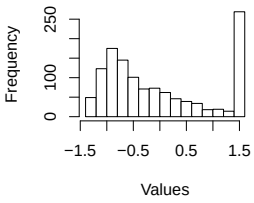
(xii) NetIncomeLossToSales



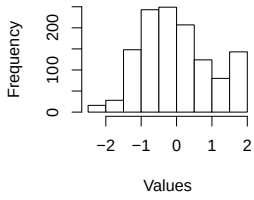
(xiii) SalestoCash



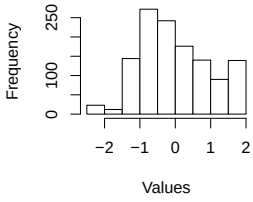
(xiv) SalestoWorking-Capital



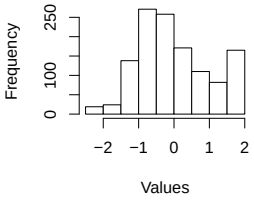
(xv) SalesToFixedAssets



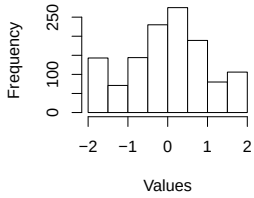
(xvi) WorkingCaital-ToAssets



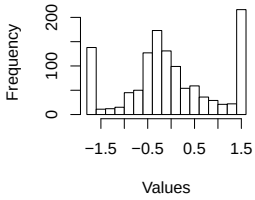
(xvii) EBITDAMargin-Ratio



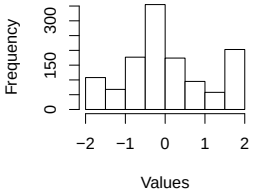
(xviii) CashFlowOperationsToDebt



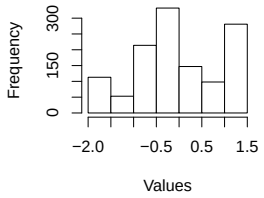
(xix) NetIncomeLossTo-Cashflow



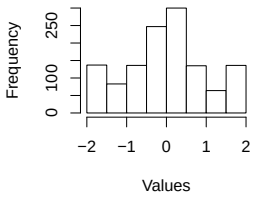
(xx) ChangeInDepreciation



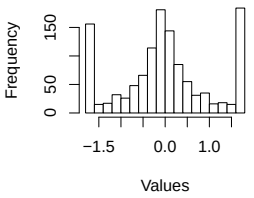
(xxi) ChangeInAssets



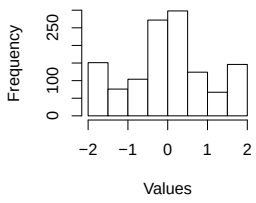
(xxii) ChangeInRevenues



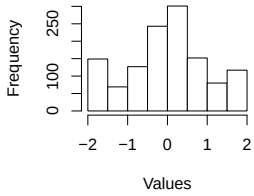
(xxiii) ChangeInCurrentRatio



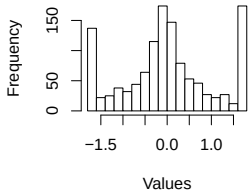
(xxiv) ChangeInDebt-ToEquity



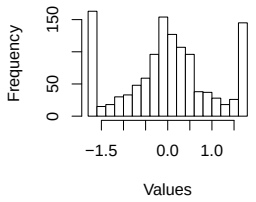
(xxv) ChangeInWorkingCapital



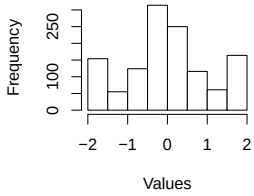
(xxvi) ChangeInDaysSalesinAccountingReceivable



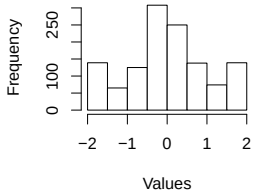
(xxvii) ChangeInDepreciationOverPlant



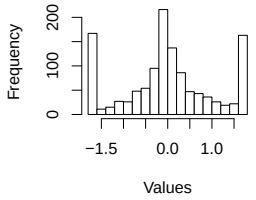
(xxviii) ChangeInEquityOverFixedAssets



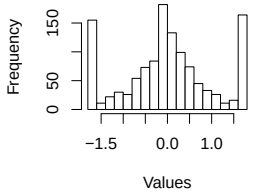
(xxix) ChangeInTimesInterestEarned



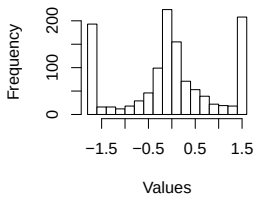
(xxx) ChangeInSalesToTotalAssets



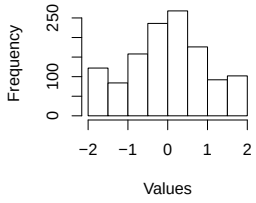
(xxxi) ChangeInPreTaxIncomeToSales



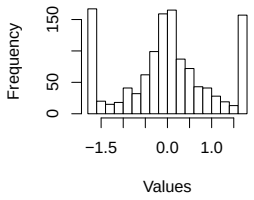
(xxxii) ChangeInNetIncomeLossToSales



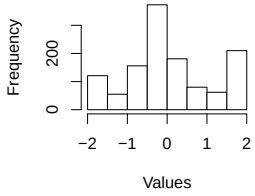
(xxxiii) ChangeInSalestoWorkingCapital



(xxxiv) ChangeInWorkingCaitalToAssets



(xxxv) ChangeInEBITDAMarginRatio



(xxxvi) ChangeInDebt

D Estimates Lasso regularization

Table 8: Estimates Lasso Regularization

The table shows the estimates of the lasso regression, based on a the penalty. Five variables are shrunken 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
ChangeInAssets	0.003635851	0.000
ChangeInRevenues	0.003635851	0.000
ChangeInDaysSalesinAccountingReceivable	0.003635851	0.000

References

Acharya, Srijana et al. (2021). "An improved gradient boosting tree algorithm for financial risk management". In: *Knowledge Management Research & Practice*, pp. 1–12.

