Analysing Environmental ratings' ability to model firms' emission behaviour

Thesis MSc Quantitative Finance - Erasmus School of Economics

April 30, 2022

Supervisor: K.P. de Wit

Second assessor: P.H.B.F. Franses

Author: M.G.B. Zwolle

Student number: 570478

Abstract

This study examines how well Environmental (E) ratings can be used to model firms' emission behaviour in terms of CO2 levels. This is done by first analysing the relation between E ratings and firms' CO2 emissions, and after by examining E ratings' predictability. For both analyses the data is retrieved from the commercial rating company Refinitiv which provides Environmental, Social and Governance ratings of over 9,000 companies. The effect of E ratings on firms' CO2 emissions is analysed using a pooled OLS regression and E ratings' predictability is assessed by applying the machine learning algorithms random forest, extreme gradient boosting, neural network and support vector regression. Our results show that E ratings are a driving factor behind firms' CO2 emissions and that random forest as well as XGBoost predict them with a low prediction error. Combining both results imply that E ratings can be used effectively in modelling firms' future emission behaviour and therefore, likely reduces prediction error in current models that do not include E ratings. This is beneficial as it allows countries to better forecast firms' reduction pathways and adjust regulations accordingly.

The content of this thesis is the sole responsibility of the author and does not reflect the view of the supervisor, second assessor, Erasmus School of Economics or Erasmus University.

Contents

1 Introduction

The Intergovernmental Panel on Climate Change (IPCC) is United Nations' body that assesses the science behind climate change and consists of 1,300 independent scientists from all over the world. In 2018, they concluded that in order to limit global warming at 1.5C°, CO2 emissions originating from human activity (”anthropogenic CO2 emissions”) have to decline by 45% from 2010 levels and reach net zero by 2050 (Allen et al., 2018), meaning that emission to the atmosphere (”sources”) are offset by removals from the atmosphere (”sinks”). The reason for CO2 emissions being the main driver behind climate change is due to the fact that it is the largest contributor to radiative forcing which means that there is more energy incoming than the Earth can absorb and thus heats up the surface (Myhre, Bréon, & Granier, 2018). Besides that, CO2 sticks around in the atmosphere the longest in comparison to other greenhouse gas emissions (GHG) such as Methane and Nitrous Oxide (Ciais et al., 2014). In specific, Myhre et al. (2018) show that the radiative forcing of CO2 is 73% and 833% higher than of the second and third biggest contributors Methane and Nitrous Oxide, respectively. Moreover, Ciais et al. (2014) show that Methane leaves the atmosphere in a decade and Nitrous Oxide in a century while for CO2 this is more severe i.e. 40% remains in the atmosphere for 100 years, 20% for 1000 years and 10% for 10,000 years.

This underlines the seriousness of CO2 emissions and shows that it is the driving force behind climate change. Due to the growing awareness of global warming and its impact on the environment, investors have been seeking sustainable investments. As a result, the global sustainable investments reached a total value of $ 35.3 trillion in 2020 which is 35.9% of the total assets under management and a 55% increase with respect to 2016 (Global Sustainable Investment Alliance, 2020). Due to this rising interest, a growing number of commercial rating companies have been assessing firms on environmental, social and governance (ESG) factors. These pillars contain numerous scores on underlying criteria such as emissions, equal opportunities and bribery respectively.

The E pillar is the main focus for ”ESG”, ”sustainable” and ”socially responsible” investing (Boffo & Patalano, 2020). This implies that these ratings are a determining factor in capital allocation. Next to that, Erragragui (2018) shows that firms' cost of debt is higher when having poor ESG ratings and lower when having good scores. This obviously directly affects firms' profit. Furthermore, Fatemi, Glaum, and Kaiser (2018) find a significant positive relation between the ratings and firm value, which has various effects on firms including increased share value for shareholders. Finally, Chatterji and Toffel (2010) show that firms are likely to improve their environmental performance more in comparison to firms that obtained a better rating in the previous year.

ESG ratings' influence on capital allocation, cost of debt and firm value underline its important role in the financial markets. These benefits incentives firms to improve their performance along the E, S and G axes. For the E pillar, this implies that firms have to improve their environmental performance including reducing CO2 emissions. This gives the suggesting that E ratings are a driving factor behind firms' CO2 emissions. This motivates the first part of this thesis which analyses this relation by regressing firms' CO2 emissions on E ratings and other control variables such as size. If a negative relation is found between E ratings and CO2 emissions then this implies that firms indeed reduce their CO2 emissions to obtain a higher E rating and thus that E rating is a driving force behind CO2 emissions. The second part of this thesis analyses the ability of machine learning algorithms to predict E ratings. If such a relation is found in the first analysis, then this implies that E ratings can be used in modelling firms' CO2 emissions. However, if it appears that there is much uncertainty in predicting E ratings, then including them in models that model future CO2 levels of firms induces only extra uncertainty. Hence, examining the predictability of E ratings becomes very relevant. Furthermore, if both analyses result in desirable outcomes, i.e. finding a significant relation between E ratings and firms' CO2 emissions together with a high predictability, would imply that E ratings can be incorporated effectively in modelling firms' future CO2 emissions. As a consequence, reducing prediction error in models that do not include E ratings in the first place. Improving prediction error is beneficial as it enables countries to better forecast firms' emission behaviour and adjust regulations accordingly. This brings us to the following research question: To what extent can Environmental ratings be used to model firms' CO2 emission behaviour?

The data is retrieved from rater Refinitiv which covers over 9,000 companies and provides the ratings on a continuous scale. The data set contains annual data from 2002 to 2020. The relation between firms' CO2 emissions and their E ratings is examined using pooled Ordinary Least Squares and confidence intervals are bootstrapped to account for non-normality and heteroskedasticity. The predictions are made using random forest (RF), extreme gradient boosting (XGBoost), neural network (NN) and support vector regression (SVR). The hyperparameters are tuned using randomized search and bayesian optimization with 3-fold cross-validation. In addition, feature selection is employed using recursive feature selection with 6-fold cross-validation and is compared with the performance of using the full feature set. The first 15 years of the data set are used for training, the next year for validation and the last two years for testing to obtain a 70:10:20 split. As performance metrics, root mean squared error (RMSE) and mean absolute error (MAE) are normalised by dividing by the range of the target variable. Furthermore, two base models are used to compare the performances with non-machine learning methods. In particular, this consists of a naive predictor that uses the previous value as prediction and an inter/extrapolated one.

The regression analysis shows a significant negative relation between the E ratings and firms' CO2 emissions with an adjusted R-squared of .985. In particular, a unit increase on a scale from 1-100 of the E ratings leads to an expected .122% decrease in firms' CO2 emissions. The negative relation implies that firms actually adjust their environmental behaviour based on the obtained E rating and thus that E ratings are indeed a driving force behind firms' CO2 emissions. This is in line with the findings of Chatterji, Levine, and Toffel (2009).

For the predictions, it is found that random forest followed by XGBoost obtain the best results in terms of NRSME. Both of these models are found using bayesian optimisation and feature selection. In particular, these algorithms can predict E ratings with an accuracy of 94.1% and 96.8% in terms of NRMSE and NMAE respectively. Moreover, these models outperform the base models i.e. an inter/extrapolated and naive predictor by 1.4% and 2.5% respectively. Hence, this implies that these algorithms predict E ratings effectively with a low prediction error.

The remaining parts of this study are organised as follows: Section 2 reviews the relevant literature, Section 3 details the data description, Section 4 explains the research methodology, Section 5 discusses the results and Section 6 contains the discussion.

2 Literature

This section first provides a brief introduction to ESG ratings in Chapter 2.1. Then it explains why E ratings are likely a driving force behind firms' CO2 emissions and discusses related research in Chapter 2.2. After that, the existing literature is reviewed related to predicting E ratings in Chapter 2.3. Finally, the implementation of firms' emission behaviour in current CO2 models is examined in Chapter 2.4.

2.1 Brief introduction to ESG

ESG ratings are scores that represent firms' performance on environmental, social and governance issues. This is intended as guidance for investors that want to invest ”responsible” or ”sustainable”. These ratings evolved from Corporate Social Responsibility (CSR) which is the process of operating a business model that takes the impact on all aspects of society into account. ESG is built on these principles but in a manner that these practices can be evaluated numerically using a concrete set of scoring parameters. The ratings are provided by a number of commercial rating companies (”raters”) and according to ERM, a large sustainability consultancy, 600+ different ESG ratings exist as of 2018 (Wong & Petroy, 2020). While new raters are emerging, the largest raters are consolidating the raters market. This resulted in the main players being ”MSCI”, ”Sustainalytics” and ”Refinitiv” covering over 14,000, 12,000 and 9,000 companies respectively (Wong & Petroy, 2020).

The environmental (E), social (S) and governance (G) pillars have many underlying scoring criteria and together form a composite score. As explained in the introduction, this thesis focuses on the E pillar. For Refinitiv's ratings this is further categorised in the following themes:

Environmental	Environmental	Emissions Waste Biodiversity Environmental management system
	Innovation	Product innovation Green revenues, R&D, CapEx
	Resource use	Water Energy Sustainable packaging Environmental supply chain

Table 1: Themes within Refinitiv's Environmental Pillar

These subcategories (right-hand side of the table) obtain a score based on many underlying assessment topics. The full composition of the E pillar is shown in Table 27 in the appendix. The environmental, innovation and resource use scores are determined by the sum of its subcategories and the final composite E score is constructed by the weighted sum of these subpillars. The weights applied to each subpillar are based on the relative importance of each individual industry (*Environmental, Social and Governance Scores from Refinitiv*, 2021). The next chapter delves deeper into the consequences of ESG ratings and discusses their likely effect on firms' environmental behaviour.

2.2 ESG ratings' influence

The ”Principles of Responsible Investment” was launched by some of the world's largest institutional investors and supported by the United Nations in 2006. This set of principles aims to induce sustainability into the capital markets by incorporating ESG issues into the investment considerations. Currently, more than 4,600 institutional investors have signed these principles, representing over $121 trillion in assets under management (*Signatory Update*, 2021). This showcases the fast rise of using ESG ratings in the investment decisions and underlines its effect on capital allocation.

Another implication of ESG ratings is related to the cost of debt. Erragragui (2018) analyses the relationship between Corporate Social Performance (CSP) and firms' cost of debt and finds that firms' CSP affects firms' cost of debt. In this analysis, CSP is proxied by ESG ratings and is provided by rater MSCI, which constructs ratings as binary strength and concern variables. Erragragui, in particular, finds that environmental concern ratings such as Hazardous Waste, Regulatory Problems and Substantial Emissions increase firms' cost of debt whereas environmental strength ratings such as Pollution Prevention, Recycling and Clean Energy decrease firms' cost of debt. For governance strengths, the author finds a similar negative relation.

Also between ESG ratings and firm value, a significant relation is found by Fatemi et al. (2018). In specific, their results show a positive relation between ESG strengths and firm value using the same strength and concern variables of MSCI. Likewise, a negative relation is found between firm value and ESG concerns. Fatemi et al. (2018) further analyse how the amount of ESG disclosure affects the above-mentioned relations. They find that disclosure weakens the positive and negative valuation effect in the case of ESG strengths and concerns, respectively. Fatemi et al. (2018) give as a possible explanation that the market perceives disclosure as an attempt to justify overinvestments in ESG issues in case of strengths and to show their contributions and developments in improving their ESG weaknesses in case of concerns.

Alongside, Chatterji and Toffel (2010) show that firms that obtained a poor E rating in a certain year, improve their environmental performance in the next more significantly than firms that were not rated or initially rated more favourably. In addition, they show that this effect is prevalent for firms active in industries that are environmentally sensitive or that face the cheapest improvement opportunities. On the one hand, this implies that firms' ratings give firms that are poorly rated the incentive to improve sustainable performance. On the other hand, this means that firms that were rated initially well have less incentive to do so.

The above-mentioned benefits incentives firms to improve their environmental performance including CO2 reduction. Hence, this suggests that E ratings could be a driving force of firms' CO2 emissions. Chatterji et al. (2009) investigate how well E ratings capture historical environmental performance and also how well it predicts future performance. The latter is similar to examining if E ratings are a driving force behind firms' environmental performance but they interpret it from a predictability standpoint. This thesis, however, uses the model of Chatterji et al. (2009) but interprets it not in terms of predictability but as E ratings being a driving force of environmental performance based on the knowledge of more recent literature as discussed above. Moreover, Chatterji et al. (2009) use binary ratings of the predecessor of MSCI and found a significant negative relation between net environmental score (strengths-concerns) and emissions. Their analysis was conducted in 2009 and is limited to firms in America. Hence, it is interesting to investigate whether similar relations can be found for firms worldwide using the latest data and a different rater that provides continuous ratings.

