Understanding the performance of machine learning models to predict credit default: a novel approach for supervisory evaluation

Andrés Alonso, and José Manuel Carbó (*)(**)

Abstract

In this paper we study the performance of several machine learning (ML) models for credit default prediction. We do so by using a unique and anonymized database from a major Spanish bank. We compare the statistical performance of a simple and traditionally used model like the Logistic Regression (Logit), with more advanced ones like Lasso penalized logistic regression, Classification And Regression Tree (CART), Random Forest, XGBoost and Deep Neural Networks. Following the process deployed for the supervisory validation of Internal Rating-Based (IRB) systems, we examine the benefits of using ML in terms of predictive power, both in classification and calibration. Running a simulation exercise for different sample sizes and number of features we are able to isolate the information advantage associated to the access to big amounts of data, and measure the ML model advantage. Despite the fact that ML models outperforms Logit both in classification and in calibration, more complex ML algorithms do not necessarily predict better. We then translate this statistical performance into economic impact. We do so by estimating the savings in regulatory capital when using ML models instead of a simpler model like Lasso to compute the risk-weighted assets. Our benchmark results show that implementing XGBoost could yield savings from 12.5% to 17% in terms of regulatory capital requirements under the IRB approach. This leads us to conclude that the potential benefits in economic terms for the institutions would be significant and this justify further research to better understand all the risks embedded in ML models.

PRELIMINARY DRAFT – WORK IN PROGRESS

(Please do not circulate, and do not quote without permission)

Keywords: machine learning, credit risk, prediction, probability of default, IRB system

JEL codes: C45, C38, G21

(*) The authors work as senior economists in the Financial Innovation Division of Banco de España, and appreciate the comments received from José Manuel Marqués, Ana Fernández and Sergio Gorjón, as well all the feedback received in two internal seminar done at the Central Bank.

(**) The opinions and analyses expressed in this paper are the responsibility of the authors and, therefore, do not necessarily match with those of the Banco de España or the Eurosystem.

Index

1. Introduction - Motivation

Recent surveys show that credit institutions are increasingly adopting Machine Learning (ML) tools in several areas of credit risk management, like regulatory capital calculation, optimizing provisions, credit-scoring or monitoring outstanding loans (IIF 2019, BoE 2019, Fernández 2019). While this kind of models usually yield better predictive performance (Albanessi et al 2019, Petropoulos et al 2019)[1], from a supervisory standpoint they also bring new challenges. Aspects like interpretability, stability of the predictions and governance of the models are amongst the most usually mentioned factors and concerns arising from the supervisors when evaluation ML models in financial services (EBA 2017, EBA 2020, BdF 2020). All of them point towards the existence of an implicit cost in terms of risk that might hinder the use of ML tools in the financial industry, as it becomes more difficult (costly) for the supervisor to evaluate these innovative models in order to ensure that all the regulatory requirements are fulfilled. In Alonso and Carbó (2020), we identified a trade-off between predictive performance and supervisory cost of ML models, suggesting a framework to quantify this dilemma. The absence of more regulatory and supervisory transparency is indeed mentioned by market participants when asked about the major impediments that may limit further implementation or scalability of ML technology in the financial industry (IIF 2019, BoE 2019, NP 2020). However in order to define an adequate regulatory approach it is important to understand not only the risks associated with the use of this technology but also the tools available to mitigate this risks. Given the novelty and complexity of some of this statistical methods this is not an easy task. Therefore, prior to enter into the risk analysis it could be relevant to ask what will be the real economic gains that credit institutions might get when using different ML models. While there exists an extensive and growing literature on the predictive gains of ML on credit default prediction, any comparison of results from different academic studies carries the caveat of relying on different sample sizes, types of underlying assets and several other differences, like observed frequency of defaults, which would prevent us from having a robust result to be used for this purpose.

In this paper we aim to overcome this limitation by running a simulation exercise on a unique and anonymized database provided by a major Spanish bank. To this extent we compare the performance of a logistic regression (Logit), a well-known econometric model in the banking industry (BIS 2001), with the performance of the following ML models: Lasso penalized logistic regression, Classification And Regression Tree (CART), Random Forest, XGBoost and Deep Neural Networks. As a result, we will compute, firstly, the benefits in terms of statistical performance of using ML models from a micro-prudential perspective. Evaluating the macro-prudential effects from the use of ML models is out of the scope of this paper.[2]

Finally, we propose a novel approach to translate the statistical performance into actual economic impact of using this type of models in credit risk management. Assuming the Basel formulas for risk-weighted assets and capital requirements in the Internal Ratings-Based (IRB) approach, assuming a retail type of exposure of our dataset, we compute the savings in terms of regulatory capital which could be achieved by using more advanced

[1] For further references, see next section on literature review.

