CDS APPROXIMATION ACCURACY IMPROVEMENT WITH CART AND RANDOM FOREST ALGORITHMS BASED ON A TIME-SPAN INCLUDING THE COVID-19 PANDEMIC PERIOD

Mathieu Mercadier*

*ESC Clermont Business School, CleRMa-UCA, 4 Boulevard Trudaine, 63000 Clermont-Ferrand, France.

Abstract: This study uses decision tree and random forest regressions to improve the accuracy of an approximation of CDS spreads called the Equity-to-Credit (E2C) formula, based on a time-span including the COVID-19 pandemic period. Certain sections are dedicated to explaining deeper important concepts in machine learning. Random forest regressions run with the E2C and selected additional financial data results in an accuracy in CDS approximations of 82% out-of-sample. The transparency property of these algorithms confirms that, for CDS spreads' forecasting, the most used feature is the E2C formula and to a lower extent companies' debt rating and size.

1. INTRODUCTION

The accuracy of the assessment of companies' credit risk concerns financial professionals and an extensive branch of literature is dedicated to this matter (e.g., Merton, 1974; Black and Cox, 1976; Vasicek, 1987; Finger et al., 2002). Since the mid-nineties, a derivative, the Credit Default Swap (CDS), can be used as an insurance against the uncertainty regarding the capacity of a debtor to fulfill its contractual obligations. Nevertheless, not all companies have actively traded CDS, leading to the development of estimations. Merton (1974) was the first to notice a relationship between the probability of default and the capital structure of a company. In brief, he used the option pricing framework to relate the three segments of the balance sheet: asset, debt and equity. Other academics, Black and Cox (1976) integrated to the model the assumption that a default can occur prior to maturity. But, when CDS started to be traded, it is in the private sector that many approximations flourished. In the midst of all these highly proprietary models, the CreditGrades model (Finger et al., 2002) has been developed by practitioners and is available for use to everyone. This formula relies on the asset values of the companies modeled dynamically with a diffusion process and includes the concept of Black and Cox (1976) of a default barrier that can be crossed prior to maturity. Since then, academics have contributed to define further some input variables of the CreditGrades model to improve its accuracy (Zhou ,2001; Sepp, 2006; Stamicar and Finger, 2006; Escobar et al., 2012).

In the literature, CDS is a topic of interest (Guarin et al., 2011; Tomohiro, 2014; Cont and Minca, 2016; Chalamandaris and Vlachogiannakis, 2018; Koutmos, 2018; Irresberger et al., 2018), but in practice, models for traded CDS spreads tend not to be popular among portfolio managers. Although the cited models provide very close estimations (e.g., Imbierowicz and Cserna, 2008), they are perceived as too complex. It is in this context that Mercadier and Lardy (2019) (from now on this original paper is referred to as ML2019) develop a concise, transparent and broad-based approximation assessing CDS spreads. This Equity-to-Credit (E2C) formula is a pared-down elementary equation inspired by the CreditGrades model (Finger et al., 2002). After an empirical confrontation of the E2C formula to the actual CDS to evaluate its reliability, the authors propose to improve its accuracy using machine learning, which have become more prevalent in the financial sector.

In fact, they highlight that the E2C formula and structural models in general exclude some parameters influencing credit spreads. Thus, the accuracy of their model is improved running a supervised learning algorithm on a multivariate universe. The E2C formula and selected complementary features are set as independent variables, and the dependent variable is the 5y CDS spread. The random forest regression algorithm (Breiman, 2001) is selected for its intelligibility: broadly speaking, this methodology averages multiple randomly bootstrapped decision trees constructed with subsets of the features randomly chosen at each node. Ultimately, the goal is to obtain a high out-of-sample accuracy. An additional interesting property of this algorithm is linked to its transparency. In fact, one can observe how the decision is made at each node, it is thus, straightforward to display the feature chosen to reduce the error. This allows to quantify the contribution of each feature in predicting the CDS spreads and confirms that the decision trees mainly select the E2C formula. Additionally, it identifies the next best source of improvements as the credit rating and the size of the companies.

In this study, my aim is to replicate this empirical process based on a time-span including the COVID-19 pandemic period with two non-linear methods, decision trees & random forest regressions. I choose to add decision trees to the algorithm picked in the original paper as they are the underlying components of the random forests and thus, they help in the understanding of the latter. In the empirical part of their paper, Mercadier and Lardy dealt with 308 listed companies during the 2016-2018 period. But we currently are in a peculiar period, the Covid-19 pandemic that had an impact on the financial markets (Ashraf, 2020a; Ashraf, 2020b; Ali et al., 2020; Zhang et al., 2020; Albulescu, 2021). Here, the study is conducted on 326 listed companies and almost twice the number of dates reaching 298 dates (from 155 dates in

ML2019) from February 3rd, 2016 until September 24th, 2021. According to the National Bureau of Economic Research (NBER[1]) the COVID-19 crisis heavily affected the financial market from February to April 2020.

The remainder of this study is organized as follows. In the next section, I present the literature review on the selected algorithms, and define important words from the machine learning jargon. I then present the formula on which this study is based and the additional attributes that are used in the empirical part. Additionally, I introduce how the pruned decision tree regressions and random forest regressions should be used and their hyper-parameters tuned. Before the conclusion, the results for the accuracies of the models and the features' importance are discussed.

2. LITERATURE REVIEW

Originally, decision trees are decision support tools based on tree-like model of decisions that explicitly represent decisions. Broadly speaking, it is a flowchart tree-like model in which each node stands for a test on an independent variable, and the first node is called the root node. Each outcome of the test is represented by a branch. Finally, a node at which there is no further split is called a leaf node. The entire path that records the set of classification rules goes from the root node to the leaf node and is composed of internal nodes (Figure 1). On the one hand, decision trees are transparent non-linear models that are easy to understand. On the other hand, they are sensitive to a small change in the data and can lead to inaccurate out-of-sample results. Therefore, one may prefer running other algorithms like random forests of decision trees.

[1] https://www.nber.org/research/data/us-business-cycle-expansions-and-contractions

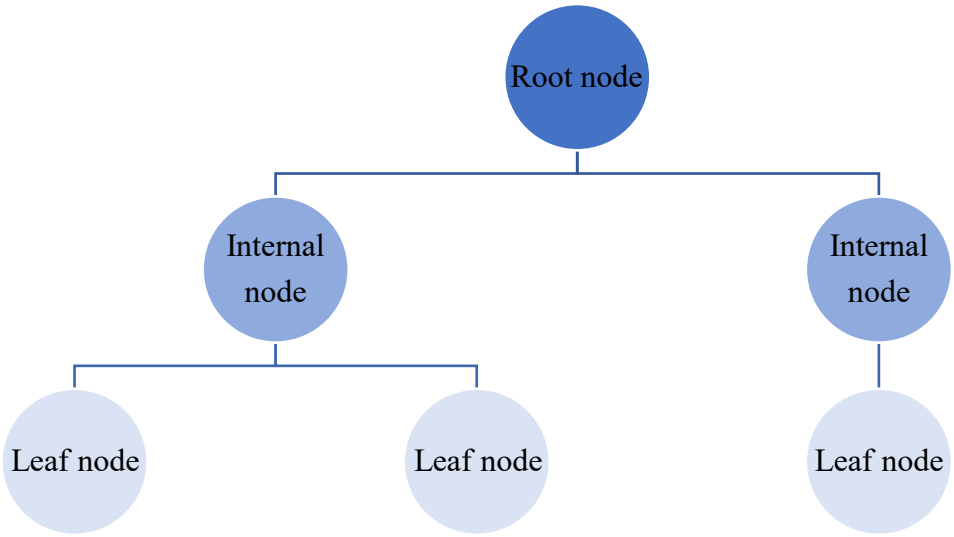


Figure 1: Main components of a decision tree

The ability to perform well out-of-sample is at the core of the field of supervised learning and is linked to the "bias-variance tradeoff" (cf. Geman, 1992; Kohavi, 1996; Fortmann-Roe, 2012; Neal, 2019) (Figure 2). In fact, the expected error on an unseen sample, the square difference between the actual and predicted values, can be decomposed between the bias, variance, and irreducible errors. The bias is the difference between the average prediction of the model and the actual values, and it quantifies the error induced by oversimplification: models with high bias may not pay enough attention to the data in-sample, called training set in machine learning. At the opposite, models with high variance are based on too complex models that pay too much attention to the data sample, but do not generalize well on the out-of-sample data, called test set in machine learning. A model that depicts very well the training set but does poorly on the test set is said to overfit. The role of data scientists is to choose and set the parameters of a model that represents well the training set (low bias) but not too much to avoid a high variance and thus are generalizable (low variance).