2.3 Predicting E ratings

This chapter discusses the existing literature regarding the prediction of ESG ratings. First of all, it is good to note that the research on this topic, especially involving machine learning, is very limited. The majority of the research regarding ESG ratings examines whether ESG investing results in higher returns (Antoncic, Bekaert, Rothenberg, & Noguer, 2020) and whether there is divergence between ratings of different raters (Berg, Koelbel, & Rigobon, 2019; Chatterji, Durand, Levine, & Touboul, 2016). As a result of the former, studies investigated the ability of machine learning algorithms to predict stock performance (De Franco, Geissler, Margot, & Monnier, 2020; Mitsuzuka, Ling, & Ohwada, 2017). As a result of the latter, studies investigated the ability to construct new ESG measures using machine learning (Svanberg et al., 2022). However, the field of study in regard to predicting ESG ratings themselves is very narrow and to our knowledge, the research of Garcia, González-Bueno, Guijarro, and Oliver (2020) and Krappel, Bogun, and Borth (2021) are the only ones that aim to do so.

Both studies are different to this thesis as the former uses a ”rough set approach” which is a mathematical tool to discover hidden patterns in data sets (Garcia et al., 2020) and the latter examines the ability to predict ESG ratings without their previous value. However, both contain relevant information as the former shows that firm characteristics such as return on assets (ROA) and earnings per share (EPS) have predictive power in predicting ESG ratings whereas the latter shows that neural network, XGboost and Catboost are able to predict Refinitiv's E ratings with a test R squared of around 50%. Hence, the findings of Garcia et al. (2020) motivate including firm characteristics in our predictive model and the results of Krappel et al. (2021) motivate using these algorithms. However, Krappel et al. (2021) use ”fundamental data” that also includes various non-financial facts that are categorical variables which is why they include Catboost. Nevertheless in this analysis, the majority of the variables consist of continuous variables so it is chosen to leave this method out of the analysis. Finally, it is good to note that Krappel et al. (2021) do not include previous values of ESG ratings which means that comparing predictive performances is not relevant. Since we have overlapping features, i.e. firm characteristics, the same target variable and because their used algorithms obtain relatively good results, these algorithms are taken into consideration.

However, predicting E ratings using their previous value as well as firm characteristics has not been analysed before. Hence, it becomes relevant to analyse the predictive performance of this model and whether firm characteristics indeed contribute to the predictions.

2.4 CO2 models

Thus far, this thesis has been focusing on the emissions of firms specifically. However, it is also relevant to analyse how this relates to current models that forecast CO2 emissions in general. This chapter, therefore, examines which current models seem suitable for integrating firms' emission behaviour and discusses the potential consequences.

Evans and Hausfather (2018) state that the current school of CO2 modelling primarily consists of two sorts of models. The first are ”climate models” which are based on fundamental physical principles. The second ones are ”integrated assessment models (IAMs)” which analyse the effect of human development and societal choice on nature. The IAMs integrate modules on economic growth and on climate, energy and land systems to see how they interact with each other. Changes in the gross domestic product (GDP), population size and policies are often used as input to model economic prospects, emission levels, energy pathways and land use (Evans & Hausfather, 2018). The following example shows how these models work. If the population grows then food demand rises which could lead to the need for more land use, which in its turn leads to deforestation, rising prices and higher emissions. Since these IAMs involve modelling behavioural elements i.e. societal choice, it seems natural to include or integrate the emissions behaviour of firms. In addition, IAMs' setup of combining individual modules seems to allow for easy implementation of firms' emission behaviour.

Next to that, Evans and Hausfather (2018) mention that the main uncertainty in IAMs is caused by the difficulty of forecasting changes in socioeconomic behaviour. This affects economic activity which has an effect on the amount of CO2 emissions. However, modelling socioeconomic behaviour might be more difficult than modelling the CO2 emissions of firms. Of course, these two subjects are related to each other and changes in socioeconomic behaviour and economic activity affect the polluting levels of firms but it is questionable how directly this effect translates to firms. Hence, including the CO2 emission behaviour of firms could reduce the uncertainty in the IAMs.

3 Data

Both the databases of MSCI and Refinitiv are available to Erasmus students but the former consists of categorical data while the latter consists of continuous data. It is chosen to use the continuous database of Refinitiv to capture smaller differences between firms' ESG ratings. Moreover, their world market list is used, containing all equities for which ESG data is available. This section provides definitions of the variables and analyses the data on stationarity, outliers, distributions and scales. The choice for using certain variables or lags will be explained in the methodology section. The data analysis concerning the regression part is discussed in Chapter 3.1 and the prediction part in Chapter 3.2.

3.1 Firms' emission behaviour

The data set used for the regression analysis consists of yearly observations reported during the period 2002-2020. After removing zeros and NaN, the sample contains 1,729 firms which equals 14,889 company-year observations. Hence, it is a micro panel with relatively many time series of short length. In particular, the average length of all time series equals about 9 observations. This is calculated based on the number of times that a firm is used in the analysis. However in many cases, these observations are separated by missing values which makes the length of adjacent observations even shorter. Baltagi (2008) states that non-stationarity should not be a point of concern in such micro-panels. Hence, it is chosen to continue with the analysis without examining the presence of non-stationary.

CO2	Total CO2 and CO2 equivalents emissions in tonnes
ENV	Environmental pillar score
REV	Total revenues
INDUS	Industry classification

Table 2: Definitions variables regression data set

Table 2 shows the variables within this sample and provides definitions. The definitions for CO2 and REV are rather straightforward and do not need extra explanations. ENV represents the Total Environmental Score and this is chosen as E rating variable as it resembles the overall score of the underlying Emissions, Resource Use and Innovation Scores. For further explanation about the construction and composition of ENV refer back to the ”Brief introduction to ESG” chapter.

The categories of INDUS including their distributions in this data set are provided in Table 3.

Industry	Ratio
Industrials	.20
Basic materials	.14
Consumer discretionary	.13
Financials	.11
Consumer staples	.08
Technology	.07
Health Care	.06
Energy	.06
Real Estate	.05
Utilities	.05
Telecommunications	.04

Table 3: Industry distribution

One can see that firms are fairly well distributed across industries which is beneficial as this improves the generalization of the regression results for a larger variety of firms.

In this analysis, outliers are defined as observations that lie further than three times the standard deviation away from the mean. This is based on the ”three sigma rule” (Pukelsheim, 1994) which states that approximately 99.7% of the observations lay within this interval. This implies that only extreme outliers are removed from the data set and this method is chosen in order to keep as many observations as possible. This results in removing 151 observations which is around 1%. Choosing a ”two sigma rule” or ”interquartile range rule” (Vinutha, Poornima, & Sagar, 2018) results in removing far more observations. Table 4 shows the descriptive statistics of the resulting data set.

	CO2$_t$	CO2$_{t-1}$	ENV$_{t-1}$	REV$_t$
Mean	1.8E5	1.8E5	62.7	6.8E8
Std	1.1E6	1.0E6	20.8	2.5E9
25%	2.1E2	2.1E2	48.5	5.5E6
50%	1.7E3	1.6E3	65.3	2.4E7
75%	2.1E4	2.0E4	79.5	2.3E8

Table 4: Descriptive statistics

Table 4 shows that all variables except ENV$_{t-1}$ have a mean that is larger than the 75th percentile which implies that these variables have a distribution that is positively skewed. Hence, these variables are logarithmically transformed in order to normalise the data. Next to that, one can see that the scales of these variables differ by a great amount which for comparison reasons, has to be accounted for. This is done by standardization.

Table 5 is the correlation matrix and shows that $\mathrm{CO2}_{t-1}$ and REV_t are relatively high correlated. This is understandable as revenue is often related to size and a larger size implies more CO2 emissions. The other variables are low correlated.

	$\mathrm{CO2}_{t-1}$	ENV_{t-1}	REV_t
$\mathrm{CO2}_{t-1}$	1.00	.04	.79
ENV_{t-1}		1.00	0.11
REV_t			1.00

Table 5: Correlation matrix

3.2 Prediction

The data set used for the predictions also consists of yearly observations reported from 2002-2020. After removing zeros and NaN, the sample consists of 2,899 firms which equals 22,475 company-year observations. Table 6 shows the variables within the prediction data set and includes definitions.

ENV	Environmental pillar score
PE	Price-to-earnings ratio
PtB	Price-to-book ratio
EV	Enterprise value
ROE	Return on equity
ROA	Return on assets
DEBT	Total debt
EpS	Earnings per share
ROI	Return on invested capital
GP	Gross profit margin
REV	Total revenues
INDUS	Industry classification
CTRY	Country of domicile

Table 6: Prediction data set including definitions

Similar as before, Tables 7 and 8 show that the firms are relatively well distributed across industries and countries. This is beneficial as it improves generalization, enabling a prediction model that is compatible with a larger variety of firms.

Industry	ratio
Industrials	.23
Consumer discretionary	.17
Basic materials	.10
Consumer staples	.09
Utilities	.07
Health Care	.06
Technology	.06
Real Estate	.06
Energy	.06
Telecommunications	.05
Financials	.03

Table 7: Industry distribution

Industry	ratio
US	.21
JP	.16
GB	.10
CA	.05
FR	.05
DE	.04
AU	.03
CH	.03
HK	.03
IN	.02
Other	.30

Table 8: Country distribution

The descriptive statistics for E ratings in particular are shown in Table 9. It is striking to see that on average firms improved their environmental behaviour as the mean of the E ratings increased over the years. In addition, the standard deviation decreased over the past 10 years which indicates that more firms are taking environmental issues seriously and are aiming to improve their environmental performance.

	2020	2019	2018	2017	2016	2015	2014	2013	2012	2011	2010	2009	2008	2007	2006	2005	2004	2003	2002
Mean	57.0	54.5	50.4	44.3	40.3	37.7	34.3	33.0	32.3	30.5	28.4	24.3	21.1	16.2	11.1	10.1	6.7	3.8	3.4
Std	22.9	23.8	25.3	28.6	29.9	30.5	30.7	31.0	31.1	31.0	30.8	30.1	28.6	25.7	21.5	20.6	16.9	13.0	12.6
25%	40.7	36.9	30.4	20.0	11.0	4.3	34.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	59.7	56.7	51.9	46.9	41.5	37.7	31.7	28.3	26.7	22.6	18.0	4.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	75.4	73.8	71.1	68.3	66.3	64.3	61.4	60.3	59.0	57.5	55.2	48.1	42.4	29.5	10.6	4.60	0.0	0.0	0.0

Table 9: Descriptive statistics

Figure 1 shows the time series of the firms that obtained an E rating each year during the full period. This graph indicates that the time series are likely to be heteroskedastic and non-stationary. The former should not be a problem for machine learning methods as its validation is based on examining the performance on a test set and not on calculating confidence intervals or performing statistical tests for which such assumptions are necessary. However, it does advocate for including a trend variable in the analysis to capture differences over the years. The latter is further examined using a Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test which is chosen based on the statement of Fedorová et al. (2016) that it performs relatively well on small data sets.

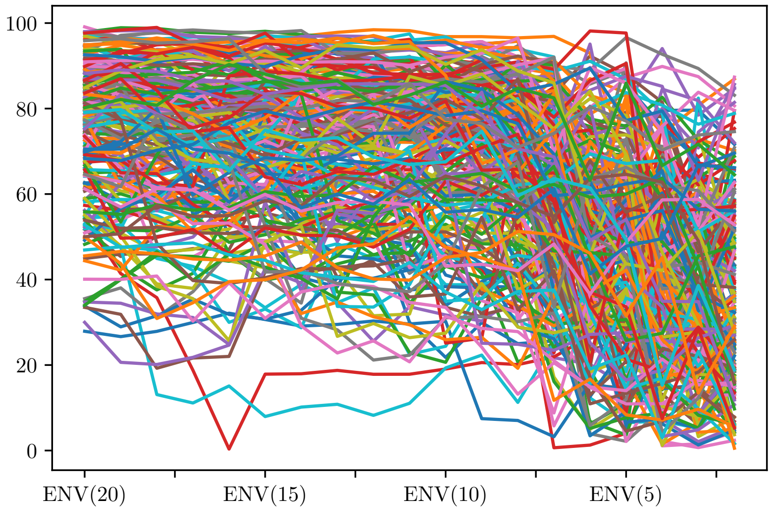

Figure 1: Fully 19-year reported time series

Performing the KPSS test on the E series results in rejecting 45.3% which implies non-stationary time series. This test is run again on their differences as well as on their logarithmic differences. The former results in rejecting 8.0% and the latter in 5.6%. Hence, it is chosen to use logarithmic differences and to remove the non-stationary time series from the data set. This results in removing 1,194 observations (e.g. 5-year time series contain 4 observations due to one step ahead predictions) which equals 5.3% of the data set.

Outliers are removed using the same method as before, i.e. based on the three-sigma rule. This results in removing 931 observations from the total 21,281 which equals 4.4%. Table 10 shows the descriptive statistics of the resulting data set.