[2] Any policy decision should take into account the potential positive impact of using ML and big-data on financial inclusion (see Barruetabeña 2020, Huang et al 2020), as well as the possibility of having negative side effects on social discrimination (Bazarbash 2019, Jagtiani and Lemieux, 2019) due to the better classification performance of ML models (Fuster et al, 2020).

techniques, in particular XGBoost as the most efficient model in this study, compared to a benchmark extensively used in the industry nowadays, such as Lasso.

The fact that we observe a potentially significant capital savings due to a better statistical performance of advanced ML tools leads us to conclude that further research is needed in the area of supervisory risks in model evaluation. There seems to be an optimal decision to be taken on model selection which will not depend only on the predictive performance, but also on the costs observed to get the approval from the supervisor due to the risks embedded in the implementation of this technology.

The paper is organized as follows. Section 2 provides a literature review on the use of ML models for credit default prediction. Section 3 explains the data and the models used in the analysis. Section 4 contains the comparison of the predictive power, in terms of classification and calibration, for the six chosen ML models. In section 5 we show the economic impact of using XGBoost for calculating the risk-weighted assets and regulatory capital requirements. Section 6 concludes.

2. Literature review

There is an extensive empirical literature on the use of ML models for default prediction in credit risk. We have methodically reviewed it in order to find those papers that compare the predictive power of ML models like Lasso, Random Forest, XGBoost or Deep Neural Networks with the predictive power of a logistic regression (Logit). While most of the papers reviewed focus on one or few ML models we found some that compare the predictive power of an ample class of ML methods, highlighting Jones (2015) and Guegan and Hassani (2018).

In **Graph 1** we summarize the key findings of the reviewed papers (as shown in Alonso and Carbó, 2020). In all of them, the target variable is the probability of default of a loan (depending on the paper, loans could be mortgages, consumer credit, or loans between firms). In order to compare the predictive power in all these papers we retrieve the metric *Area Under the Curve – Receiver Operating Characteristic* (AUC-ROC)[3] as provided by the authors. In the *y* axis we plot the percentage difference in terms of AUC-ROC between the ML model used in the paper and Logit. In the *x* axis we have organized the models in terms of perceived algorithmic complexity as a proxy of the costs to get supervisory approval due to the risks embedded (see Alonso and Carbó, 2020 for further reference[4]). The main insight from **Graph 1** is that most papers find that ML models outperform Logit in classification power. This is true regardless of the technique used and the type of underlying asset of the study. In fact, the predictive gains could be as high as 20%. However, these are very heterogeneous and non-monotonic. Another remark from our literature review is that more

[3] See section 4.1 for an explanation of this measure.

[4] We rank the models by the perceived complexity of the algorithms involved in a standard configuration of each model. First, we distinguish between parametric and non parametric models. Among the non parametric models, we consider that deep learning models are more complex than tree-based ones, since the number of parameters to estimate is higher and its interpretability is more complex, requiring the use of additional techniques. Finally, we consider reinforcement learning and convolutional nets as the most complex models, since the former needs a complicated state/action/reward architecture, while the latter entails a time dimension and thus an extra layer of complexity with respect to deep neural networks. Metrics like the VC dimension (Vapnik-Chervonenkis, 1971) could be used to account for the capacity of the algorithms, when a particular architecture is taken into account; but for comparison reasons we just aimed to illustrate the evolution on the "structural" algorithmic complexity, in terms of ability to adapt to non-linear, highly dimensional problems. Therefore, changes to this rank could be considered depending on the set of parameters and hyper-parameters considered in each model.

complex models like deep neural networks are not necessarily the best predictors in the field of credit risk management.