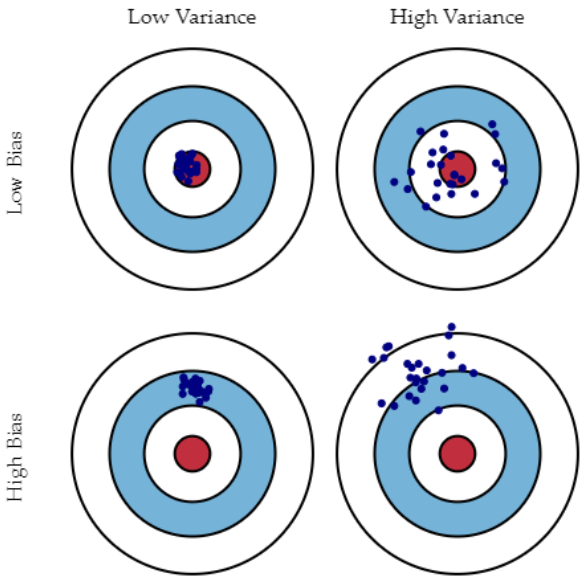


Figure 2: Graphical illustration of bias-variance trade-off

Source: Fortmann-Roe, S. (2012). Understanding the Bias-Variance Tradeoff

In this paper, I run non-linear supervised learning algorithms on the E2C and other independent variables to improve CDS approximations. The first algorithm is a simple decision tree that is pruned to avoid overfitting. Pruning a decision tree means that we do not let it grow until the leaf nodes (cf. Quinlan, 1986; Breslow and Aha, 1997). It is a data compression method that reduces the size of the trees aiming to remove the non-essential and redundant parts. In other words, it reduces the complexity of the model. Various decision tree algorithms exist; Quinlan (1986) developed the Iterative Dichotomer 3 (ID3). ID3 generates a multiway tree which finds at each node the categorical variable that yields the maximum information gain for the targets. Then, with C4.5, Quinlan (1993) removed the restriction that the features must be categorical variables in partitioning the values of continuous attribute into a discrete set of intervals. In addition, C4.5 introduces pruning technology (Pang and Gong, 2009). Finally, Quinlan improved the efficiency of C4.5 with C5.0 that was released under a proprietary license (now available under the GNU General Public License (Quinlan, 2007)).

In statistical learning, classification and regression defines different numerical types of the independent variable, called in machine learning label or target variable. Classification deals with discrete labels and regression with continuous labels. All the previously mentioned decision tree algorithms only handle categorical targets; thus, they belong to classification supervised learning. Classification and Regression Trees (CART, cf. Breiman et al., 1984)

work for regression, as they were developed to support numerical targets. For instance, the predicted class for classification trees is the class that accumulate the most votes while an averaged value is computed for the regression version. CART builds binary decision trees choosing the features and thresholds maximizing the information gain at each node. Decision trees are generally using binary splits, splitting each node into two groups. In fact, multiway splits exist but are not generally a good strategy as they fragment the data too quickly. Moreover, multiway splitting can be achieved by series of binary splits according to Hastie et al. (2009). Nowadays, decision tree algorithms are generally referred to as classification and regression trees. And in this study, I use an optimized version of the CART algorithm.

Decision trees operate as follows, at each node, they must choose an independent variable, called feature in machine learning, and the splitting condition applied to the chosen feature in order to get the highest homogeneity for the child-nodes. When dealing with regression, the reduction in variance can be used to split the node. The idea is to compute the homogeneity of the node and an entirely homogeneous node as a variance of zero. For each split, the variances of the child nodes are computed, and the variance of each split is the weighted average variance of the child nodes. The split that achieves the lowest variance is selected, and this process is repeated until completely homogeneous nodes are achieved for unpruned trees.

Another important concept is the stopping criterion that checks whether a node is an internal or a leaf node. It is common to set a minimum count on the number of training instances assigned to each leaf node. If the count is less than the defined threshold, there is no further split and it is then a leaf node. The more splits in the tree the more complex the decision tree. Thus, it is by tuning the stopping criterion that one can prune the tree and avoid overfitting.

However, it is still possible to decrease the variance while keeping unpruned trees. A solution is to rely on the strong law of large numbers (Bernoulli, 1713) and to run multiple decision trees that are averaged together. This ensemble method (Opitz and Maclin, 1999), called random forest, was developed by Breiman (2001). Earlier work on random forests had been undertaken at Bell Laboratories by Ho (1995, 1998). The multiple CARTs are generated by bootstrap aggregating (“bagging”, cf. Breiman, 1996). In brief, multiple trees are generated from sub-samples of the training set that are randomly drawn with replacement (Figure 3). Random forests integrate an additional property that also contributes to decrease the variance. In fact, at each node, the splitting criterion is chosen from a subset of the available features. An important property of the random forests, used in this article, is that it is easy to

distinguish the features the most used by the algorithm, a computation called feature importance.

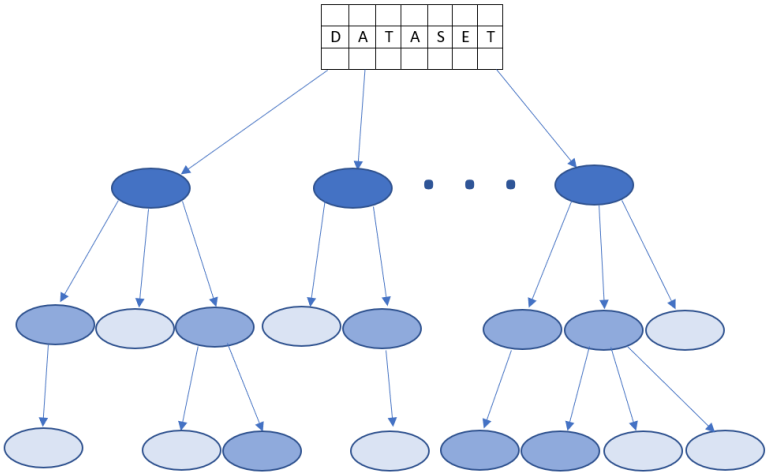


Figure 3: Graphical illustration of random forests

Machine learning algorithms are used in many research fields, with a significant expansion in economics over the last decade. As early as 2007, Tso and Yau empirically emphasized decision trees and neural networks as viable alternatives to stepwise regression models. Malliaris and Malliaris (2015) used a C5.0 decision tree to investigate the drivers of gold prices amongst financial variables. A summary of the improvements machine learning can provide to econometrics may be found in Varian (2014). Within the fields of economics and finance, random forests have recently been used to predict economic recessions (Nyman and Ormerod, 2016), to predict the direction of stock market prices (Khaidem et al., 2017), and to produce an early-warning system for predicting bank failures (Tanaka et al., 2016). Behr and Weinblat (2017) have also applied random forest to study the out-of-sample default propensities of firms in seven European countries. Furthermore, the features importance property of random forests has been used by Yeh et al. (2012) to highlight the main features among market-based information and Moody's KMV model (cf. Vasicek, 1987; Crosbie and Bohn, 2003). The validity of the use of random forests among a wide choice of learning algorithms has been studied, and confirmed, by Fernandez-Delgado et al. (2014). Brummelhuis and Luo (2017) further verifies this in the context of CDS proxy construction.