	ΔENV_{t+1}	ΔENV_t	PE_t	PtB_t	EV_t	ROE_t	ROA_t	DEBT_t	EpS_t	ROI_t	GP_t	REV_t
Mean	.08	.10	24.1	2.9	4.6E8	15.4	6.9	1.6E8	58.0	10.0	36.6	3.9E8
Std	.35	.40	30.1	5.9	1.9E9	16.9	5.2	6.4E8	290.1	7.8	21.1	1.5E9
25%	-.03	-.04	12.6	1.3	6.5E6	7.4	3.6	1.1E6	1.1	5.1	20.3	3.2E6
50%	.02	.02	17.9	2.0	2.5E7	12.9	6.0	5.1E6	3.1	8.5	32.6	1.3E7
75%	.13	.15	25.3	3.3	1.9E8	20.4	9.4	3.3E7	23.2	13.6	49.5	1.1E8

Table 10: Descriptive statistics prediction set

It is interesting to see is that many features have a mean that lies outside the 25th and 75th percentile indicating that these variables have a distribution that is skewed. It is therefore analysed whether a logarithmic transformation normalises the distributions. This is the case for PE_t, Debt_t, REV_t and EpS_t. Subsequently, it is compared whether these transformations lead to better results which appeared to be the case. Hence, it is chosen to use the transformed variables. Furthermore, Table 10 shows that the scales of the variables differ greatly which advocates for scaling the data which is done by applying standardization. In particular, this is done for the training, validation and test set separately such that no information of the validation and test set is used for training.

Table 11 shows the correlation matrix of the training set and indicates that most features are low to moderate correlated. The most correlated features are ROA_t with ROI_t. This is probably due to the fact that investments are accounted for on the balance sheet as assets. Hence, return on assets is very related to return on invested capital. Similarly, REV_t is highly correlated with DEBT_t as more funding allows firms to expand more rapidly and grow which is then translated into more revenue. Similar explanations can be given for the other moderate to high correlated features: (ROE_t & ROI_t), (ROE_t & ROA_t), (EpS_t & REV_t).

	ΔENV_{t+1}	ΔENV_t	PE_t	PtB_t	EV_t	ROE_t	ROA_t	DEBT_t	EpS_t	ROI_t	GP_t	REV_t	TREND_t
ΔENV_{t+1}	1.00	-.13	.02	.01	-.00	.01	.02	-.02	-.04	.02	-.02	-.02	-.08
ΔENV_t		1.00	.02	.01	-.01	.02	.04	-.04	-.04	.03	-.01	-.03	-.09
PE_t			1.00	.10	-.01	-.01	.00	-.06	-.17	-.00	.07	-.05	.07
PtB_t				1.00	-.03	.25	.18	-.10	-.05	.22	.07	-.08	.03
EV_t					1.00	-.05	-.04	.41	.37	-.06	-.03	.45	.01
ROE_t						1.00	.68	-.15	-.06	.75	.15	-.10	-.05
ROA_t							1.00	-.31	-.09	.93	.28	-.18	-.05
DEBT_t								1.00	.42	-.35	-.15	.82	.03
EpS_t									1.00	-.08	-.08	.56	-.01
ROI_t										1.00	.20	-.16	-.08
GP_t											1.00	-.28	.05
REV_t												1.00	-.01
TREND_t													1.00

Table 11: Correlation matrix training set

However, that is less important for this analysis. What is important to note is that all the features, except of (ROA_t & ROI_t) are not close to being perfectly correlated. It gives the idea that one of these variables could be removed during feature selection. This will be analysed along with removing other highly correlated features.

4 Method

4.1 Firms’ emission behaviour

This section (4.1) discusses the methods used in order to analyse the relation between E ratings and firms’ CO2 emissions. This section is organised as follows: it first explains the setup of the regression model and after, discusses an additional bootstrapping method.

4.1.1 Regression model

The relation between E ratings and firms’ CO2 emissions is assessed by conducting the following regression:

$$\log \mathrm{CO2}_{i,t} = \beta_0 + \beta_1 \log \mathrm{CO2}_{i,t-1} + \beta_2 \mathrm{ENV}_{i,t-1} + \beta_3 \log \mathrm{REV}_{i,t} + \beta_4 \mathrm{INDUS}_i + \beta_5 \mathrm{Trend}_t + \epsilon \quad (1)$$

Refer back to Chapter 3.1 for definitions of these variables. Also mentioned in that chapter is that $\mathrm{CO2}_{i,t}$, $\mathrm{CO2}_{i,t-1}$ and $\mathrm{REV}_{i,t}$ are skewed and therefore logarithmic transformed. INDUS_i is dummy encoded and the category ”Telecommunications” is left out of the analysis. Trend_t is a trend variable which implies that the year 2003 equals the value 1 and the year 2020 equal the value 18.

These variables are chosen based on the regression model of Chatterji et al. (2009). Model 1 differs from their model in regard to the variables $\mathrm{CO2}_{i,t}$ and $\mathrm{ENV}_{i,t-1}$. To start with the former, Chatterji et al. (2009) define an environmental performance variable as dependent variable that includes other indicators next to CO2 emissions such as ”number of violations”. This thesis chooses to only focus on CO2 emissions in particular as it is the main contributor to climate change as discussed in the introduction.

Another difference is that they measure emissions as toxic chemicals reported to ”US EPA’s Toxic Release Inventory” (Chatterji et al., 2009). Moreover, they assume that there is a delay between this data becoming publicly available and when it is actually reported. They, therefore, use second-order lags of all independent variables. Their research was conducted in 2009 which makes this reasoning understandable. However, nowadays firms have to comply with many reporting regulations and raters are likely to have progressed with their data collecting methods. Hence, it is assumed that such delays are not an issue anymore. Moreover, in our model the E ratings are included as first-order lag because it is assumed that firms are able to change their

CO2 emissions significantly within one year. Next to that, $REV_{i,t}$ is included with the same time index as the dependent variable has as it is a control variable which is related to the CO2 emissions in year t.

The last difference is related to the E ratings itself. In particular, Chatterji et al. (2009) use the binary ratings provided by MSCI's predecessor whereas continuous ones of Refinitiv are used in this analysis.

4.1.2 Violated assumptions

It appears that not all assumptions such as normality and homogeneity are met. At first thought, it was analysed whether these assumptions could be met by performing linear transformations. However, E ratings appeared to be left-skewed and normalising this variable by transformations implies using squared transformations which makes the interpretation more difficult and becomes even more involved when the dependent variable is logarithmic transformed. Therefore, it is chosen to use bootstrapping to calculate confidence intervals of the regression coefficients. In specific, this involves bootstrapping random response-predictor pairs and performing ordinary least squares (OLS) to each run. This is done 10,000 times and the regression coefficients and adjusted R-squared are saved in a new data frame. Next, the 95% confidence intervals are calculated for all independent variables and the constant as well as for the adjusted R-squared. This method circumvents the violation of homogeneity and normality assumption as it is shown that this is asymptotically similar to performing a Huber-White heteroskedasticity correction and obtaining normally distributed residuals (Cribari-Neto & Zarkos, 1999).

4.2 Prediction

This section (4.2) discusses all methods related to predicting the E ratings. The E ratings are provided on a 0-100 scale. This task could be transformed into a classification problem but it seems more valuable to predict it as numeric values since it is then better able to capture smaller differences in companies' ratings. This implies that the task at hand becomes a numeric prediction problem and since the data includes label variables this is part of supervised learning. This section contains many chapters and for overview purposes, its layout is not discussed.

4.2.1 Setup

Using the full feature set for making predictions results in the following prediction model:

Target variable	$\Delta \log \mathrm{ENV}_{t+1}$	Differentiated $\log$ E ratings
Features	$\Delta \log \mathrm{ENV}_t$	Differentiated $\log$ E ratings
	PE_t	Price-to-earnings ratio
	PtB_t	Price-to-book ratio
	EV_t	Enterprise value
	ROE_t	Return on equity
	ROA_t	Return on assets
	DEBT_t	Total debt
	EpS_t	Earnings per share
	ROI_t	Return on invested capital
	GP_t	Gross profit margin
	REV_t	Total revenues
	INDUS_t	Industry codes
	CTRY_t	Country of domicile
	TREND_t	Trend variable

Table 12: Prediction model including definitions

A recursive multiple-step forecast is chosen over a direct multiple-step forecast as ratings are shown to be dependent on their previous rating (Chatterji et al., 2016). Hence, the first lag of the dependent variable is included in the model. Choosing a direct multiple-step forecast instead would imply that these dependencies are not correctly modelled. The other features are chosen as they reflect many firm characteristics such as size (REV_t), profitability (EpS_t & (GP_t)), management effectiveness (ROE_t, ROA_t, ROI_t), valuation (PE_t, PtB_t, EV_t) and leverage (DEBT_t). The other features are chosen to capture differences between industries, countries and years.

The model shown in Table 12 is put in tabular form for all observations. This implies that each row represents one observation, i.e. the following:

Apple	$\Delta \log \mathrm{ENV}_{2020}$	$\mathrm{Features}_{2019}$
	$\vdots$	
	$\Delta \log \mathrm{ENV}_{2003}$	$\mathrm{Features}_{2002}$
Amazon	$\Delta \log \mathrm{ENV}_{2020}$	$\mathrm{Features}_{2019}$
	$\vdots$	
	$\Delta \log \mathrm{ENV}_{2015}$	$\mathrm{Features}_{2014}$
	$\vdots$	
	$\vdots$	

Table 13: Schematic overview of tabular form of the prediction model

It is chosen to stack all observations like this to obtain a machine learning model that is optimised for all time series and not for each time series individually. This results in a model that better generalizes and hence, is compatible with a larger variety of inputs.

This further implies that missing values do not affect the full length of a time series as only two adjacent values are necessary to be used as observation. This means that a time series from 2015 to 2020 with a missing value in 2018 still contains 3 observations, i.e. target years: 2016, 2017, 2020 and feature years: 2015, 2016 and 2019. Due to this limited effect of missing values on the sample size, it is chosen to handle missing values by excluding them from the data set.

4.2.2 Handling categorical features

CTRY_t is transformed into a new variable where countries are grouped together based on their GHG per \$ of GDP which is retrieved from The World Bank (“CO2 emissions”, 2022). In specific, the countries are categorised into nine groups: Low, Upper Low, Moderate, Upper Moderate, Average, Upper Average, High, Upper High and Extreme. This is done to lower the number of categories such that dummy encoding can be applied, i.e. having 54 dummies for CTRY_t alone is not favourable.

The resulting variable as well as INDUS_t are dummy encoded. In particular, the categories ”Low” and ”Financials” are left out of the analysis. The trend variable TREND_t consists of integers ascending to 18, representing the years 2002 to 2019. This is done to capture possible trends.

4.2.3 Splitting the data set

The features of the data set are reported from 2002 to 2019 and are used to make predictions from 2003 to 2020. The first 16 years, i.e. 2002 to 2017, are used as full training set (training + validation) and the last 2 years as test set, i.e. 2018 to 2019. The split is made in this way to satisfy the conventional 80:20 ratio. This results in relatively good results in comparison with other splits. Moreover, for the train-validation split a 90:10 ratio is used in order to have an approximate 70:10:20 training, validation and test split respectively. This results in 14,216, 1,789 and 4,345 observations within the training, validation and test set respectively.

4.2.4 Feature selection

For feature selection, recursive feature elimination with cross-validation (RFECV) of Scikit-Learn (Pedregosa et al., 2011) is applied. For explanation purposes, this algorithm is split into its two components. The first part contains the ”Recursive feature elimination” (RFE) which implies that an estimator is fitted using all features and subsequently, that the feature with the lowest importance is eliminated. Then using the new subset of features, this process is reiterated. This continues until the minimum number of features is reached, which is a value you set yourself (3 by default). The second part contains the ”k-fold cross-validation” which implies that the training set is split into k folds and that every fold is used once as test set while the training occurs on the other k-1 folds. This results in k individual models and their performance is averaged to obtain the final performance.

RFECV uses a combination of both. In particular, it uses cross-validation to find the best number of features. It does this by splitting the training data into k folds and applying RFE on each fold. The RFE algorithm then selects the best variables when using n, n-1, n-2, ..., 3 number of features (N). Since this is done for each fold, there are now k subsets of best-selected features for each possible N. This implies that for each N, the estimator is fitted on the k fold and evaluated on the other k-1 folds using the root mean squared error (RMSE). Hence, this results in k performances per N. The performances are then averaged across the k folds to obtain a final score for each N. The N that obtains the highest performance is then selected as best. Finally, the algorithm applies RFE to the entire training data set but now iterates until optimal N is reached.

In this thesis, the RFECV algorithm is run with random forest and XGboost as estimator. During hyperparameter tuning, the performances of the models using the full feature set is compared with the selected feature set.

4.2.5 Hyperparameter tuning

The tuning of hyperparameters is done by randomized search and bayesian optimisation using the RandomizedSearchCV and BayesSearchCV modules of the Scikit-Optimize API respectively. Randomized search takes random combinations of hyperparameters of the parameter grid. Bergstra and Bengio (2012) show that randomly chosen trials are more efficient than grid-search. Not only is it less computational expensive but it appears to also to find models that are as good or better than the ones found with grid-search. In addition, they state that randomized search can be used as a good baseline against other hyperparameter optimization algorithms.