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.

Basoglu, Kamile Asli and Clinton E White (2015). "Inline XBRL versus XBRL for SEC reporting". In: *Journal of Emerging Technologies in Accounting* 12.1, pp. 189–199.

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.

Cohen, Daniel A. and Paul Zarowin (2010). "Accrual-based and real earnings management activities around seasoned equity offerings". In: *Journal of Accounting and Economics* 50.1, pp. 2–19. ISSN: 0165-4101. DOI: https://doi.org/10.1016/j.jacceco.2010.01.002. URL: https://www.sciencedirect.com/science/article/pii/S0165410110000054.

Darmawan, I Putu Edi, Thomas Sutrisno, and Endang Mardiati (2019). "Accrual Earnings Management and Real Earnings Management: Increase or Destroy Firm Value?" In: *International Journal of Multicultural and Multireligious Understanding* 6.2, pp. 8–19.

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, Richard G Sloan, and Amy P Sweeney (1995). "Detecting earnings management". In: *Accounting review*, pp. 193–225.

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.

Henselmann, Klaus, Dominik Ditter, and Elisabeth Scherr (2015). "Irregularities in accounting numbers and earnings management—A novel approach based on SEC XBRL filings". In: *Journal of Emerging Technologies in Accounting* 12.1, pp. 117–151.

Hoffmann, Arvid OI and Hersh Shefrin (2014). "Technical analysis and individual investors". In: *Journal of Economic Behavior & Organization* 107, pp. 487–511.

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.

Huang, Xuerong Sharon and Li Sun (2017). "Managerial ability and real earnings management". In: *Advances in accounting* 39, pp. 91–104.

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.

Islam, Md Aminul, Ruhani Ali, Zamri Ahmad, et al. (2011). "Is modified Jones model effective in detecting earnings management? Evidence from a developing economy". In: *International Journal of Economics and Finance* 3.2, pp. 116–125.

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.

Jones, Jennifer J. (1991). "Earnings Management During Import Relief Investigations". In: *Journal of Accounting Research* 29.2, pp. 193–228. ISSN: 00218456, 1475679X. URL: http://www.jstor.org/stable/2491047 (visited on 06/05/2023).

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.

Kliestik, Tomas et al. (2021). "Earnings management in V4 countries: The evidence of earnings smoothing and inflating". In: *Economic Research-Ekonomska Istraživanja* 34.1, pp. 1452–1470.

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.

Larson, Chad R, Richard Sloan, and Jenny Zha Giedt (2018). "Defining, measuring, and modeling accruals: a guide for researchers". In: *Review of Accounting Studies* 23, pp. 827–871.

Li, Chunyu et al. (2021). "Chinese corporate distress prediction using LASSO: the role of earnings management". In: *International Review of Financial Analysis* 76, p. 101776.

Li, Shuangyan, Guangrui Wang, and Yongli Luo (2022). "Tone of language, financial disclosure, and earnings management: a textual analysis of form 20-F". In: *Financial Innovation* 8.1, p. 43.

Liu, Chengwei et al. (2015). "Financial fraud detection model: Based on random forest". In: *International journal of economics and finance* 7.7.

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.

McNichols, Maureen F and Stephen R Stubben (2018). "Research design issues in studies using discretionary accruals". In: *Abacus* 54.2, pp. 227–246.

Mercadier, Mathieu and Jean-Pierre Lardy (2019). "Credit spread approximation and improvement using random forest regression". In: *European Journal of Operational Research* 277.1, pp. 351–365.

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.

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.

Rushin, Gabriel et al. (2017). "Horse race analysis in credit card fraud—deep learning, logistic regression, and Gradient Boosted Tree". In: *2017 systems and information engineering design symposium (SIEDS)*. IEEE, pp. 117–121.

Rustam, Zuherman and Glori Stephani Saragih (2018). "Predicting bank financial failures using random forest". In: *2018 International Workshop on Big Data and Information Security (IWBIS)*. IEEE, pp. 81–86.

Ruwanti, Gemi, Grahita Chandrarin, and Prihat Assih (2019). "Corporate social responsibility and earnings management: The role of corporate governance". In: *Humanities & Social Sciences Reviews* 7.5, pp. 1338–1347.

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.

Sowah, Robert A et al. (2019). "Decision support system (DSS) for fraud detection in health insurance claims using genetic support vector machines (GSVMs)". In: *Journal of Engineering* 2019.

Sun, Jie et al. (2021). "Multi-class financial distress prediction based on support vector machines integrated with the decomposition and fusion methods". In: *Information Sciences* 559, pp. 153–170.

Sun, Kunpeng, Dan Wang, and Xing Xiao (2022). "Another victory of retail investors: Social media's monitoring role on firms' earnings management". In: *International Review of Financial Analysis* 82, p. 102181.

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.

Xue, Ruixiang and Hua Ding (2022). "Risk Prediction of Corporate Earnings Manipulation Based on Random Forest Model". In: *Application of Intelligent Systems in Multi-modal Information Analytics: The 4th International Conference on Multi-modal Information Analytics (ICMMIA 2022), Volume 1*. Springer, pp. 100–107.

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.