Even though recent machine learning research is more oriented toward deep learning algorithms (Badrinarayanan et al., 2017; Sun et al., 2018; Sangineto et al., 2018), recent papers in applied machine learning have shown that the outputs of deep learning and ensemble-based methods are comparable, at least for reasonable sample sizes. An argument in

favor of deep learning is its marginally better accuracy (Ahmad et al., 2017) although random forests have in some cases outperformed neural networks (Liu et al., 2013; Rodriguez-Galiano et al., 2015; Krauss et al., 2017).

In line with ML2019, I want to keep the chosen methods as simple and understandable as possible, leading me to lean in favor of decision trees and random forests. The latter can deal with unbalanced and small sample data without deep pre-processing procedures (Liu et al., 2013). Beyond performing an internal cross-validation, random forests require less parameter tuning (Ahmad et al., 2017), an advantage compared to support vector machines (SVM) that requires the tuning of the regularization parameter and the determination of the right kernel, among other parameters. As far as transparency goes, random forests are also preferable to neural networks, which are often considered to be black boxes (although see Shwartz-Ziv and Tishby, 2017, for a recent improvement).

3. METHODOLOGY

3.1 THE E2C FORMULA

The elementary CDS approximation equation used in this study is borrowed to Mercadier and Lardy (2019) and is called the Equity-to-Credit (E2C) formula. This equation is built defining default as the stock price dropping to zero, hence the financial insolvency of the firm, prior to maturity (Black et Cox, 1976). The upper bound of this probability of default is derived from the Gauss inequality (Gauss, 1821), which sharpens Chebyshev's inequality (Chebyshev, 1867) by a $\frac{4}{9}$ factor, assuming that the relevant distribution is unimodal. Following Roy's (1952) conservative "principle of safety first", the inequality is converted to an equality leading to the following E2C formula:

$$C = (1 - R).\frac{4}{9}.\frac{\bar{L}D}{S_0 + \bar{L}D}.\sigma_{S_0}^2$$

This formula is computed using both market-based and fundamental data. The current stock price (S_0), volatility (σ_0) and debt per share (D) of each company are extracted. The methodology of CreditGrades is used to compute the debt per share and the average recovery on the debt ($\bar{L}$) is set at 0.5. The volatility is estimated as the median of various historical and implicit volatilities. The recovery rate (R) of the underlying CDS debt is set at 0.3 in a conservative manner. The authors emphasize that the E2C formula is very intuitive as the

impact of the variables is consistent. Their CDS spread approximation is an increasing function of the corresponding stock volatility and the debt. Meanwhile, an increase in the share value decreases the credit spread value.

They then conduct diverse statistical tests and compare the E2C formula with its closest parent, CreditGrades, and with the actual 5y CDS. They notably identify the closeness between the two models and the actual CDS. Overall, they emphasize that the E2C is statistically slightly closer than CreditGrades to the actual CDS. These results are reinforced by positive results from regressions of the CDS respectively onto E2C and CreditGrades with fixed-effects models. Then, they analyze it further focusing on two universes: unsecured senior debt ratings and industrial sectors. To reduce the influence of outliers, they compare the medians and truncated means. After a presentation of the updated data, I display below the same kind of illustrations including the COVID-19 pandemic.

3.2 DATA & SOFTWARE

The well-balanced panel data sample is a blend of market and fundamental reports information over 326 listed companies from various developed countries. Each company belongs to one out of ten major sectors. The data set spans weekly, every Friday over almost five years (i.e. 298 dates), from 2016-02-03 until 2021-09-24, and is based entirely on Bloomberg data. Data preparation and handling is entirely conducted in Python 3.6 (Python Software Foundation, 2016), relying on the packages “numpy” by Van der Walt et al. (2011), “pandas” by McKinney (2010) and for visual outputs on “matplotlib” by Hunter and Dale (2007) and Excel. In addition, I use “sci-kit learn” by Pedregosa et al. (2011) for the decision trees and the random forests.

3.3 UPDATED RATING & SECTORIAL ANALYSIS

As stated previously, I illustrate this study with two universes, one based on unsecured senior debt ratings and the other on industrial sectors (cf. Figures 4 & 5). The results of the E2C are compared, along with those of CreditGrades, to the actual CDS spreads using the median and truncated mean methods. More precisely, the average is computed after having dropped the extreme 10% top and bottom points. The senior unsecured debt ratings are provided by Standard & Poor’s and Moody’s. Their different rating scales are handled like in ML2019. In brief, if both agencies provide the same grades (based on Santos’s (2008) comparison scale), the corresponding grade is selected; if only one agency gives a grade, this

grade is chosen; and if the grades are different, the worst one is kept remaining conservative. The ten major sectors are: basic material, communications, consumer cyclical, consumer non-cyclical, energy, financials, industrial, utilities, technology and diversified.

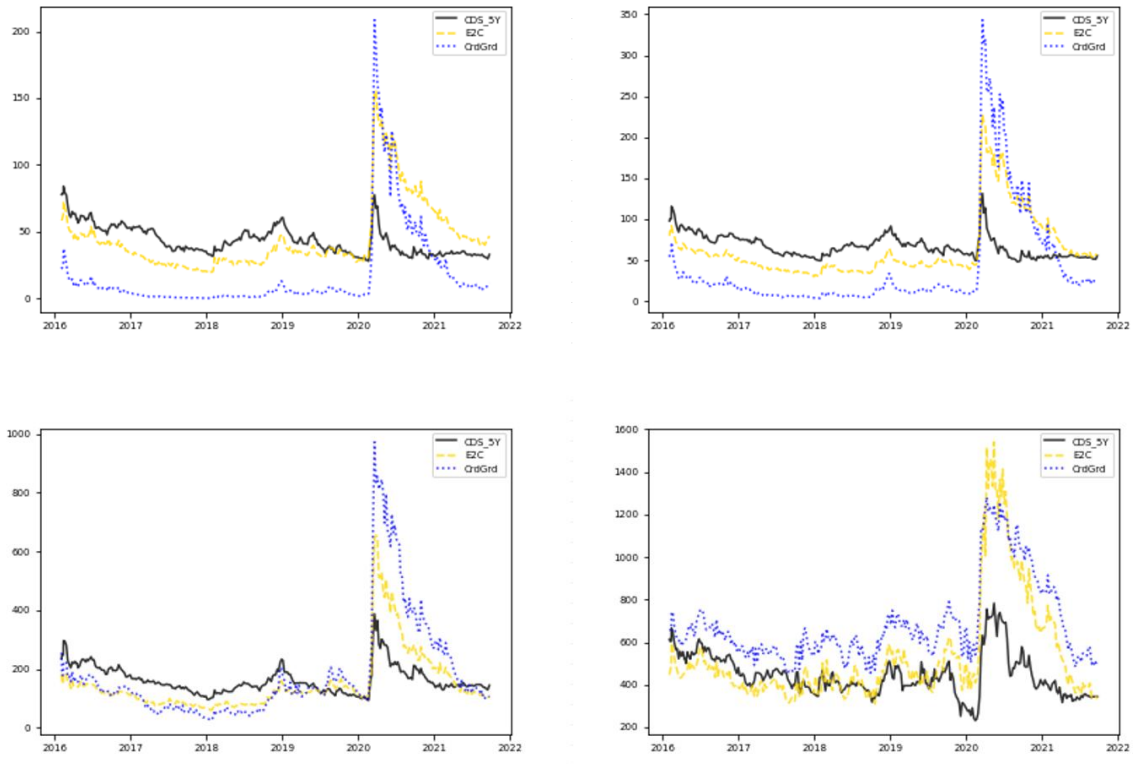


Figure 4: Median based comparisons between original 5y CDS, E2C and CreditGrades approximations in terms of senior unsecured debt rating (A: top left, BBB: top right, BB: bottom left, B: bottom right)