The objective function in bayesian optimisation for hyperparameter tuning is finding the set of hyperparameters that minimises the validation metric, which is the RMSE in this case. Furthermore, it uses a surrogate function that aims to reflect the objective function. Then, it samples more observations close to local minima. Having more observations, the algorithm updates the surrogate function accordingly. These surrogate functions are reflected by Gaussian processes and many functions are fitted based on these data points, which have probabilities attached to them. The bayesian part is that it puts the surrogate functions as probabilistic distributions which are updated when inserting observations.

The performance of each tested hyperparameter combination is evaluated using k-fold cross-validation. This improves generalization as the hyperparameter set is selected that performs well on different subsets of the data set. In this analysis, 3-fold cross-validation is chosen for all randomized and bayesian searches. Furthermore, a wide range of values for each hyperparameter is used for both the randomized and bayesian search to capture as many different combinations as possible. The outer values of these ranges are used as limits for the bayesian optimization. Next to that, it is chosen to test 20 different hyperparameter combinations. This is relatively low as the total possible combinations for e.g. random forest are 4,800 which means that less than 0.5% of the entire parameter space is assessed. However, it is chosen to value generalisation over hyperparameters optimization in order to obtain better results when inserting new data. In addition, these settings already obtain good results and this, therefore, does not advocate for increasing the number of combinations. On top, the additional computational time that would be involved makes this choice justifiable.

4.2.6 Evaluation

The performance of all algorithms and their models with different hyperparameters are compared based on the root mean squared error (RMSE). For hyperparameter tuning the performance is calculated only on the training set which means that the validation and test set are left out of the analysis. After finding the best hyperparameter sets per algorithm, the model is trained again on the full training set (training + validation set) and tested on the test set.

The RMSE differs from the mean absolute error (MAE) as the RMSE penalises larger prediction errors more than MAE does. This is preferable as this induces milder errors. However, for the interpretation the MAE is included as well. Both the RMSE and MAE are normalised (called NRMSE and NMAE) by dividing by the range of the target variable in the training and test set separately and are multiplied by 100 to obtain percentages. This is done because it gives a better understanding of whether a certain RMSE or MAE score is good. For example, a RMSE of .5 when the target variable ranges between 0 and 1000 is much better than when it ranges between 0 and 100. The construction of the evaluation metrics are explained below.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \quad \text{and} \quad MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{2}$$

$$NRMSE_{test} = RMSE/[y_{test}] * 100\% \quad \text{where} \quad [\cdot] = range$$

The same concept of equation 2 applies to $NMAE_{test}$ and for $NRMSE_{train}$, the y_{test} within the $[\cdot]$ is substituted by y_{train}. Note that "train" now refers to the full training set (training plus validation). This is because the final performances are evaluated on the test set while training on the full training set.

The optimised models are compared with two base models, one is a naive predictor which sets the predictions equal to the previous value and the other one makes predictions by inter- and extrapolation. These models are further explained at the end of chapter "Regression algorithms" (4.2.8).

4.2.7 SHAP values

After selecting the best performing model, SHapley Additive exPlanations (SHAP) for tree-based models (Lundberg, Erion, & Lee, 2018)) are used to interpret the effect of the features on the target variable. This method computes Shapley values from coalitional game theory (Shapley, 1953). This method implies that coalitions of features are constructed by adding one feature at a time, i.e. all possible combinations when having two, three, etc features (F). Then the SHAP algorithm trains a model using each individual coalition and subsequently makes predictions. This implies that one can calculate the difference between the predictions using e.g. two and three features again using all feature combinations for these numbers. Hence, one can calculate the change in predictions when adding one feature. However, adding a certain feature to a coalition happens multiple times as all possible combinations of features are used as coalitions. The total marginal contribution of a feature is therefore constructed as the weighted sum of all its marginal contributions. These weights are equal to the reciprocal of the number of possible marginal contributions per F level. By way of example, suppose there are 3 features. The first coalition contains zero features and the next ones contain one feature. This implies that the weight is 1/3 as there are three edges from going from zero features to one feature. Similarly, from one feature to two features there are 6 edges which makes the weight equal to 1/6. The SHAP values of each feature are thus equal to their marginal contribution constructed as described above.

4.2.8 Regression algorithms

In the coming section, the algorithms are described that are used in this thesis. In particular, these are random forest (RF), extreme gradient boosting (XGB), neural network (NN) and support vector regression (SVR), which are all suitable for regression.

Random Forest

A random forest (Breiman, 2001) is an ensemble learning method that is part of the bagging methods which implies that regression trees are grown in parallel and that there is no interaction between the trees when constructing them. Subsequently, the predictions of each tree are averaged to obtain a final prediction.

Random forest is known to obtain good results by the default settings. However, a small performance gain is still achievable by tuning the hyperparameters as shown in (Probst, Wright, & Boulesteix, 2019). Table 14 shows the tuned hyperparameters of random forest including definitions, their effect and other important notes. Boehmke and Greenwell (2019) state that the number of estimators and maximum features often have the largest effect on predictive performance followed by a moderate effect of the others. The tested values of the hyperparameters are chosen primarily based on the literature of Boehmke and Greenwell (2019); Géron (2019); Probst et al. (2019). However, the tested range for each hyperparameter is extended if the results indicate that a performance gain is obtainable, e.g. if there is a relatively large gap between test and train error. This is also done for the other algorithms.

Number of estimators	[300, 500, 800, 1000]
The number of trees that are grown before taking averages for prediction	1. A larger value improves performance but takes more computational time
Maximum features	$[\frac{p}{3}, \sqrt{p}, p],\ p = \#$ predictors
The number of randomly drawn candidate variables which are then used to make the splits of each tree	1. A small value implies that the trees are split using only a few predictors and causes the trees to be less correlated with each other. This trades a small bias for a lower variance which generally yields a better performance (Géron, 2019) 2. A lower value could also lead to worse performance as suboptimal variables could be chosen when building the trees (Probst et al., 2019)
Maximum depth	[1, 5, 8, 10, 25, 30]
The maximum size of each tree	1. The deeper the tree, the more splits it has and the better it is able to capture information 2. A too deep tree is likely to induce overfitting whereas a too shallow tree can lead to underfitting (Boehmke & Greenwell, 2019)
Minimum samples split	[2, 5, 10, 100]
The minimum number of observations required to split an internal node	1. Reduces overfitting by controlling the depth of the trees
Minimum samples leaf	[1, 2, 5, 10]
The minimum number of observations at a terminal node (Probst et al., 2019)	1. Similarly to minimum samples split, reduces overfitting by controlling the tree depth 2. Larger values reduce computational time (Boehmke & Greenwell, 2019)
Maximum samples	[0.25, 0.5, 0.75, 1]
The sampling scheme for growing each tree	1. The default sample scheme is bootstrapping 100% of the observations with replacement 2. Decreasing the sample size induces more variability between the trees and less between-tree correlation which can improve prediction performance (Boehmke & Greenwell, 2019)

Table 14: Random forest's tuned hyperparameters; Each single line belongs to a hyperparameter and relative to each line it can be described as follows. Upper left = hyperparameter name, upper right = tested values, lower left = definition, lower right = it's effect and other relevant notes

EXtreme Gradient Boosting

EXtreme Gradient Boosting (XGBoost) introduced by Chen and Guestrin (2016) is a variant of gradient boosting of (Friedman, 2001). Boosting implies that the algorithm is run sequentially. Moreover, it constructs a base tree with a single root node and makes an initial prediction. Then it constructs another tree from the errors of the previous tree. These errors are minimised by the gradient descent algorithm and each new tree is scaled by the learning rate to determine its contribution. This process iterates itself until the errors do not further decline by adding more trees. Subsequently, the predictions are made using all grown trees but their contribution is now determined by their individual importance rather than having equal importance as with random forest.

XGBoost extends this algorithm by adding L1 (Lasso) and L2 (Ridge) regularisation terms, by using the second-order derivative of the loss function instead of the first and by performing the computations in parallel. This results in a faster and better performing algorithm. Table 15 shows the tuned hyperparameters for XGBoost and their tested values are chosen based on the literature of Bengio (2012); Brownlee (2019); Friedman (2001, 2002).

Number of estimators	[5, 20, 60, 100, 150, 180, 200, 250, 300, 350, 400]
The number of decision trees used in the boosting algorithm	1. In contrary to bagging methods, adding trees to boosting algorithms does not lead to an ever-increasing performance gain due to its sequential structure (Brownlee, 2019). Hence, relatively smaller values are tested
Learning rate (eta)	[0.01, 0.015, 0.025, 0.05, 0.1, 0.15]
A shrinkage parameter applied to the feature weights after each boosting step	1. A small value results in making fewer corrections per tree added to the model (Brownlee, 2019) and hence reduces overfitting 2. The smaller the learning rate, the higher the computational time
Maximum tree depth	[1, 3, 5, 7, 15]
The maximum size of the trees	1. Gradient boosting algorithms generally perform well with trees that have modest depth (Brownlee, 2019). Hence, the relatively smaller tested values
Minimum child weight	[1, 3, 5, 7, 10, 15, 20]
The minimum sum of instance weight needed in a leaf node	1. Reduces overfitting and setting a higher value means trading a larger bias for a smaller variance
Minimum split gain (gamma)	[0.05, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0]
The minimum loss reduction required for a leaf node to be further partitioned	1. This hyperparameter also helps to reduce overfitting as it controls tree depth
Row sampling (subsample)	[0.2, 0.4, 0.6, 0.8, 1.0]
The fraction of observations to be randomly sampled without replacement before growing each tree	1. It helps reduce overfitting as the trees are then grown using different subsamples of the training set
Column sampling (colsample)	[0.2, 0.4, 0.6, 0.8, 1.0]
The fraction of columns, i.e. features, to use for creating each tree	1. Similar effect as row sampling and thus prevents overfitting 2. Can be done by tree, by split and by node, but only column sampling by tree is used as it is shown to be sufficient in many cases (Brownlee, 2019)
Lambda and alpha	[0.2, 0.4, 0.6, 0.8, 1.0] [0, 0.1, 1, 10]
L2 and L1 regularization factors on the weights respectively	1. L2 shrinks the weights towards zero whereas L1 induces sparsity

Table 15: XGBoost's tuned hyperparameters; Similar structure as before

Neural Network

An (artificial) neural network can be described as a network of interconnected neurons organised in layers, i.e. input, hidden and output layers. The neurons in the hidden layers receive the weighted sum of the output of the neurons in the previous layer and process this via activation functions to new output for the neurons in the next layer. The output layer processes the output of the last hidden layer through an output activation function to obtain the final prediction. The network is trained by minimising a loss function, i.e. MSE for regression, and updates the weights of the neurons accordingly. Tables 16 & 17 provide a detailed explanation of the tuned hyperparameters and also gives a better understanding of how a neural network works. Again, the range of tested values is chosen based on the literature, which in this case are the ones of Géron (2019); Goodfellow, Bengio, and Courville (2016). A final remark in regard to the regularization parameters λ and α which target the weights and the activations respectively. It is chosen to not use these parameters as other forms of regularization, i.e. early stopping and dropout, are already applied.

Learning rate	[1e-4, 1e-3, 1e-2, 1e-1, 2e-1]
The stepsize of each update iteration	1. Setting a too large value can lead to overshooting the minimum of the loss function whereas a too small value can take up extremely large computational time
Optimizer	[Adam, Nadam]
The optimizer that updates the weights to minimise the loss function	1. Both Adam (Kingma & Ba, 2014) and Nadam (Dozat, 2016) are adaptive learning algorithms and therefore require less tuning of the learning rate parameter 2. Nadam involves using Nesterov momentum which is a form of momentum and heads the optimizer in the right direction and accelerates the training process
Batch Size	[64, 256, 512, 1024]
The size of randomly selected and disjoint subsets of the training data that go through the network	1. Mostly affects computational time but also induces a form of regularization Bengio (2012) and hence improves generalization. This is because small batches have more variation from one another such that the convergence rate and direction are more variable 2. This hyperparameter is very useful within this thesis' training strategy, i.e. 3-fold cross-validation, since this strategy can take up large computational time
Number of epochs	[10, 50, 100, 200, 400]
The number of times that the algorithm sees the entire data set	1. It is very related to batch size, e.g. with a sample size of 1,000 observations and a batch size of 500, it means that 2 iterations are needed to complete 1 epoch. Hence, setting a low batch size with a high number of epochs increases computational time 2. Setting a too high value induces overfitting while a too low value induces underfitting
Number of hidden layers	[1, 2, 3]
The number of layers between the input and output layers that contain neurons	1. Goodfellow et al. (2016) mention that with one hidden layer, the neural network can approximate any function that is required and is therefore sufficient in most cases
Number of neurons	[1, 5, 7, 11, 14, 17, 20, 23, 25, 28, 30]
The number of neurons per layer that process inputs through activation functions into outputs	1. Controls the capacity of the model 2. Larochelle, Bengio, Louradour, and Lamblin (2009) found in a large comparative study that using the same number of hidden units for all layers works at least as good as using an increasing or decreasing size. Therefore, it is chosen to keep the size constant in this analysis

Table 16: Neural network's tuned hyperparameters part (1/2); Similar structure as before