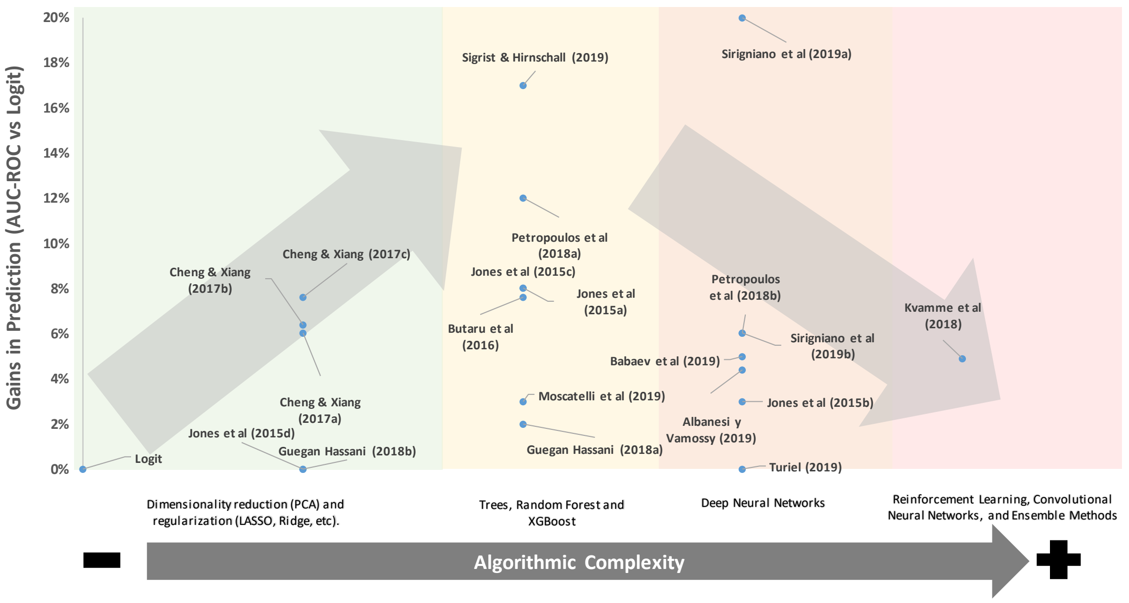
Our contribution to this literature is that we are able to assess the predictive performance of a wide range of ML methods under different circumstances (different sample sizes, and different amount of explanatory variables, as shown in Section 4) using a unique database on consumer credits granted by a big Spanish bank. Unlike the aforementioned comparisons in the literature, this allows us to test whether the statistical performance comes from an information advantage (associated to the access to big amounts of data) and/ or model advantage (associated to ML) when comparing these innovative tools to traditional quantitative models, as suggested by Huang et al (2020).

We also contribute to the literature by assessing the economic impact of using ML for credit default prediction. Khandani et al (2010) and Albanesi and Vamossy (2019) computed the Value Added (VA), as net savings to lenders of adjusting credit lines to borrowers based on the predictions of ML models. This method, while useful, has its limitations. First, it is limited to the assessment of ML for credit scoring, while we aim to evaluate models in a broader area of credit risk management. Second, it is a backward looking metric, as it assumes that loans or credit lines could be granted or cut retrospectively on the outstanding portfolio[5]. We, instead, propose to monetize the impact through the comparison of calculated risk-weighted assets and capital requirements under a baseline scenario, using Lasso[6], against a scenario in which the credit institution would have chosen to implement a more statistically efficient model, like XGBoost. We show that the latter scenario can yield savings from 12.5% to 17% in terms of regulatory capital requirements under an IRB approach, depending on the corresponding risk factors associated to the type of exposure or underlying assets. This is, to the best of our knowledge, the first attempt to measure the impact of using ML methods in terms of regulatory capital savings. This impact could be interpreted as a floor amount, since it does not account for the potential benefit of using the model out-of-sample. Therefore, while conservative, this estimated amount could be immediately materialized by the credit institution, complementing the exercise that could be additionally carried out through the estimation of the VA.

[5] See section 5.1 for an explanation of this method.

[6] Although in the literature review the comparison in the evaluation has been performed using Logit as benchmark, we assume for this exercise that currently the use of a logistic penalized regression with Lasso is common practice in the banking industry.