As in ML2019, the E2C formula provides a closer median (or almost alike the double-B grade) to the CDS than CreditGrades, before the COVID-19 pandemic. However, after the COVID-19 shock early 2020, it seems for the safest rating that the E2C takes more time than CreditGrades to decrease back to the actual CDS. Although, the E2C seems to remain closer to the actual CDS, while CreditGrades goes beyond it. For the two riskiest clusters, the E2C is almost always closer than the CDS. It is also interesting to point out that the conservative E2C gives higher results for extreme moments compare to CreditGrades for the riskiest rating during the worst period of the pandemic. Regarding the truncated means, I find similar results to those of ML2019 until the end of 2018. And since 2019, close outputs to those of the medians' analysis are obtained. Additionally, structural models, like the E2C or CreditGrades, do not underestimate spreads for riskier companies (cf. Teixeira, 2007).

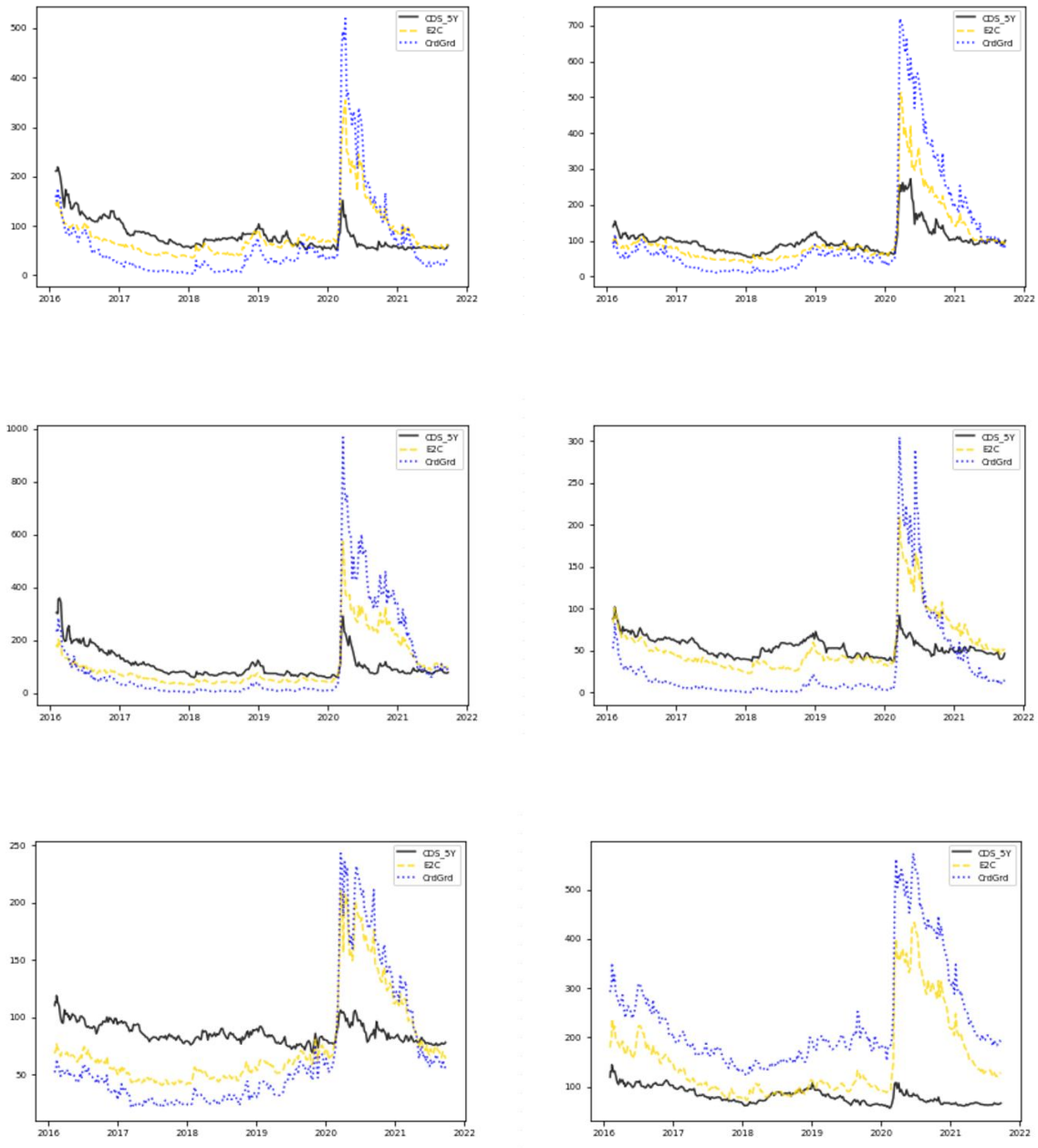


Figure 5: Median based comparisons between original 5y CDS, E2C and CreditGrades approximations in terms of industrial sectors (Basic Materials: top left, Consumer Cyclical: top right, Energy: middle left, Industrial: middle right) and truncated mean based comparisons (Communications: bottom left, Financials: bottom right)

In the original paper, Mercadier and Lardy found results before the crisis, except for utilities and financials in line with Rodrigues and Agarwal (2011), Lardic and Rouzeau (1999) and Eom et al. (2004), who emphasized that structural models generally underpredict the observed credit spreads, at least for low levels of risks. However, the COVID-19 shock increases the approximations well above the actual 5y CDS in all sectors, playing their

conservative role during stressed times. Even after the COVID-19 shock and according to the median, the E2C formula approximates the CDS spreads at least as well as CreditGrades and above all for basic materials, communications, consumer cyclical and non-cyclical, energy and industrial[2]. Similar conclusions are reached with the truncated mean. The E2C results are unquestionably better for basic materials, communications, consumer cyclical, industrial, utilities, energy and financials. It is interesting to acknowledge the rather good estimate provided by the E2C for financials, as CreditGrades and more generally "Merton" models are traditionally poor in assessing this sector.

3.4 FEATURE ENGINEERING

Feature engineering is a key step in machine learning, it is the process of mixing domain knowledge and data formatting to extract and transform relevant features from the original data set.

> "To efficiently perform a basic machine learning algorithm, it is fundamental to preprocess the data, reduce dimensions and extract hand-crafted, domain specific features" (LeCun, 2012).

In line with ML2019 but on a longer timeframe, my aim is to improve the CDS approximation given by the sole E2C formula, integrating additional meaningful input variables. The target variable is the 5y CDS spreads of the selected companies. Overall, I use 4 independent variables and 23 dummy variables. Although I expect the E2C formula to be a major independent variable in the model, it is based on equity information and lacks information on the credit market. The credit market is the one on which are traded the CDS. Thus, the most liquid index on the credit market, the Investment Grade CDS index (IG CDX), is set as feature. In addition, credit rating agencies assess the debtors' probability of default assigning ratings to companies which are set as independent variable. As it is related to a level of stability and liquidity, I also consider the size of the studied companies, measured with the market capitalization. Companies' location and industrial sector are included in the model as well.

Once the features chosen, they are handled as follows. First examples with at least one missing data are removed. Both decision trees and random forests are robust to instabilities and unaffected whether the data are scaled or not, thus I neither check for multicollinearity

[2] More results are available upon request to the authors.

nor standardize the data. However, I use label encoding for the ordinal variable, debt rating and one-hot encoding for the two nominal polytomous variables, i.e., locations and industrial sectors.