Activation functions	[ReLU, ELU]
Functions that define how the input of nodes as weighted sum is transformed into output for each neuron in a layer in the network	1. Rectified linear unit (ReLU) and exponential linear unit (ELU) are preferred over the sigmoid and hyperbolic tangent as it is not subject to vanishing or exploding gradients which happens for very small and large inputs Géron (2019) 2. ReLU is subject to another problem that is called ”dying ReLU” which basically means that some neurons die due to fact that this activation function outputs zeros for negative input 3. ELU solves this problem by having negative-valued outputs following an exponential function such that the gradient is nonzero 4. In the output layer, a simple linear activation function is used as recommended in (Sharma, Sharma, & Athaiya, 2017)
Weights initialization	[He uniform, He normal]
The starting point of the optimisation process	1. It can prevent the output of the activation functions to explode or vanish. This can lead to convergence problems as the gradient of the loss function might then be too small or too large to flow backwards through the network optimally (He, Zhang, Ren, & Sun, 2015) 2. ”Kaiming He Initialization” (He et al., 2015) is now the standard when using ReLU or its variants. It initialises the weights by taking random numbers from a Gaussian distribution with mean zero and standard deviation of $\sqrt{\frac{2}{n}}$, where n is the number of inputs to the node 3. This is compared with ”He uniform” which is similar to ”He normal” but takes random numbers from a uniform distribution with limits -l and l, where $l = \sqrt{\frac{6}{\text{fan\_in}}}$ and fan_in is equal to the number of input units per layer
Early stopping	
Regularization technique that stops training early when the validation loss is not decreasing sufficiently	1. It reduces overfitting 2. It stops when the validation loss is not decreasing with more than 0.01 over 10 epochs
Dropout	[0.1, 0.2, 0.3]
Regularization technique that randomly deletes units and their connections during training (Géron, 2019)	1. It reduces overfitting 2. It is placed just after the input layer

Table 17: Neural network’s tuned hyperparameters part (2/2); Similar structure as before

Support Vector Regression

Support vector regression (SVR) (Cortes & Vapnik, 1995) aims to fit a hyperplane that contains most data points. It does this by defining an ϵ-insensitive tube that indicates how much error is tolerated, i.e. how far the observations can lie from the hyperplane without correcting for them. However, the error with respect to the observations outside of the tube is minimised while regularizing their importance (hyperparameter C) to reduce model complexity. Another important feature of SVR is called the ”kernel trick”, this refers to a function that maps linearly inseparable input to a higher dimensional feature space which leads to linearly separable data. This together with the explanations of other hyperparameters is further discussed in Table 18.

Again, the tested values of the hyperparameters are chosen based on the literature of Cherkassky and Ma (2004); Smola and Schölkopf (2004). Furthermore, it is chosen to keep the bias parameter ”c” and the degree of the polynomial ”d” equal to the default values 0 and 3 respectively, as this is shown to obtain good results (Cherkassky & Ma, 2004).

A final remark, it is chosen to reduce the sample size to 2,000 observations when using randomized search in order to improve computational time. This is done by subsampling without replacement.

ϵ	[1e-3, 1e-2, 1e-1, 1e0]
The width of the ϵ-insensitive tube	1. A larger value of ϵ induces more error but reduces running time (Smola & Schölkopf, 2004) 2. A larger value implies that fewer support vectors are used to fit the data which makes the estimates flatter (less complex) (Cherkassky & Ma, 2004)
C	[1e-1, 1e0, 1e1, 1e2]
The relative importance measure that controls the trade-off between the regularization term and the empirical error (Smola & Schölkopf, 2004)	1. When C is large, the SVR algorithm tends to be overfitting and lead to high computational time while a too small value of C can lead to underfitting. Thus, ϵ and C both control model complexity but in different ways (Cherkassky & Ma, 2004)
Kernel function	[Linear, Polynomial, RBF, Sigmoid]
Function that transforms linearly inseparable data to separable ones by mapping the data to a higher dimensional space	1. Linear: $K(x_i, x_j) = x_i' x_j$, with x_i, x_j being vectors of the input space 2. Polynomial: $K(x_i, x_j) = (\gamma x_i' x_j + c)^d$ γ = scaling factor explained below c = bias parameter d = degree of the polynomial 3. Radial basis function (RBF): $K(x_i, x_j) = exp(-\gamma \|\|x_i - x_j'\|\|^2)$ 4. Sigmoid: $K(x_i, x_j) = \tanh(\gamma x_i' x_j + c)$
Kernel parameter (γ)	$[\frac{1}{\text{Var(X)}*p}, \frac{1}{p}]$, X = features and p = # predictors
It is a scaling factor that controls how far the influence of a single training example reaches. Applicable to all kernel functions except Linear	1. Choosing a small value induces constraints and causes the model to not capture the complexity of the data and setting a value too large can lead to overfitting. Hence, it works as a regularization parameter

Table 18: Support vector regression's tuned hyperparameters; Similar structure as before

Naive predictor

The naive predictor that is used as a comparison to the machine learning algorithms can be described as follows. It simply uses the previous observations as predictions for the next, e.g. it uses the values of $\Delta \log \text{ENV}_{2019}$ as predictions for $\Delta \log \text{ENV}_{2020}$ for all companies. Since test performances are compared, it simply implies that $y_{t+1,\text{test}} = \Delta \log \text{ENV}_{t,test}$.

Inter/extrapolated predictor

The other base model is a predictor that uses linear inter- and extrapolation for making predictions. In particular, it uses the values of the pairs (y_{t+1}, $\Delta \log \text{ENV}_t$) in the full training set (training + validation) to assign y-values to $\Delta \log \text{ENV}_t$ in the test set. This procedure is conducted using the ”scipy.interpolate.interp1d” API of scipy package.

5 Results

5.1 Firms' emission behaviour

Table 19 shows that a significant regression equation is found *(F(14,14723)=69,210, p =.00)*, with an adjusted R-squared of .985. Furthermore, it shows that significant regression coefficients are found for all variables except some dummy variables.

Dep. variable = $\log \text{CO2}_t$ *Df residuals* = 14,723 *Df model* = 14						*Adj. R-squared* = .985 *F-statistic* = 69,210 *Prob (F-statistic)* = .00
	β_i	p-value	.025	.975	$.025_b$	$.975_b$
Constant	.007	.217	-.004	.018	-.003	.017
$\log \text{CO2}_{t-1}$	.947	.000	.943	.952	.940	.954
ENV_{t-1}	-.008	.000	-.010	-.006	-.010	-.006
$\log \text{REV}_t$	.046	.000	.041	.050	.039	.053
Trend_t	-.001	.000	-.002	-.001	-.002	-.001
Basic Materials	.042	.000	.030	.053	.030	.053
Consumer Discretionary	-.002	.664	-.013	.008	-.011	.007
Consumer Staples	.008	.180	-.004	.020	-.002	.018
Energy	.035	.000	.022	.050	.022	.050
Financials	-.037	.000	-.048	-.026	-.050	-.030
Health Care	-.008	.215	-.020	.005	-.020	.002
Industrials	.006	.238	-.004	.017	-.003	.015
Real Estate	.010	.150	-.003	.023	-.003	.022
Technology	.000	.958	-.012	.012	-.009	.010
Utilities	.058	.000	.044	.072	.041	.077

Table 19: Regression results

Figure 2 shows a plot of the residuals against the fitted values. It shows that the residuals are distributed around zero. Furthermore, it indicates that a linear model is appropriate as the errors seem to be symmetrical around a horizontal line through zero (Date, 2022). It also shows signs of heteroskedasticity as the residuals are less dispersed at the right-hand side of the plot in comparison to the left-hand side. Hence, one of the assumptions of OLS is violated.

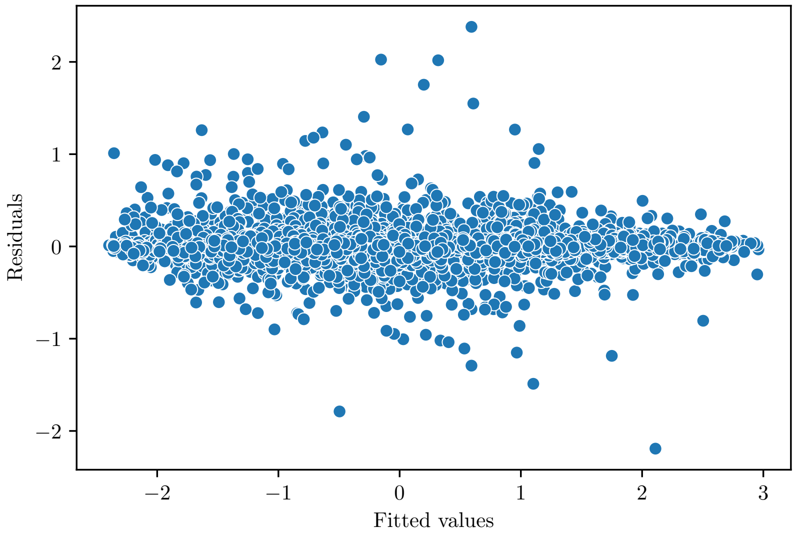

Figure 2: Regression residuals

Furthermore, Figure 3 shows a Quantile-Quantile (Q-Q) plot where the red line represents the expected data distribution if normally distributed. The blue observations represent the distribution of the residuals, indicating that they are not normally distributed, violating another assumption of OLS. This is also confirmed by a Jarque-Bera statistic of 1,863,757 which is highly significant.

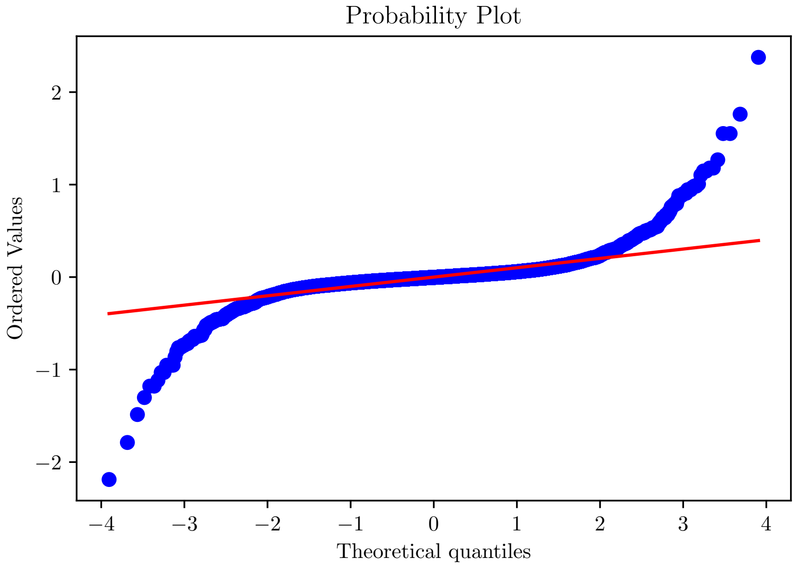

Figure 3: Quantile-Quantile plot

To circumvent these two assumptions of OLS, the confidence intervals of the regression coefficients are bootstrapped as described in the Method section. These values are shown in

Table 19 as $.025_b$ and $.975_b$, representing the lower and upper bounds of the 95% confidence intervals. Looking at the tables, one can see that the obtained bootstrapped confidence intervals are very similar to the confidence intervals obtained from OLS. Hence, it is chosen to continue with interpreting the regression coefficients derived from performing OLS.

Figure 4 shows the autocorrelation of the residuals at various time lags. The x-axis represents the lags and the y-axis show the autocorrelation. Every line within the blue rectangle implies that the found statistic is not significant. Hence, only at the 22nd time lag there is a significant result. However, Brockwell and Davis (2002) mention that autocorrelation functions that are close to zero for all non-zero lags indicate that errors are independently distributed, which is the case. The absence of autocorrelation is also supported by a Durbin-Watson statistic of 2.018. This test examines if there is autocorrelation in the first lag. A test statistic towards zero implies positive autocorrelation whereas a test statistic towards 4 implies negative autocorrelation. However, a test statistic around 2 implies no autocorrelation in this lag (Durbin & Watson, 1950). Therefore, it seems that the residuals are independently distributed.

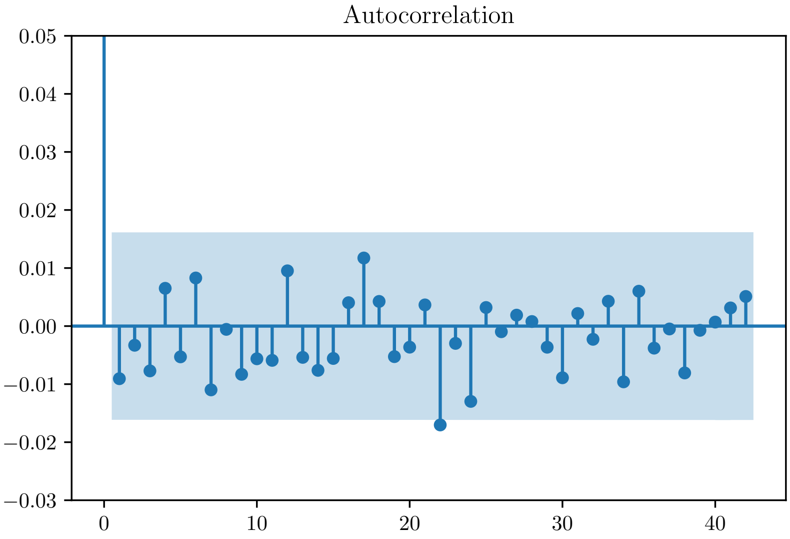

Figure 4: Autocorrelation of the residuals at various time lags

Besides non-normality and heteroskedasticity, the model fulfils the assumptions of linearity and no autocorrelation in the residuals. However, an adjusted R-squared of .985 is suspiciously high and raises the question of whether the model is potentially misspecified due to e.g. non-stationarity. Since CO2 emissions, E ratings and total revenue are all likely to exhibit a trend it is analysed whether taking differencing or logarithmic differencing leads to sensible results. However, this led to a drop in the adjusted R-squared to .06. Next to that, Wooldridge (2015) mentions that including a trend variable is sufficient to capture trends within individual variables which is already incorporated in the model. So we conclude that this high adjusted R-squared is likely not caused by wrongly accounting for trends in the data.