Graph 1. Trade-off between predictive power and algorithmic complexity



Source: Alonso and Carbó (2020)

3. Data collection and ML models

An anonymized database from Banco Santander has been used to conduct this analysis. It contains data from a subset of consumer credits, granted by the aforementioned bank in unspecified dates. This data has been completely and previously anonymized by Banco Santander through an irreversible dissociation process in origin which prevents the possibility of identifying costumers in any way. The dataset contains information from more than 75,000 credit operations which have been classified into two groups, depending on whether they resulted on default or not. Additionally, each operation has a maximum of 370 risk factors associated to it, whose labels or description have not been provided. Consequently, the nature of these variables is unknown to us, and they cannot be used to establish the identity of the customers they refer to. Around 3.95% of the loans resulted in default, but the data has no temporal dimension, so we do not know when the loan was granted, and if resulted in default, when it happened.[7]

As mentioned in the introduction, we will compare the predictive performance of Logit vs several ML models. In particular, we have chosen Lasso penalized logistic regression, Classification and Regression Trees (CART), Random Forest, XGBoost and Deep Neural Networks[8] because they are amongst the most cited ones in the literature review.[9] We have conducted our analysis using Python and open source libraries like Sklearn and Keras. The hyper-parameters have been chosen according to standard cross-validation techniques,

[7] Therefore, we will focus on the estimation of probabilities of defaults point-in-time, leaving out of the scope of this work any assessment on the impact of macroeconomic variables that could explain observed defaults through-the-cycle.

[8] Our benchmark Neural Network has 5 layers, with 3 hidden units of 300, 200 and 100 neurons. We have selected this architecture after implementing the proper cross-validation and hyper parameter tuning. Our main results are not significantly affected by choosing other variations of Neural Networks.

[9] For an introduction into the functioning of each model, please see WB (2019).

as the purpose of our exercise is neither feature engineering nor optimization, but comparing between correctly calibrated models. All results have been tested out-of-sample, with a partition of 80% train sample and 20% test sample. As it is common in the literature, input values have been standardized by removing the mean and scaling to unit variance[10].

4. Predictive performance

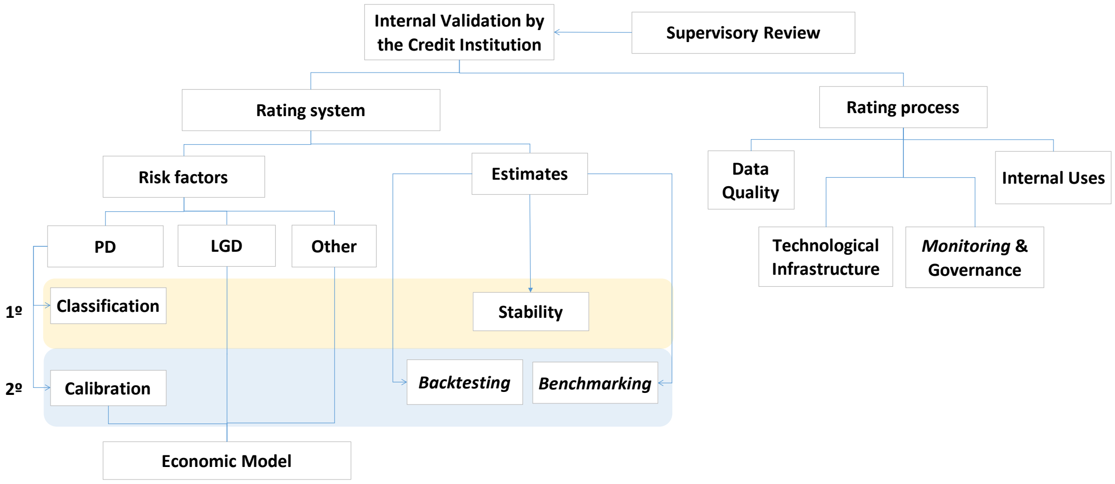
To assess the predictive performance of the 6 ML models we will focus on two measures: classification and calibration. Classification means the ability of the model to discriminate defaulted loans from those that have been repaid, being able to classify them in different risk buckets. We will use the AUC-ROC *or Area Under the Curve of the Receiving Operating Characteristic* (Fawcett, 2005) in order to measure the discriminatory power (BIS, 2005).[11] On the other hand, calibration refers to the quality of the estimation of the probability by looking, per risk bucket, at how good the estimated probability fits the average default rate. To this purpose we will use the Brier score (BIS, 2005) to measure how precise the estimations are, along with calibration plots, or reliability curves, in which we will divide the predictions into groups, and for each group we will compare the average estimated probability of default with the corresponding average default rate observed. For both measures, we perform a sensitivity analysis in two dimensions, simulating the impact in the AUC-ROC and Brier score of the models for different sample sizes, and for different number of available features.