3.5 HYPER-PARAMETERS' TUNING

3.5.1 DECISION TREE REGRESSION

Once the algorithm chosen, the first task is to tune the hyper-parameters, that must be distinguished from parameters. In machine learning, a parameter is optimized by the model while a hyper-parameter is to be set by the user beforehand. For instance, the number of clusters for a k-means algorithm is set by the user and not modified by the algorithm. However, the user still needs to select a value for the hyper-parameter. To do so, the most common method is a heuristic called the elbow method. It consists in plotting on the y-axis the sum of squared errors corresponding to each setting plotted on the x-axis. The trade-off is as usual between the complexity and the tolerance to the remaining error of the model. The user will select the parameter that is at the inflection point. Hyper-parameters' tuning is performed on cross-validation sets. Generally, a k-fold cross-validation divides the training set in k subsamples. Then, the methodology alternatively uses k-1 sample as training set and the remaining one as test set. The selected hyper-parameter is the one that minimizes the error or, as I do here, maximizes the accuracy.

For decision trees, three related parameters can be tuned, the maximum depth, the minimum samples split and the minimum samples leaf. The first parameter indicates how deep the tree is allowed to grow. As usual, the deeper it is the more information it holds but the more prone to overfit it is. Minimum samples split indicates the minimum number of samples required for a split to occur. Finally, the minimum samples leaf specifies the minimum number of samples required to be at a leaf node. In other words, minimum samples leaf guarantees a minimum number of samples in a leaf while minimum samples split may create small leaves. In addition, it is possible to tune the maximum number of features used to split each node as for the random forests (cf. section 3.5.2) but it is not used here.

In the graphs below I draw the method by which the decision on the maximum depth tuning is taken considering as input the sole E2C or with the additional universe.

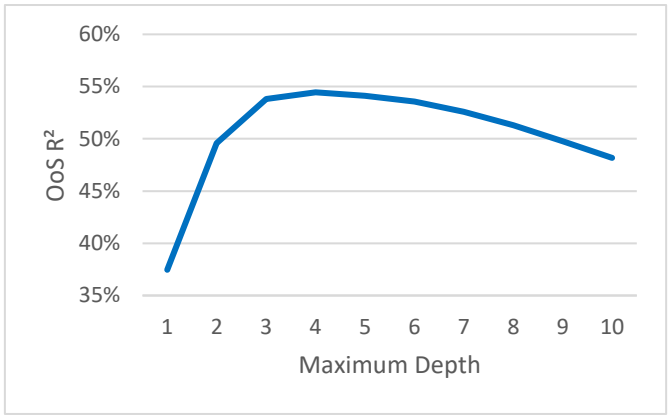


Figure 6: Maximum depth for the decision tree regression onto the E2C

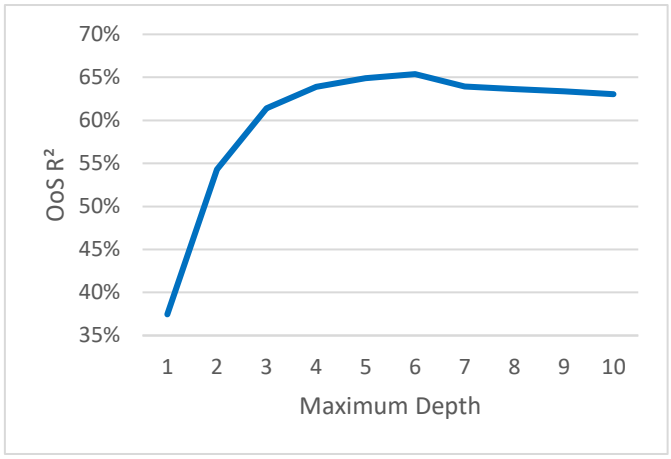


Figure 7: Maximum depth for the decision tree regression onto the entire universe

According to Figures 6 and 7, I allow the trees to respectively grow up to 4 and 6 nodes deep whether inputting the sole E2C or with the entire universe.

3.5.2 RANDOM FOREST REGRESSION

As in the previous section, the first step is to tune the hyper-parameters. It is also possible for this algorithm involving decision trees to set a maximum depth, minimum samples split & leaf, but generally the risk of overfitting is overcome by the number of trees and the number of features (cf. section 2). Therefore, I use unpruned trees, except when the input is the sole E2C for which I set a maximum depth of 4 nodes. In addition, one can consider pruning the trees to obtain quicker computations (as in ML2019 that targeted practitioners). Doing so the tree would make sense for someone frequently running the algorithm. The first parameter I deal with is the number of estimators, to set the number of trees I want the algorithm to generate. As explained in section 2, multiple decision trees are randomly drawn with replacement. One could think that producing more trees would give a more generalized

output, but one may not forget that it would increase the time complexity of the model as well. Another hyper-parameter, the maximum features parameter stands for the maximum number of features to consider when looking for the best split. As explained in section 2 it also contributes to decrease the variance.

In the graphs below I draw the method by which the decision on the number of features and the number of trees is taken considering as input the E2C with the additional universe.

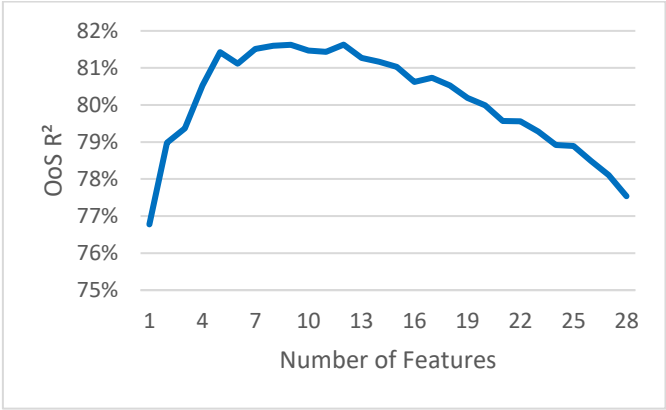


Figure 8: Number of features for the random forest regressions onto the entire universe

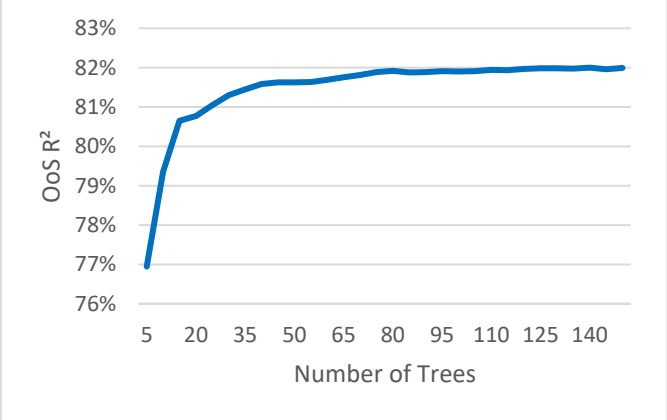


Figure 9: Number of estimators for the random forest regressions onto the entire universe

According to Figures 8 and 9, I allow the trees to check 9 features at each node for splitting, and the number of trees randomly generated is set at 80. From figure 9, we see that one may consider building more than 80 estimators, which would needlessly increase computation time (although it will not degrade the output).