The fact that this analysis uses panel data is only due to the motivation of having more observations. However, this regression could also be conducted using cross-sectional data in 2020 as this already contain 1,678 observations. The same regression is run using this cross-sectional data set to assess whether the high adjusted R-squared is caused by other forms of non-stationarity. This results in similar regression results and an adjusted R-squared of .983. Since there are no non-stationarity issues in a cross-section and since the adjusted R-squareds are similar, it is further concluded that the high adjusted R-squared is not generated by non-stationarity. In addition, the regression outcomes are intuitively understandable and similar to the results of Chatterji et al. (2009). Considering all these observations, it is concluded that the model as shown in Table 19 is correctly specified and thus, we proceed with the interpretation of regression results.

It is good to note that all variables except the dummies are standardized. This means that for a simplified version of the model it be expressed as follows:

$$\frac{\log \mathrm{CO2}_t - \mu(\log \mathrm{CO2}_t)}{\sigma(\log \mathrm{CO2}_t)} = \beta_0 + \frac{\mathrm{ENV}_{t-1} - \mu(\mathrm{ENV}_{t-1})}{\sigma(\mathrm{ENV}_{t-1})}\beta_1 + \epsilon \tag{3}$$

Where μ and σ refer to the sample mean and standard deviation respecitvely. In order to interpret the results in original scales, equation 3 can be written as:

$$\begin{aligned}\log \mathrm{CO2}_t = \mu(\log \mathrm{CO2}_t) + \sigma(\log \mathrm{CO2}_t)\beta_0 - \frac{\sigma(\log \mathrm{CO2}_t)\mu(\mathrm{ENV}_{t-1})\ \beta_1}{\sigma(\mathrm{ENV}_{t-1})} \\ + \frac{\sigma(\log \mathrm{CO2}_t)\beta_1}{\sigma(\mathrm{ENV}_{t-1})}\mathrm{ENV}_{t-1} + \sigma(\log \mathrm{CO2}_t)\epsilon\end{aligned} \tag{4}$$

So in order to compare the effect of a unit increase of the independent variables on the dependent variable, the regression coefficients have to be rescaled as shown in equation 4. In particular by:

$$\frac{\sigma(\log \mathrm{CO2}_t)\beta_j}{\sigma(\mathrm{X}_{j,(t/t-1)})} \tag{5}$$

Where $X_{j,(t/t-1)}$ refers to independent variable j in time t or t-1 depending on the independent variable and β_j to its regression coefficient. The regression coefficients of the trend variable and the dummy variables remain the same as these are non-standardized. Computing equation 5 for all the standardized independent variables results in the following betas:

$X_{j,(t/t-1)}$	σ	$\beta_{j,scaled}$	$\beta_{j,rescaled}$
$\log \text{CO2}_{t-1}$	3.1	.947	.951
ENV_{t-1}	20.8	-.008	-.001
$\log \text{REV}_t$	2.5	.046	.057

Table 20: Rescaled regression coefficients

Note that $\log CO2_t$ to ENV_{t-1} is log-linear and $\log CO2_t$ to $\log CO2_{t-1}$ and $\log REV_t$ is log-log. Hence, a unit increase in the lagged E rating (on a 1-100 scale) results in a .122% decrease in total CO2 and equivalent emissions ($(\exp^{-.001} - 1) * 100 = -.122\%$). This implies that firms that obtain a higher E rating tend to reduce their CO2 emissions in the following year.

For $\log CO2_{t-1}$, this implies that a one percent increase in CO2 emissions in the current year leads to a .951% increase in CO2 emissions in the next. This means that bad performers are likely to continue to perform badly. For the size variable, i.e. firms' revenue, this implies that a 1% increase results in a .057% increase in their CO2 emissions. This positive relation is intuitively comprehensible as growth in size implies e.g. producing more products which leads to higher CO2 emission levels.

For the trend variables, the regression coefficient is equal to -.001. This implies that on average, firms' CO2 emissions decrease every year by -.122%. This decrease is in line with the global growth in climate actions and regulations over the past years.

Looking at the dummy variables one can see that there are significant results obtained for the categories Basic Materials, Energy, Financials and Utilities with regression coefficients .042, .035, -.037 and .058 respectively. Their coefficients are not rescaled as these variables are only used as control variables. However, it is interesting to briefly discuss their sign. All these variables except Financials pose a positive sign. This implies the Financial industry emits fewer CO2 emissions than the Telecom industry whereas the Basic Materials, Energy and Utilities industries emit more. This is understandable as the last three mentioned industries are more production heavy industries in comparison to the others.

The significant negative relation between E ratings and CO2 emissions confirms that firms reduce their CO2 emissions to obtain a higher E rating. Moreover, this proves that E ratings are a driving factor behind firms' CO2 emissions. This motivates examining the predictability of E ratings as the above provides a reason to incorporate E ratings in CO2 models. This is done in the following section.

5.2 Prediction

5.2.1 Feature selection

Performing feature selection with RFECV and using a non-tuned XGBoost estimator results in selecting the following features:

ΔENV$_t$	ROI$_t$	GP$_t$	REV$_t$	TREND$_t$	~~DEBT$_t$~~	~~ROE$_t$~~	~~ROA$_t$~~	~~EV$_t$~~	~~PE$_t$~~	~~PtB$_t$~~	~~EpS$_t$~~

Table 21: Selected features using RFECV and XGBoost

Using the same algorithm but with random forest and SVR as estimator results in selecting the full set of features. It is also tested whether using an untrained random forest estimator as feature selection method leads to better results but this is not the case. The same is done for the feature importance coefficients of XGBoost but again, this does not lead to better results. Hence, it is chosen to proceed with fitting the algorithms using the full set of features and compare this with the selected features as described in Table 21.

5.2.2 Model selection

This chapter shows the best performing hyperparameters using randomized search and bayesian optimization with and without feature selection, i.e. randomized search with full feature set (RF), with a subset using feature selection (RS), bayesian optimization with full feature set (BF) and with a subset (BS). Next to that, the differences between test and training error are compared across other models, i.e. RF, BF and BS, in order to detect overfitting. Finally, computational time consisting of training and predicting time for each cross-validated hyperparameter combination is presented to compare the models and algorithms on computational speed. These acronyms and methods of presenting are used for all algorithms.

Base models

As said earlier the algorithms are compared with a naive predictor and a linear inter/extrapolated predictor. The former obtains a test NRSME and NMAE of 8.377% and 4.400% respectively. The latter achieves a performance of 7.281% and 4.307% respectively.

Random Forest

	RF	RS	BF	BS
N_estimators	1,000	800	848	618
Max_features	auto	.33	.99	.94
Max_depth	25	30	30	14
Min_samples_split	2	10	21	20
Min_samples_leaf	5	10	9	5
Max_samples	.5	.5	.49	.33
NMAE$_{test}$	3.170	3.287	3.128	3.223
NRMSE$_{test}$	5.977	5.937	5.959	5.930
NRMSE$_{train}$	3.899	3.747	4.220	4.445
Computation time	158.1	23.9	84.1	21.5

Table 22: Hyperparameter tuning for random forest with Randomized and Bayesian search using Full feature set and a Subset respectively (i.e. RF, RS, BF and BS)

Looking at Table 22, one can see that the bayesian optimisation with feature selection resulted in the best performance with a NRMSE$_{test}$ of 5.930% and NMAE$_{test}$ of 3.223%. This means that this model predicts the E ratings with 94.1% accuracy in terms of NRSME and with 96.8% in terms of NMAE. In comparison with the naive and inter/extrapolated models, this is a 2.447% and 1.351% NRMSE$_{test}$ improvement respectively. NRMSE$_{train}$ refers to the performance on the full training set (training + validation) and one can see that it is 1.485% lower than the one of the test set. Furthermore, the gap between training-test (”spread”) performance is lowest in comparison with the other models. This implies that the hyperparameters are relatively very well optimised.

Table 22 further shows that the best predictive performance is obtained using hyperparameter sets that reduce overfitting the most. This can be seen by comparing the hyperparameter values of the BS and RS models with those of the others. The BS model uses the lowest number of estimators, the smallest depth, the second-highest minimum samples split and the lowest fraction of samples that are used when building the trees. A similar explanation can be given for the RS model. Using 3-fold cross-validation for hyperparameter tuning implies that the best hyperparameter combination is selected based on its predictive performance on three subsets of the training set. This together with the fact that the best models are obtained using strict overfit-avoiding (OA) settings, implies that there is quite some noise in the training set.

Furthermore, one can see that RFECV does a good job in selecting the best features. In particular, this is interesting as it only uses the ΔENV$_t$, ROI$_t$ GP$_t$, REV$_t$ and TREND$_t$ next to the dummy variables.

Next to that, it is striking to see that selecting the best model based on NRMSE or on NMAE results in different models. The former results in selecting BF whereas the latter in selecting BS. This means that in absolute terms the BF model performs better but that it is not selected as it contains more extreme errors.

XGBoost

	RF	RS	BF	BS
N_estimators	400	400	152	152
Max_depth	3	3	4	4
Learning_rate	.015	.015	.033	.033
Colsample_bytree	.8	.8	.8	.8
Subsample	1.0	1.0	.83	.83
Gamma	1.0	1.0	.94	.94
Min_child_weight	7	7	17	17
Reg_alpha	10	10	4.6	4.6
Reg_lambda	.1	.1	53.5	53.5
NMAE$_{test}$	3.165	3.192	3.148	3.198
NRMSE$_{test}$	5.989	5.978	5.943	5.933
NRMSE$_{train}$	4.648	4.667	4.678	4.668
Computation time	16.9	13.2	6.5	3.9

Table 23: Hyperparameter tuning for XGBoost with Randomized and Bayesian search using Full feature set and a Subset respectively (i.e. RF, RS, BF and BS)

Table 23 shows that for XGBoost the best model is found using bayesian optimisation and feature selection with a test and training NRMSE of 5.933% and 4.668% respectively. This result is very similar to the best model using random forest, i.e. 5.930%. Hence, this model performs 2.444% and 1.348% better than the base models.

The difference between the test-training NRMSEs of this model is 1.265%, which is together with the BF model the lowest. Again, the best performances are obtained using relatively the most strict OA settings, i.e. a lower number of estimators, fraction of subsample and gamma together with a higher minimum child weight and more regularization by lambda. This supports the earlier made suggestion that data is noisy.

Next to that, one can see that the RF spread is lower than the best model of random forest, i.e. 1.485%. This means that the hyperparameters are fairly well-tuned in comparison to the random forest models. However, most of the random forest models perform better even though they are less well-tuned. Hence, random forest is likely more suited for this data set.

Neural Network

	RF	RS	BF	BS
N_hidden	1	3	2	2
N_neurons	5	23	2	28
Learning_rate	.1	.2	.000	.020
Optimizer	adam	adam	adam	adam
Activation	relu	elu	relu	elu
Batch_size	64	64	947	960
Epochs	50	50	337	303
Dropout	.1	.1	.2	.1
Kernel_initializer	he_unif	he_norm	he_norm	he_norm
NMAE_{test}	3.102	3.097	3.183	3.079
NRMSE_{test}	5.989	6.029	6.007	6.047
NRMSE_{train}	4.791	4.689	4.820	4.683
Computation time	59.5	80.4	44.6	59.1

Table 24: Hyperparameter tuning for neural network with Randomized and Bayesian search using Full feature set and a Subset respectively (i.e. RF, RS, BF and BS)

From Table 24, one can see that the best model of neural network is found using randomized search and using the full feature set. In particular, this model achieves a test NRMSE of 5.989% which is slightly worse than the best models of random forest and XGBoost, i.e. .059% and .056% respectively. The training-test difference is 1.198% which is smaller than both the best random forest and XGBoost models. Hence, the hyperparameters are relatively well-tuned.

The best performance is obtained using one hidden layer. This corresponds with the statement of Goodfellow et al. (2016) that a neural network with one hidden layer is able to approximate any function and therefore sufficient in most cases. Next to that, it is interesting to see that similar results are obtainable using very different values for the learning rate, i.e. RF and BF models differ .018% in NRMSE_{test} while using a learning rate of .1 and .0001 respectively. Perhaps, this indicates that there is a deep minimum in the loss function which is easy to find and which is hard to overshoot.