The reason why we have decided to use these two measures to pursue the evaluation of the performance of the ML models is that they are explicitly mentioned in the supervisory process for the validation of IRB systems, which we find to be the most complete framework to understand the potential and limitations of these predictive models when applied to credit risk management (Alonso and Carbó, 2020). In this sense, in **Figure 1** we represent a simplified IRB validation process, which helps us to understand how the deployment of a predictive model is seen from the supervisor's lenses, in particular, when having to assess its adequacy for regulatory capital purposes.

[10] Our results do not change ostensibly if we use input values without standardization, but standardizing them helps to reduce computing time, especially in the case of deep neural networks.

[11] There are other metrics that evaluate the performance of a classifier, like F1, Gini index, recall, precision and accuracy. We choose AUC because is the most used metric across the papers we reviewed in Graph 1 and one of the most popular metrics in the literature (Dastile et al, 2020). We will additionally use recall as a robustness check.

Figure 1. IRB validation process



Source: Alonso and Carbó (2020)

For a supervisor, there are two separated phases when evaluating the adequacy of an IRB system. First, as seen in the left hand side of the previous Figure, a supervisor should carry out an assessment of the design of the rating system. In credit risk, institutions have to estimate several risk factors, like the Probability of Default (PD), Loss-Given-Default (LGD), or even credit conversion factors or maturity adjustments. As a general rule, institutions have to provide their own estimates of PD and rely on standard values for other risk components. In this paper we will assume this is the case.[12] The estimation of the PD is a two-folded task. First institutions are required to identify the risk in different buckets, discriminating those exposures which are more risky from the rest. Secondly, the risk must be quantified. To this purpose, the risk buckets must be well calibrated, resembling the observed default rate. Once the risk factors are estimated, they will be plugged into an economic model as inputs in order to compute the (un)expected losses, which in the case of regulatory capital requirements, comes from the Basel framework.

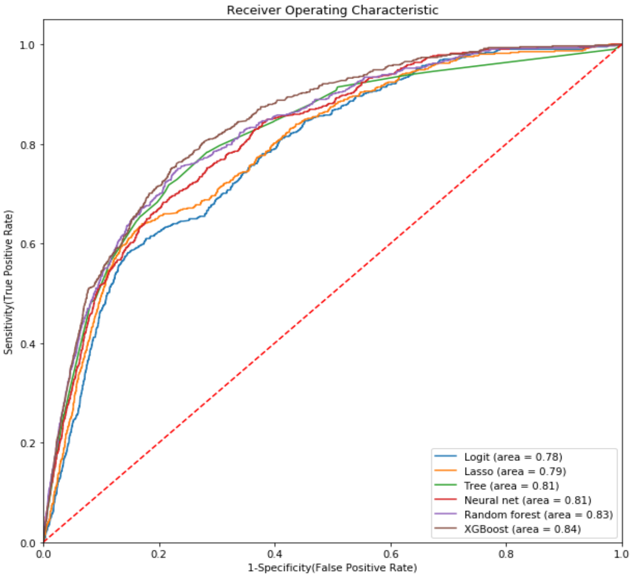
In sum, to understand the benefits of ML models applied to estimating PDs, it is not enough to evaluate the models in terms of discriminatory power, but we must get a grasp as well on the calibration performance. Once this work is done, supervisors will get deeper into the rating process, which usually includes an investigation on the data sources, privacy of the information and quality of the data sets, technological infrastructure required to put the model into production, and its governance, all subject to the use that each institution gives internally to these models.

4.1. Classification

In this section we use the AUC-ROC to study the discriminatory or classification power of the selected models. As shown in **Figure 2**, this curve plots the true positive rate (TPR) vs the false positive rate (FPR) at different classification thresholds.

[12] As further explained in Section 5.1.