4. RESULTS

4.1 ACCURACY IMPROVEMENT

Like ML2019, I assess the accuracy with the widely known R-squared measure. Where,

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}$$

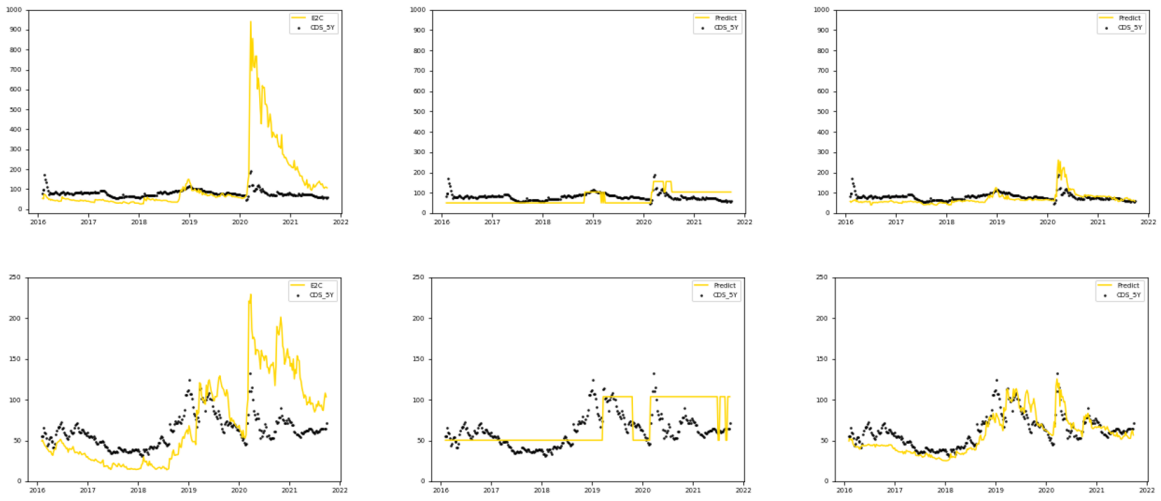
I end up with the following results.

R^2	IS Mean (Std)	OoS Mean (Std)
DecTree (E2C)	58.24% (1.82%)	54.45% (3.26%)
DecTree (All)	81.87% (1.10%)	65.39% (9.36%)
RndFrst (E2C)	58.74% (1.79%)	55.20% (3.28%)
RndFrst (All)	99.35% (0.06%)	81.73% (6.26%)

Table 1: Goodness-of-Fit of averaged decision tree and random forest regressions using the sole E2C or with the additional universe

Running decision trees or random forests on only one input feature (E2C formula) provides similar out-of-sample results slightly above a 50% R-squared. But when I consider the entire universe, decision tree regressions furnish 65% R-squared, and random forest regressions lead to an R-squared close to 82%. Only studying the pre-crisis period, ML2019 find an 87% R-squared. Thus, it seems that integrating the COVID-19 pandemic and almost doubling the considered number of dates only slightly decreases the accuracy, which vouch for the robustness of the method and the choice of the features.

To illustrate my point, I draw for 7 large companies in Figure 10 the 5y CDS benchmarks (dotted line) and the various approximations (solid line) such as the sole E2C (left), multivariate sample using decision tree regressions (middle) and using random forest regressions (right).



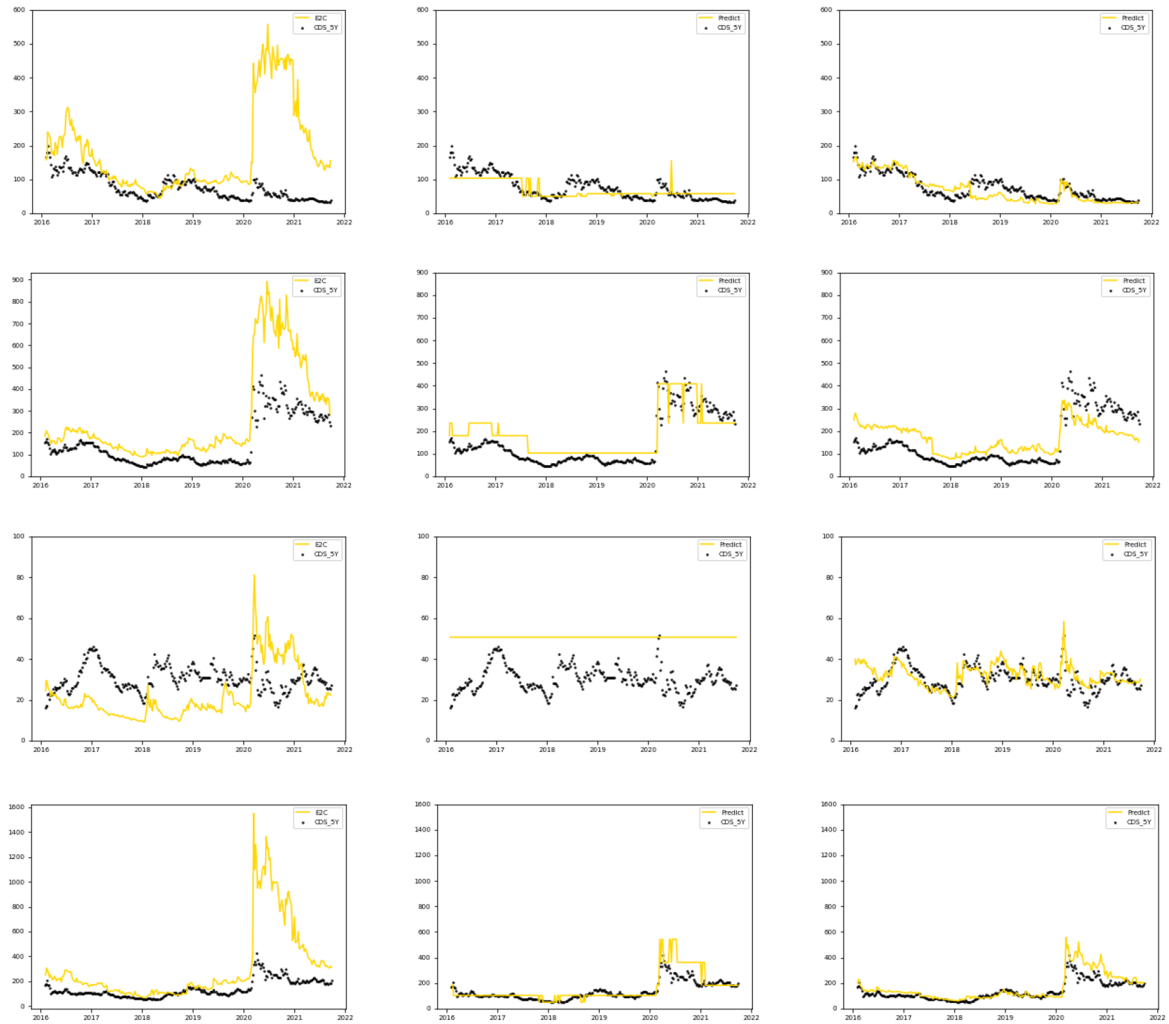


Figure 10: Goodness-of-Fit comparison between the sole E2C approximations (left) and the multivariate set-up handled with decision trees (middle) and random forests (left) for AIG (top), Bayer, BBVA, Deutsche Lufthansa, Pfizer, and Renault (bottom).

The early 2020 shock due to the COVID-19 pandemic has a tremendous impact on the sole E2C approximation. But as expected adding relevant variables and running decision tree and random forest regressions contribute to improve the overall approximation. In line with the results from Table 1, the random forest regressions seem to highly improve the overall prediction regardless of the company.

4.2 FEATURE IMPORTANCE

Random forest regressions inherit an interesting property from the decision trees highlighting the transparency of these algorithms. They both quantify the importance of each variable. This attribute allows me to evaluate the contribution of the variables to the improvement given by these algorithms. I use two methods for assessing this contribution.