Support Vector Regression

	RF	RS	BF	BS
Kernel	rbf	rbf	rbf	rbf
Gamma	scale	scale	auto	auto
C	100	100	1.22	1.16
Epsilon	.001	.001	.001	.001
NMAE$_{test}$	3.121	3.131	3.212	3.324
NRMSE$_{test}$	6.197	6.151	6.071	6.078
NRMSE$_{train}$	4.710	4.783	4.725	4.766
Computation time	2.8	1.6	38.4	25

Table 25: Hyperparameter tuning for support vector regression with Randomized and Bayesian search using Full feature set and a Subset respectively (i.e. RF, RS, BF and BS)

Table 25 shows that the best performance is achieved using bayesian optimisation and using the full feature set. This leads to a test NRMSE of 6.071% which is .144% worse than the overall best performing model. Together with the BS model, the spread is the lowest in comparison to the RF and RS models. The best performing models, i.e. BF and BS, differ mainly from the other models in regard to C. A lower value for C reduces overfitting, so again best models are found using strict OA settings.

Also interesting to see is that epsilon is set to the smallest tested value. This implies that the error margin is very low and that fewer observations fall within the ϵ-insensitive tube. This also means that more errors outside this tube are penalised for fitting the regression function. This increases the chance of overfitting. However, this is accounted for by a low value for C.

5.2.3 Best model

	NRMSE$_{test}$	NMAE$_{test}$	Spread	Method	Computational time
Random forest	5.930	3.223	1.485	BS	21.5
XGBoost	5.933	3.198	1.265	BS	3.9
Neural network	5.989	3.102	1.198	RF	59.5
SVR	6.071	3.212	1.346	BF	38.4

Table 26: Descriptives of the best models

Looking at Table 26, one can see that random forest obtained the best results closely followed by XGBoost. Furthermore, the spread of random forest's best model is the highest among all algorithms. This indicates that the random forest model could be optimised even further. Hence, it seems that the random forest algorithm is most suitable for this data set.

The reason that the tree-based models, i.e. random forest and XGBoost outperform neural network might be due to the small data size. The training data consists of 14,216 rows and all the time series do not contain more than 19 observations. Hence, it might be more difficult for the neural networks to assign the right weights to the neurons whereas for tree-based models the weights are binary, i.e. 0 or 1 (Ye, 2020). In addition, the problem at hand is relatively not that complex with few layers, neurons and features which is why tree-based models are likely to perform as good or even better than the neural network (Ye, 2020).

In this case, random forest outperforms XGBoost and this is probably due to the difference in nature of the algorithms i.e. the former uses bagging whereas the latter uses boosting. Bagging aims to decrease variance and reduce overfitting whereas boosting aims to reduce bias. Hence, this implies that there is quite some noise in the training data and that the test set is quite different from the training set. The latter makes sense as the test-train split is made based on the years, i.e. the last two years, and within these years firms can change much.

The noisy data also explains why the SVR algorithm has more difficulty fitting the right hyperplane to the data and why this model performs less than the other models.

5.2.4 Interpretation

This chapter interprets the best performing model using the full feature set and assesses whether similar relations can be found as the ones shown in the existing literature. This is done by applying the SHAP algorithm as explained in chapter ”SHAP values” (4.2.7). Figure 5 shows the resulting relative importance of each feature, which is equal to the mean of all SHAP values of all observations. Figure 6 (a beeswarm boxplot), however, shows the individual SHAP values of all observations. A blue value represents small values of the respective feature whereas the red value represents large values. The x-axis shows their impact on the predictions.

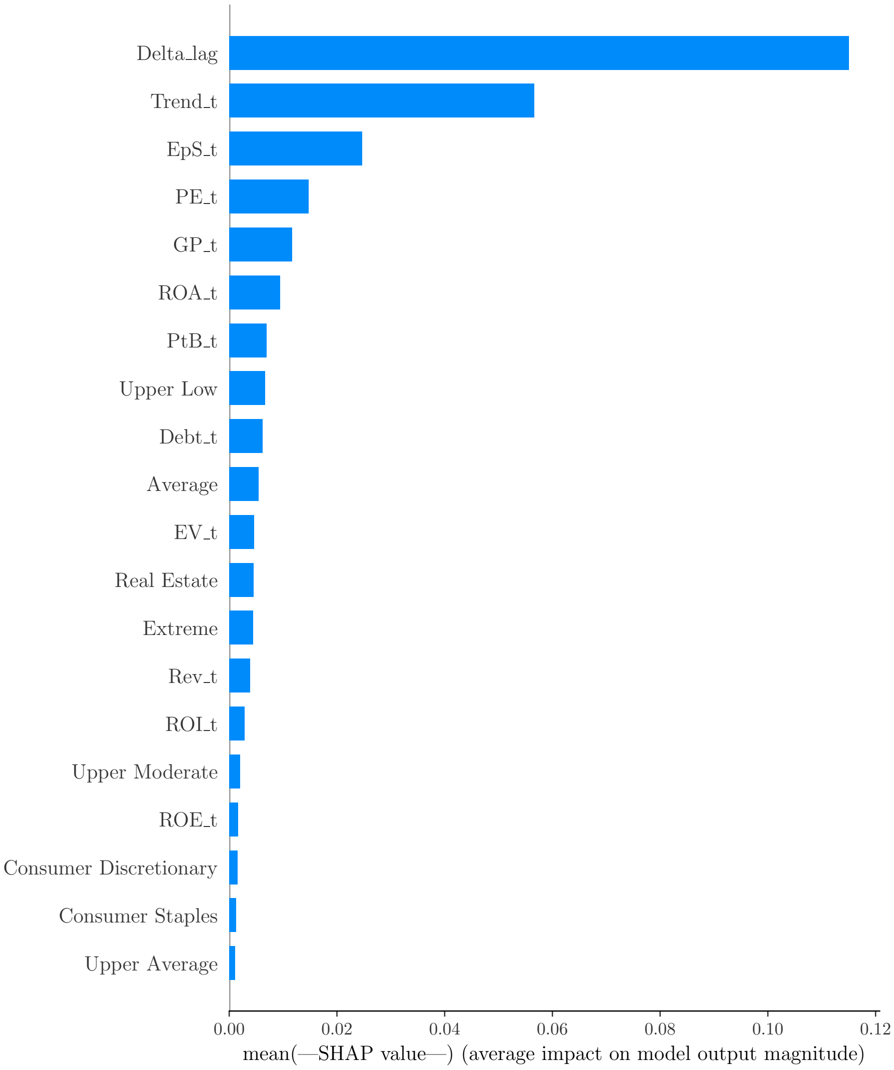

Figure 5: SHAP feature importance

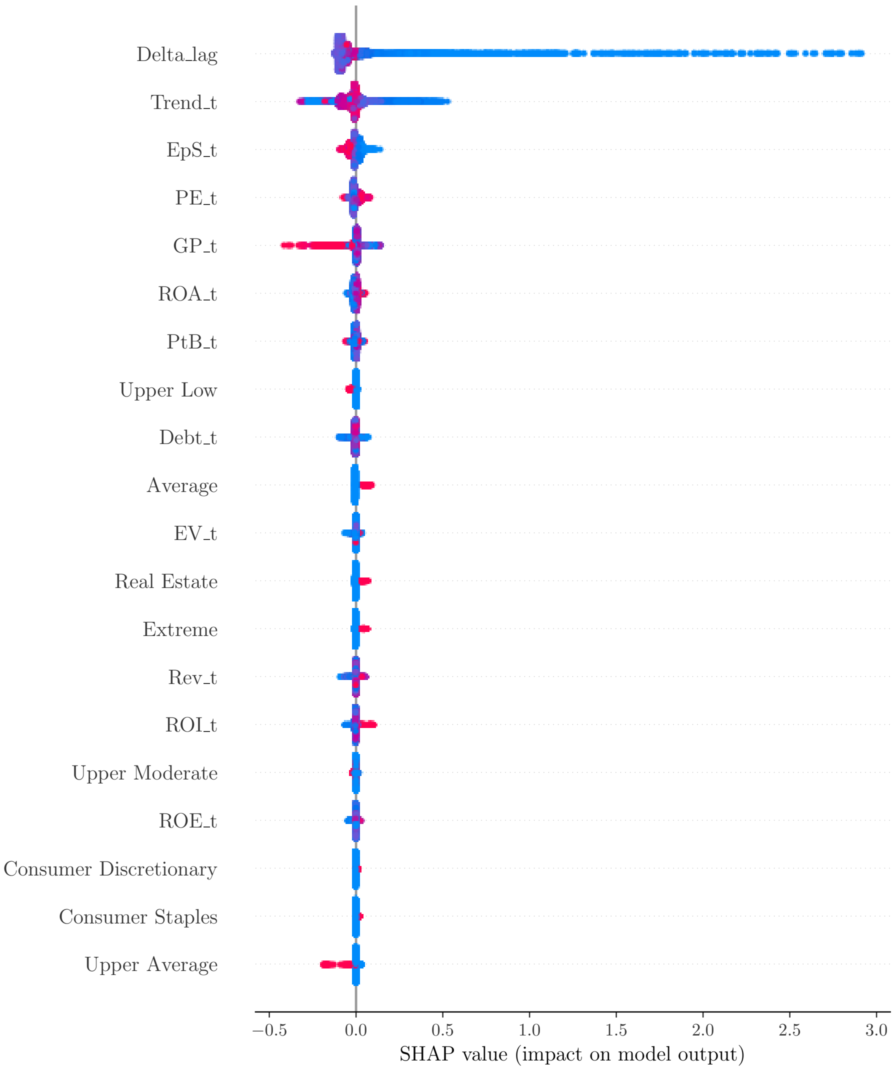

Figure 6: SHAP plot; blue and red indicate low and high feature values respectively

It is good to note that the best overall performing model is obtained using feature selection. In particular, the RFECV algorithm removed more than half of the features in question including DEBT$_t$ and EV$_t$. This is unfortunate in regard to interpretation as it does not allow for analysing the relations between debt and enterprise value with E ratings as discussed in the literature section. However, the best overall performing model that uses the full feature set (XGBoost's BF model) obtains a NRMSE$_{test}$ of 5.943% which is only .013% higher than the best performing model. Also, their NRMEs are not statistically different when comparing their confidence intervals. Hence, it is chosen to report the SHAP values of XGBoost's BF model.

Looking at Figure 6, one can see that it is unclear to interpret the sign of the relations of total debt and enterprise value with E ratings. However, Figure 5 shows that these variables do have a decent feature importance value. Next to that, the fourth most important feature is price-to-earnings ratio which represents a firm's market value relative to its earnings. Hence, it is very related to firm value. Figure 6 shows a clear positive relation between PE ratio and E ratings. This implies that firms with a higher PE ratio, i.e. higher valuation perceived by the market, have a higher E rating. This is in line with the findings of Fatemi et al. (2018), which find that a higher E rating is associated with a higher firm value.

Another interesting observation is that the lagged dependent variable has negative relation with the differentiated E ratings implying that firms that achieved a large sustainable improvement in one year are likely to have a smaller change in the next year. This is in line with the findings of Chatterji and Toffel (2010) that show that firms that are initially rated poorly are likely to improve their environmental performance relatively more than firms that obtained good ratings in the beginning.

Finally, Figure 5 shows that the firm characteristics EpS$_t$, PE$_t$ and GP$_t$ are in the top five of most important features. This implies that these characteristics indeed have predictive performance as suggested by the research of Garcia et al. (2020).

6 Discussion

6.1 Conclusion

This study analyses environmental ratings' ability to model firms' CO2 emissions. This is done by splitting this research into two parts. The first studies the relation between Environmental ratings and firms' CO2 emissions, and the second examines the predictability of these ratings. The idea of analysing the former relation is cultivated by the findings of previous literature that high ESG ratings lead to capital inflow, lower cost of debt and higher firm value. Hence, this incentives firms to improve their environmental performance which includes reducing CO2 emissions. If it appears that firms' CO2 emission behaviour is indeed driven by its E rating then it implies that E ratings should be incorporated in models that forecast this behaviour as it then simply has more explanatory power. However, if the predictability of E ratings appears to be poor then incorporating E ratings in modelling future CO2 levels induces extra uncertainty which loses the point of including them in the first place. Hence, this motivates conducting the second part of this thesis. This further implies that the results of the first part of this thesis stand or fall with the results of the second part and vice versa.

Regarding the relation between E ratings and firms' CO2 emissions, this thesis shows that the former is indeed a driving factor behind the latter. In particular, the regression results show that a unit increase in E ratings leads to a .122% decrease in firms' CO2 emissions in the next year. Regarding E ratings' predictability, this thesis shows that machine learning algorithms can effectively predict E ratings with a low uncertainty. In particular, it shows that random forest and XGBoost tuned with bayesian optimisation and using feature selection predict E ratings with 94.1% accuracy in terms of NRSME and more significantly, with 96.8% in terms of NMAE. Moreover, this is a NRMSE improvement of 1.4% and 2.5% in comparison to an inter/extrapolated predictor and a naive predictor respectively.