Figure 2. Comparison of AUC-ROC per model



$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

Where TP (true positives) are the loans that, having defaulted, are correctly predicted as such, FN (false negatives) are the loans that, having defaulted, are incorrectly predicted as non-default, FP (false positives) are the loans that did not default but were predicted as default, and TN (true negatives) are the loans that did not default and were correctly predicted as non-default. For each threshold, if a loan has a probability of default higher than such threshold, then we classify it as defaulted. Therefore, the lower the threshold, the higher the TPR and the lower the FPR (upper right of the AUC curve). On the other hand, the higher threshold, the lower the TPR and the higher the FPR (bottom left of the AUC curve). In **Figure 2** we plot the out-of-sample AUC-ROC curves for the six models. The curves show a nonlinear trade-off between TPR and FPR. The discriminatory power is given by the area under the curve. For reference, we plot a dotted 45 degrees line. This line yields an area of 0.5, and it represents a decision rule that categorizes randomly a binary response. The further up from the red dotted area, the more classification power the model would have. In our estimation Logit achieves a 0.79, Lasso 0.8, CART 0.81, Deep Neural Net 0.81, Random Forest 0.83, and XGBoost 0.84. The results are in line with our previous findings on the literature review, which suggest that ML models have better predictive performance than Logit, but deep neural networks do not necessarily outperforms tree based methods.

From a credit institution point of view, the cost of a FP (not granting a loan to a performing counterparty) will presumably have a smaller impact on the benefits than the importance of getting a TP correctly (not granting a loan to a non-performing counterparty). Therefore, model selection rules for credit scoring would usually take into account that the actual cost of having a FN outweighs the opportunity cost of a FP. In this sense, the AUC-ROC treats both costs symmetrically, while from an economic point of view both are not equally important. Bearing this in mind, we propose an additional exercise in which we compare the TPR or recall of each of the ML models. In order to compare the TPR, we need to specify which threshold we consider to decide when a loan will default or not. We choose from 10% to 30% thresholds, since a loan with an estimated default probability of ca. 10% is associated with speculative grade and 30% corresponds to average default rates observed in ratings at least CCC+ by Standard & Poors and Moody's (Cardoso et al 2013). Both levels are therefore representative of early warning or limits when deciding to grant a loan in any credit scoring system. The results are in **Table 1**.

Table 1: True Positive Rata for different classifier thresholds

Method	TPR, Classifier threshold = 10%	TPR, Classifier threshold = 20%	TPR, Classifier threshold = 30%
Logit	33%	6%	1%
Lasso	37%	7%	2%
Tree	49%	18%	4%
Random Forest	55%	9%	2%
XGBoost	55%	24%	8%
Deep learning	52%	16%	2%

With a classifier threshold of 10%, the ranking of ML models in terms of TPR is the same as in our benchmark exercise, when we compared ML models in terms of the AUC metric. XGBoost and Random Forest have the highest TPR, around 55%, followed by Deep learning, Tree, Lasso and Logit. If we consider a classifier threshold of 20% or 30% instead, then XGBoost continues to be the ML model with highest TPR, but Random Forest falls behind Deep learning and Tree. In any case, for each possible classifier thresholds, ML models outperform again traditional methods like Logit.

From **Table 1** we can see that resulting TPRs are relatively small (never above 60%) even for low classifier thresholds. This happens because in our dataset there are approximately 3.95% of defaulted loans. While it is not a heavily imbalanced dataset, we test the robustness of our results by performing two additional exercises in which we balance our dataset, first by giving more weight to observations which defaulted in the loss function, and second, by performing oversampling techniques. The results are in the Appendix. The main conclusion is that these rebalancing techniques do no change the main results from our benchmark exercise, and the ranking among algorithms remain the same. Taken this into consideration, we continue to work with our original dataset.

We then analyse if the model´s classification power depends on the number of observations and features available. Our aim is to statistically isolate any information advantage due to a better access to big amounts of data from a hypothetical model advantage (see Huang et al, 2020). To this purpose, first we compare the classification performance of each model for different sample sizes. We perform 400 simulations, and for each of them a random number of loans, from 1,000 to 65,000[13], is selected. In **Figure 3** we show the area under the curve of each model for different sample sizes, so that we can find the model with the best classification performance depending on the sample size. Random Forests and XGBoost outperform the rest of the models when 5,000 observations or more are included. It is often believed that algorithmically complex ML methods surpass traditional linear models because they can handle a larger amount of data (the so-called, information advantage). But this exercise shows that, given the same amount of data, Random Forest and XGBoost always exceed the discriminatory power of Logit or Lasso thanks to the non-