The first method called feature importance evaluate the contribution of a given variable to the trees. More specifically, it combines how frequently a variable is used for splitting and the overall improvement it brings each time it is used. For the random forests, the average over all trees of the total contribution of the variables is computed. The second method establishes the importance of a feature by how much permuting it (or shuffling the data) contributes to decrease the overall accuracy. Only using permutation, this method does not affect the distributions and helps to distinguish the importance among correlated features.

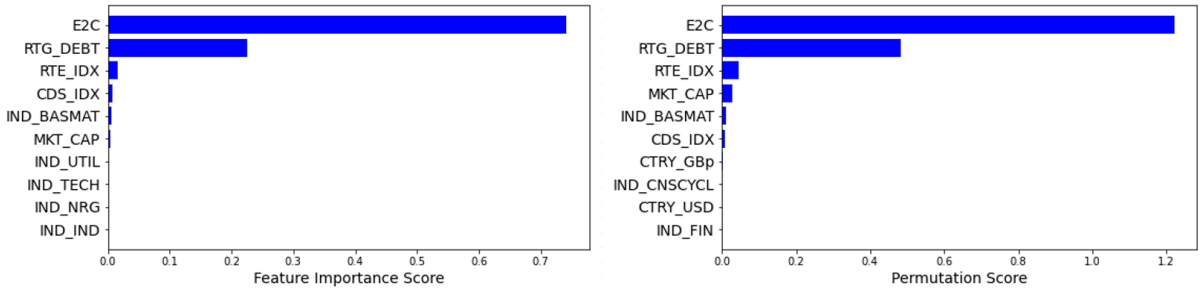


Figure 11: Feature importance assessment for the decision tree regressions using the entire universe

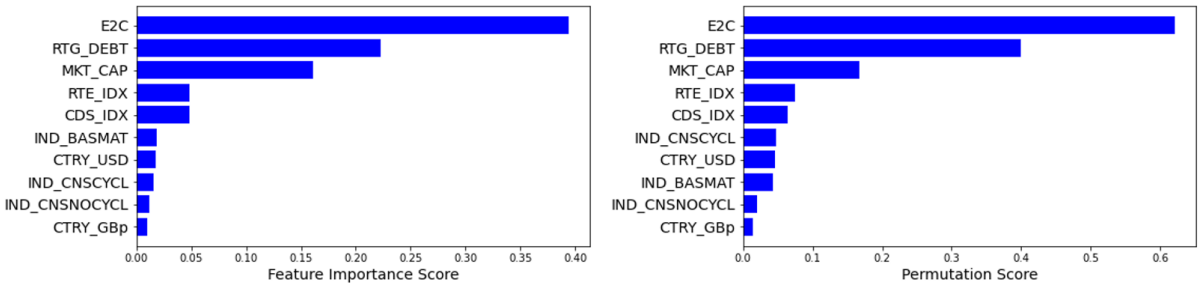


Figure 12: Feature importance assessment for the random forest regressions using the entire universe

In line with ML2019, Figures 11 and 12 confirms, as expected, the prominent contribution of the E2C formula to the approximation of the CDS. Additionally, the debt ratings of the companies play a role in this improvement. However, it is interesting to emphasize that their size, measured by the market capitalization is only highlighted by the random forest regressions. This result is likely caused by the limited authorized depth of the decision trees, as the size is the third largest contributor if the maximum depth is set to 10 (cf. Figure A.1).

5. CONCLUSION

In this study, I use two non-linear supervised learning algorithms based on decision trees to improve the accuracy of an approximation of CDS spreads following the empirical process of Mercadier and Lardy (2019), based on a time-span including the COVID-19 pandemic period. More precisely, both decision tree and random forest regressions are run on the sole E2C formula and on a dedicated multivariate sample. After the introduction of the elementary E2C formula, I present the other features of interest. Certain sections are dedicated to explaining deeper important concepts in machine learning such as feature engineering and hyper-parameter tuning.

Once the algorithms are run, the accuracies provided for each method are discussed. By far, the random forest regressions on the entire universe gives the highest accuracy, i.e., an out-of-sample R-squared around 82%. An accuracy that remains close to the one found by ML2019. Thus, integrating the COVID-19 pandemic and therefore almost doubling the number of dates considered only slightly decreases the accuracy, which vouch for the robustness of the method and the choice of the multivariate sample.

Both methods studied here remain simple among non-linear supervised algorithms, yet the literature supports their efficiency. Moreover, they have the additional benefit of being transparent providing the possibility to enquire the importance of the variables. In line with the original paper, it allows me to emphasize that the most used variable for this approximation is by far the E2C formula. Additionally, the debt rating has an impact on CDS, along with the size to a lower extent.

This study partly answers to ML2019 that proposed for future research to rerun the experiment on a longer time-span. Furthermore, this peculiar time-span includes the highly volatile period observed at the beginning the COVID-19 pandemic.

REFERENCES

Ahmad, M. W., Mourshed, M., and Rezgui, Y. (2017). Trees vs neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption. *Energy and Buildings*, 147:77-89.

Albulescu, C., T. (2021). COVID-19 and the United States financial markets' volatility. *Finance Research Letters*, 38(101699):1-4

Ali, M., Alam, N., and Rizvi, S., A., R. (2020). Coronavirus (COVID-19)- An epidemic or pandemic for financial markets. *Journal of Behavioral and Experimental Finance*, 27(100341) :1-6

Ashraf, B., N. (2020a). Stock markets' reaction to COVID-19: Cases or fatalities?. *Research in International Business and Finance*, 54(101249) :1-7

Ashraf, B., N. (2020b). Economic impact of government interventions during the COVID-19 pandemic: International evidence from financial markets. *Journal of Behavioral and Experimental Finance*, 27(100371) :1-9

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:2481-2495.

Behr, A. and Weinblat, J. (2017). Default patterns in seven eu countries: A random forest approach. *International Journal of the Economics of Business*, 24(2):181.

Bernoulli, J. (1713). Ars Conjectandi: Usum & Applicationem Praecedentis Doctrinae in Civilibus, *Moralibus & Oeconomicis*, Chapter 4.

Black, F. and Cox, J. (1976). Valuing corporate securities: Some effects of bond indenture provisions. *Journal of Finance*, 31:351-367.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24:123-140.

Breiman, L. (2001). Random forests. *Machine Learning*, 45:5-32.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and Regression Trees*. Wardsworth, Belmont, CA.

Breslow, L., and Aha, D. (1997). Simplifying decision trees: A survey. *The Knowledge Engineering Review*, 12(1), 1-40.

Brummelhuis, R. and Luo, Z. (2017). CDS rate construction methods by machine learning techniques. pages 1-51. *SSRN Electronic Journal 2967184*.

Chalamandaris, G. and Vlachogiannakis, N. (2018). Are financial ratios relevant for trading credit risk? evidence from the cds market. *Annals of Operations Research*, 266:395-440.

Chebyshev, P. L. (1867). Des valeurs moyennes. *Journal de mathématiques pures et appliquées*,12:177-184.

Cont, R. and Minca, A. (2016). Credit default swaps and systemic risk. *Annals of Operations Research*, 247:523-547.

Crosbie, P. and Bohn, J. (2003). Modeling default risk. *Moody's KMV*.

Eom, Y. H., Helwege, J., and Huang, J.-Z. (2004). Structural models of corporate bond pricing: An empirical analysis. *The Review of Financial Studies*, 17:499-544.

Escobar, M., Arian, H., and Seco, L. (2012). Creditgrades framework within stochastic covariance models. *Journal of Mathematical Finance*, 2:303-314.

Fernandez-Delgado, M., Cernadas, E., Barro, S., and Amorim, D. (2014). Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research*, 15:3133-3181.

Finger, C. C., Lardy, J. P., Finkelstein, V., Pan, G., Ta, T., and Tierney, J. (2002). Credit-Grades. Technical report, RiskMetrics Group.