Combining the outcomes of both the analyses imply that E ratings can be used effectively to model firms' future emission behaviour. This further means that incorporating E ratings in these models is likely to reduce prediction error. This is beneficial as it helps countries to better forecast the emission behaviour of firms and adjust regulations accordingly. This thesis further argues that this lower prediction uncertainty could be translated into lower prediction uncertainty in other CO2 models when including or incorporating firms' emission behaviour. Hence, this study suggests that E ratings should be incorporated in all models that forecast firms' emission behaviour or include/integrate this behaviour.

6.2 Limitations and future research

This thesis is, to our knowledge, the first that analyses the ability of E ratings to model firms' emission behaviour. Naturally, this comes with limitations which point the way for future research.

This thesis limits itself to ratings provided by one rater. This is caused by the fact that Erasmus students have access to the binary ratings of MSCI and the continuous ones of Refinitiv, and because it is chosen to only analyse continuous ratings. This means that these results of the regression analysis as well as the prediction part only hold for the data provided by this rater. So for future research, it could be assessed whether similar results can be found using a different rater and thus if this thesis' findings apply to multiple raters.

Using the ratings of one rater brings about another limitation which is that the effect of different ratings on firms' emission behaviour is not analysed. There might be divergence between ratings of various raters which raises the question of how this relates to firms' emission behaviour. It would be interesting to analyse if firms use one particular rater or a combination of raters as target for reducing their CO2 emissions. This would have modelling implications as it could imply that a combination of ratings should be analysed as a predictor.

Next to that, this thesis advises to include E ratings in modelling firms' emission behaviour but does not analyse its exact implementation. This implies that exact figures regarding the possible reduction of prediction uncertainty when including E ratings cannot be given. Hence for further research, it is interesting to measure this effect.

References

Allen, M., Babiker, M., Chen, Y., Coninck, H. d., Connors, S., Diemen, et al. (2018). Global warming of 1.5° c. summary for policymakers.

Antoncic, M., Bekaert, G., Rothenberg, R. V., & Noguer, M. (2020). Sustainable investment-exploring the linkage between alpha, esg, and sdg's. *ESG, and SDG's (August 2020).*

Baltagi, B. H. (2008). *Econometric analysis of panel data* (Vol. 4). Springer.

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade* (pp. 437–478).

Berg, F., Koelbel, J. F., & Rigobon, R. (2019). *Aggregate confusion: The divergence of esg ratings*. MIT Sloan School of Management.

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, *13*(2).

Boehmke, B., & Greenwell, B. (2019). *Hands-on machine learning with r*. Chapman and Hall/CRC.

Boffo, R., & Patalano, R. (2020). Esg investing: Practices, progress and challenges. Retrieved from `https://www.oecd.org/finance/ESG-Investing-Practices-Progress-Challenges.pdf`

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

Brockwell, P. J., & Davis, R. A. (2002). *Introduction to time series and forecasting*. Springer.

Brownlee, J. (2019). Xgboost with python. *Machine Learning Mastery.*

Chatterji, A. K., Durand, R., Levine, D. I., & Touboul, S. (2016). Do ratings of firms converge? implications for managers, investors and strategy researchers. *Strategic Management Journal*, *37*(8), 1597–1614.

Chatterji, A. K., Levine, D. I., & Toffel, M. W. (2009). How well do social ratings actually measure corporate social responsibility? *Journal of Economics & Management Strategy*, *18*(1), 125–169.

Chatterji, A. K., & Toffel, M. W. (2010). How firms respond to being rated. *Strategic Management Journal*, *31*(9), 917–945.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785–794).

Cherkassky, V., & Ma, Y. (2004). Practical selection of svm parameters and noise estimation for svm regression. *Neural networks*, *17*(1), 113–126.

Ciais, P., Sabine, C., Bala, G., Bopp, L., Brovkin, V., Canadell, J., ... others (2014). Carbon and other biogeochemical cycles. In *Climate change 2013: the physical science basis. contribution of working group i to the fifth assessment report of the intergovernmental panel on climate change* (pp. 465–570). Cambridge University Press.

Co2 emissions. (2022). Retrieved from `https://data.worldbank.org/indicator/EN.ATM.CO2E.PP.GD`

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, *20*(3), 273–297.

Cribari-Neto, F., & Zarkos, S. G. (1999). Bootstrap methods for heteroskedastic regression models: evidence on estimation and testing. *Econometric Reviews*, *18*(2), 211–228.

Date, S. (2022). *How to build a pooled ols regression model for panel data sets.* Retrieved from `https://medium.com/towards-data-science/how-to-build-a-pooled-ols-regression-model-for-panel-data-sets-a78358f9c2a`

De Franco, C., Geissler, C., Margot, V., & Monnier, B. (2020). Esg investments: Filtering versus machine learning approaches. *arXiv preprint arXiv:2002.07477*.

Dozat, T. (2016). Incorporating nesterov momentum into adam.

Durbin, J., & Watson, G. S. (1950). Testing for serial correlation in least squares regression: I. *Biometrika*, *37*(3/4), 409–428.

Environmental, social and governance scores from refinitiv. (2021). Retrieved from `https://www.refinitiv.com/content/dam/marketing/en_us/documents/methodology/refinitiv-esg-scores-methodology.pdf`

Erragragui, E. (2018). Do creditors price firms' environmental, social and governance risks? *Research in International Business and Finance*, *45*, 197–207.

Evans, S., & Hausfather, Z. (2018). *How 'integrated assessment models' are used to study climate change.* Retrieved from `https://www.carbonbrief.org/qa-how-integrated-assessment-models-are-used-to-study-climate-change`

Fatemi, A., Glaum, M., & Kaiser, S. (2018). Esg performance and firm value: The moderating role of disclosure. *Global Finance Journal*, *38*, 45–64.

Fedorová, D., et al. (2016). Selection of unit root test on the basis of length of the time series and value of ar (1) parameter. *Statistika*, *96*(3), 3.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189–1232.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational statistics & data analysis*, *38*(4), 367–378.

Garcia, F., González-Bueno, J., Guijarro, F., & Oliver, J. (2020). Forecasting the environmental, social, and governance rating of firms by using corporate financial performance variables: A rough set approach. *Sustainability*, *12*(8), 3324.

Géron, A. (2019). *Hands-on machine learning with scikit-learn, keras, and tensorflow: Concepts, tools, and techniques to build intelligent systems.* " O'Reilly Media, Inc.".

Global Sustainable Investment Alliance. (2020). Global sustainable investment review. Retrieved from `http://www.gsi-alliance.org/wp-content/uploads/2021/08/GSIR-20201.pdf`

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the ieee international conference on computer vision* (pp. 1026–1034).

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krappel, T., Bogun, A., & Borth, D. (2021). Heterogeneous ensemble for esg ratings prediction. *arXiv preprint arXiv:2109.10085*.

Larochelle, H., Bengio, Y., Louradour, J., & Lamblin, P. (2009). Exploring strategies for training deep neural networks. *Journal of machine learning research*, *10*(1).

Lundberg, S. M., Erion, G. G., & Lee, S.-I. (2018). Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv:1802.03888*.

Mitsuzuka, K., Ling, F., & Ohwada, H. (2017). Analysis of csr activities affecting corporate value using machine learning. In *Proceedings of the 9th international conference on machine learning and computing* (pp. 11–14).

Myhre, G., Bréon, F.-M., & Granier, C. (2018). Anthropogenic and natural radiative forcing 2. *Notes*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Probst, P., Wright, M. N., & Boulesteix, A.-L. (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery*, *9*(3), e1301.

Pukelsheim, F. (1994). The three sigma rule. *The American Statistician*, *48*(2), 88–91.

Shapley, L. (1953). Quota solutions op n-person games1. *Edited by Emil Artin and Marston Morse*, 343.

Sharma, S., Sharma, S., & Athaiya, A. (2017). Activation functions in neural networks. *towards data science*, *6*(12), 310–316.

Signatory update. (2021). Retrieved from `https://www.unpri.org/download?ac=14962`

Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, *14*(3), 199–222.

Svanberg, J., Ardeshiri, T., Samsten, I., Öhman, P., Rana, T., & Danielson, M. (2022). Prediction of environmental controversies and development of a corporate environmental performance rating methodology. *Journal of Cleaner Production*, *344*, 130979.

Vinutha, H., Poornima, B., & Sagar, B. (2018). Detection of outliers using interquartile range technique from intrusion dataset. In *Information and decision sciences* (pp. 511–518). Springer.

Wong, C., & Petroy, E. (2020). Rate the raters 2020. Retrieved from `https://www.sustainability.com/thinking/rate-the-raters-2020/`

Wooldridge, J. M. (2015). *Introductory econometrics: A modern approach.* Cengage learning.

Ye, A. (2020). When and why tree-based models (often) outperform neural networks. Retrieved from `https://towardsdatascience.com/when-and-why-tree-based-models-often-outperform-neural-networks-ceba9ecd0fd8`

A Appendix

A.1 Tables

Environmental	Innovation	Resource use
Policy Emissions	Environmental Products	Resource Reduction Policy
Targets Emissions	Eco-Design Products	Policy Water Efficiency
Emission Reduction Target Percentage	Revenue from Environmental Products	Policy Energy Efficiency
Emission Reduction Target Year	Percentage of green products	Policy Sustainable Packaging
Biodiversity Impact Reduction	Total Env R&D / Million in Revenue	Policy Environmental Supply Chain
Estimated CO2 Equivalents Emission Total	Environmental R&D Expenditures	Resource Reduction Targets
CO2 estimation method	Noise Reduction	Targets Water Efficiency
Total CO2 Emissions / Million in Revenue $	Fleet Fuel Consumption	Targets Energy Efficiency
CO2 Equivalent Emissions Total	Hybrid Vehicles	Environment Management Team
CO2 Equivalent Emissions Direct, Scope 1	Fleet CO2 Emissions	Environment Management Training
CO2 Equivalent Emissions Indirect, Scope 2	Environmental Assets Under Mgt	Environmental Materials Sourcing
CO2 Equivalent Emissions Indirect, Scope 3	ESG Assets Under Management	Toxic Chemicals Reduction
Carbon Offsets/Credits	Equator Principles	Total Energy Use / Million in Revenue $
Emissions Trading	Equator Principles or Env Project Financing	Energy Use Total
Cement CO2 Equivalents Emission	Environmental Project Financing	Energy Purchased Direct
Climate Change Commercial Risks Opportunities	Nuclear	Energy Produced Direct
Flaring Gases To Revenues USD in million	Nuclear Production	Indirect Energy Use
Flaring Gases	Labeled Wood Percentage	Electricity Purchased
Ozone-Depleting Substances	Labeled Wood	Electricity Produced
NOx and SOx Emissions Reduction	Organic Products Initiatives	Grid Loss Percentage
NOx Emissions To Revenues USD in million	Product Impact Minimization	Renewable Energy Use Ratio
NOx Emissions	Take-back and Recycling Initiatives	Renewable Energy Supply
SOx Emissions To Revenues USD in million	Products Recovered to Recycle	Total Renewable Energy
SOx Emissions	Product Environmental Responsible Use	Renewable Energy Purchased
VOC or Particulate Matter Emissions Reduction	GMO Products	Renewable Energy Produced
VOC Emissions Reduction	Agrochemical Products	Renewable Energy Use
Particulate Matter Emissions Reduction	Agrochemical 5 % Revenue	Cement Energy Use
VOC Emissions To Revenues USD in million	Animal Testing	Coal produced (Raw Material in Tonnes) Total
VOC Emissions	Animal Testing Cosmetics	Green Buildings
Total Waste / Million in Revenue $	Animal Testing Reduction	Total Water Use / Million in Revenue $
Waste Recycled To Total Waste	Renewable/Clean Energy Products	Water Withdrawal Total
Total Hazardous Waste / Million in Revenue $	Water Technologies	Fresh Water Withdrawal Total
Waste Total	Sustainable Building Products	Water Recycled
Non-Hazardous Waste	Real Estate Sustainability Certifications	Environmental Supply Chain Management
Waste Recycled Total	Fossil Fuel Divestment Policy	Environmental Supply Chain Monitoring
Waste Recycling Ratio		Env Supply Chain Partnership Termination
Hazardous Waste		Land Environmental Impact Reduction
Waste Reduction Initiatives		
e-Waste Reduction		
Total Water Pollutant Emissions / Million in Revenue $		
Water Discharged		
Water Pollutant Emissions		
ISO 14000 or EMS		
EMS Certified Percent		
Environmental Restoration Initiatives		
Staff Transportation Impact Reduction		
Accidental Spills To Revenues USD in million		
Accidental Spills		
Environmental Expenditures Investments		
Environmental Expenditures		
Environmental Provisions		
Environmental Investments Initiatives		
Self-Reported Environmental Fines		
Environmental Partnerships		
Internal Carbon Pricing		
Internal Carbon Price per Tonne		
Policy Nuclear Safety		

Table 27: Composition Environmental Pillar