Fortmann-Roe, S. (2012). Understanding the Bias-Variance Tradeoff. http://scott.fortmann-roe.com/docs/BiasVariance.html, Accessed date: 5 May 2019.

Gauss, C. F. (1821). Theoria combinationis observationum erroribus minimus obnoxiae (pars prior). *Gauss Werke*, 4:3-26.

Geman, S., Bienenstock, E., and Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, 4(1):1-58.

Guarin, A., Liu, X., and Ng, W. L. (2011). Enhancing credit default swap valuation with meshfree methods. *European Journal of Operational Research*, 214:805-813.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer, New York, Second edition.

Ho, T. K. (1995). Random decision forests. In *Proceedings of the third international conference on Document analysis and recognition*, pages 278-282.

Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:832-844.

Hunter, J. and Dale, D. (2007). The matplotlib users guide. Technical report, R Cran.

Imbierowicz, B. and Cserna, B. (2008). How efficient are credit default swap markets? an empirical study of capital structure arbitrage based on structural pricing model. In *21st Australasian Finance and Banking Conference*.

Irresberger, F., Weiss, G., Gabrysch, J., and Gabrysch, S. (2018). Liquidity tail risk and credit default swap spreads. *European Journal of Operational Research*, 269:1137-1153.

Khaidem, L., Saha, S., and Dey, S. R. (2017). Predicting the direction of stock market prices

using random forest. arXiv:1605.00003, pages 1-20.

Kohavi, R., and Wolpert, D. (1996). Bias Plus Variance Decomposition for Zero-One Loss Functions. *ICML*, 96.

Koutmos, D. (2018). Interdependencies between cds spreads in the european union: Is greece the black sheep or black swan? *Annals of Operations Research*, 266:441-498.

Krauss, C., Do, X. A., and Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, 259:689-702.

Lardic, S. and Rouzeau, E. (1999). Implementing Merton's model on the french corporate bond market. In *AFFI Conference*.

LeCun, Y. (2012). Learning invariant feature hierarchies. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 7583 LNCS, PART 1*.

Liu, M., Wang, M., Wang, J., and Li, D. (2013). Comparison of random forest, support vector machine and back propagation neural network for electronic tongue data classification: Application to the recognition of orange beverage and Chinese vinegar. *Sensors and Actuators B: Chemical*, 177:970-980.

Malliaris, A., G., Malliaris, M. (2015). What drives gold returns? A decision tree analysis. *Finance Research Letters*, 13:45-53.

McKinney, W. (2010). Data structures for statistical computing in python. *Proceedings of the ninth Python in science conference*, 445:51-56.

Mercadier, M., and Lardy, J.-P. (2019). Credit Spread Approximation and Improvement using Random Forest Regression. *European Journal of Operational Research*, 277(1):351-365.

Merton, R. (1974). On the pricing of corporate debt: The risk structure of interest rates. *Journal of Finance*, 29:449-470.

Neal, B., (2019). On the Bias-Variance Tradeoff: Textbooks Need an Update. arXiv:1912.08286, pages 1-63.

Nyman, R. and Ormerod, P. (2016). Predicting economic recessions using machine learning algorithms. arXiv:1701.01428, pages 1-14.

Opitz, D. and Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169-198.

Pang, S., L., and Gong J., Z. (2009) C5.0 Classification Algorithm and Application on Individual Credit Evaluation of Banks. *Systems Engineering – Theory & Practice*, 29(12):94-104.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825-2830.

Python Software Foundation. (2016). Python 3.6.0 documentation.

Quinlan, J., R. (1986). Induction of decision trees. Machine Learning, 1, 81-106.

Quinlan, J., R. (1993). C4.5: programs for machine learning. Morgan Kaufmann.

Quinlan, J., R. (2007). C5.0. https://www.rulequest.com/

Rodrigues, M. and Agarwal, V. (2011). The performance of structural models in pricing credit spreads. *Midwest Finance Association 2012 Annual Meetings Paper*, pages 1-26.

Rodriguez-Galiano, V., M.Sanchez-Castillo, M.Chica-Olmo, and M.Chica-Rivas (2015). Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines. *Ore Geology Reviews*, 71:804-818.

Roy, A. D. (1952). Safety first and the holding of assets. *Econometrica*, 20:431-449.

Sangineto, E., Nabi, M., Culibrk, D., and Sebe, N. (2018). Self-paced deep learning for weakly supervised object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1-1.

Santos, K. (2008). Corporate credit ratings: a quick guide. *Treasurer's Companion*, pages 45-49.

Sepp, A. (2006). Extended CreditGrades model with stochastic volatility and jumps. *Wilmott Magazine*, pages 50-62.

Shwartz-Ziv, R. and Tishby, N. (2017). Opening the black box of deep neural networks via information. arXiv:1703.00810, pages 1-19.

Stamicar, R. and Finger, C. C. (2006). Incorporating equity derivatives into the creditgrades model. *Journal of Credit Risk*, 2(1):3-29.

Sun, Y., Yen, G. G., and Yi, Z. (2018). Evolving unsupervised deep neural networks for learning meaningful representations. *IEEE Transactions on Evolutionary Computation*, PP:1-1.

Tanaka, K., Kinkyo, T., and Hamori, S. (2016). Random forests-based early warning system for bank failures. *Economics Letters*, 148:118-121.

Teixeira, J. C. A. (2007). An empirical analysis of structural models of corporate debt pricing. *Applied Financial Economics*, 17(14):1141-1165.

Tomohiro, A. (2014). Bayesian corporate bond pricing and credit default swap premium models for deriving default probabilities and recovery rates. *Journal of the Operational Research Society*, 65:454-465.

Tso, G., K., F., and Yau, K., K., W. (2007). Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. Energy, 32:1761-1768.

Van der Walt, S., Colbert, S., and Varoquaux, G. (2011). The numpy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13:22-30.

Varian, H., R. (2014). Big data: New tricks for econometrics. *The Journal of Economic Perspectives*, 28:3-27.

Vasicek, O. A. (1987). Probability of loss on loan portfolio. *KMV Corporation*.

Yeh, C. C., Lin, F., and Hsu, C. Y. (2012). A hybrid kmv model, random forests and rough set theory approach for credit rating. *Knowledge-Based Systems*, 33:166-172.

Zhang, D., Hu, M., Ji, Q. (2020). Financial markets under the global pandemic of COVID-19. *Finance Research Letters*, 36(101528):1-6.

Zhou, C. (2001). The term structure of credit spreads with jump risk. *Journal of Banking & Finance*, 25:2015-2040.

APPENDIX

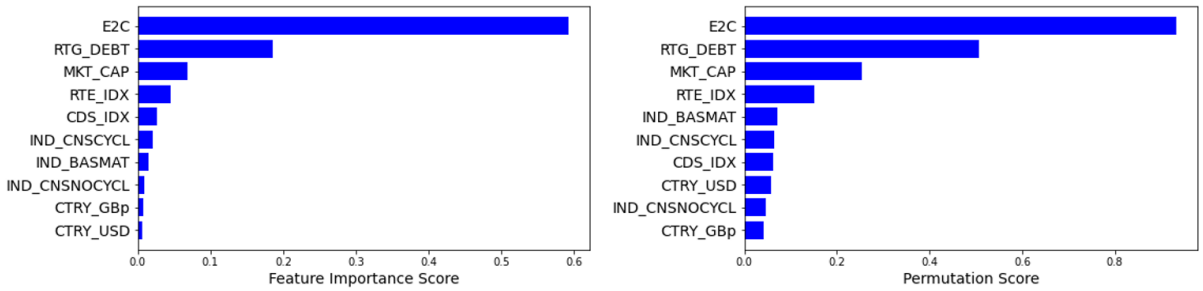


Figure A.1: Feature importance assessment for the decision tree regressions using the entire universe and a maximum depth set at 10