[13] Randomly we select a number from 1,000, 5,000, 10,000, 15,000 up to 65,000 (12 groups in total, around 33 simulations per group).

linearity nature of their algorithms, even when a relatively small amount of data (5,000 observations) is used. In this sense, ML techniques offer a model advantage, adding value on top of what traditionally might be understood as *big data*. All six models experience an increase in their classification performance when more observations are included, but the slope of this gain is smaller for Logit (blue line) and Lasso (yellow line) from 10,000 observations onwards. This means that traditional quantitative models do not benefit as much when more data is available. On the other hand, Logit and Lasso can outperform the rest of models when the sample is 5,000 loans or less. **Figure 3** also shows that Deep Neural Networks only outperform Logit and Lasso when more than 20,000 observations are available. In the appendix, **Figure 14** shows these simulations for the six models along with their 95% confidence intervals.

Secondly, we compare the classification performance of each model for different number of available features, in order to isolate the second dimension of a potential information advantage[14]. We perform again 400 simulations in which we select all available observations (75,000 loans), but in each simulation we choose a random number of features, from 125 to 375[15]. **Figure 4** shows the AUC-ROC of each of the models for different number of features. The area under the curve increases for every model as we increase the number of features available. The magnitude of the increase is very similar for all the models, except the CART that seems to benefit the most from the inclusion of more features. Again, Random Forest and XGBoost show the best results, since they outperform the rest of the models regardless of the number of features chosen. We also show in **Figure 15** (in the appendix) these simulations along with their 95% confidence intervals. [16]

Figure 3. Simulation of AUC-ROC performance to sample size

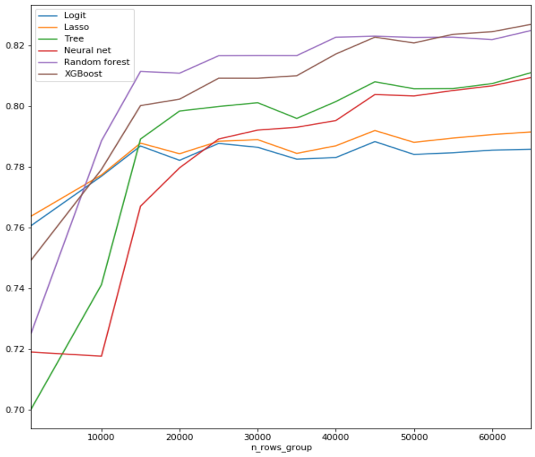
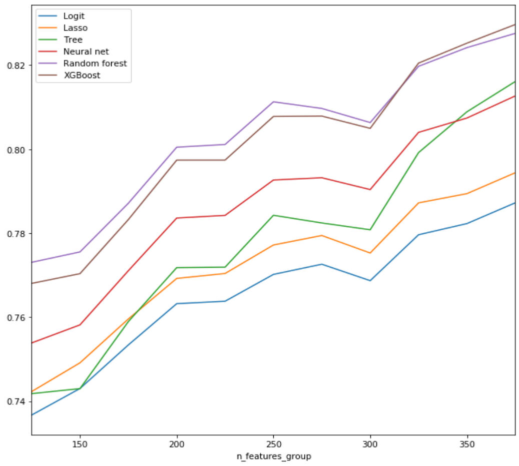


Figure 4. Simulation of AUC-ROC performance to number of features



[14] Assuming that any information advantage may come from the access to a larger MxN dataset, where M is the number of observations (length) and N the number of features (width).

[15] Randomly we select a number from 125, 150, 175, up to 375 (12 groups in total, around 33 simulations per group).

[16] These results are in line with Huang et al (2020), where a model advantage was found using a dataset from the Fintech industry in China. While in that study the features' selection was done discretionally (traditional vs innovate explanatory variables), in our exercise we used a randomized selection of features, which allows us to add statistical robustness to the conclusion that a model advantage exists on top of information advantage.

4.2. Calibration

In this section we will use the Brier score to study the calibration power of the six ML models. We will also use reliability curves in which we will divide the predictions into groups depending on their estimated probability of default, and for each group we will compare the average probability of default with the rate of defaulted loans observed over total loans in that group.

The Brier score is the main measure to quantify the accuracy of a probability forecast (BIS, 2005). The formula to compute this metric is:

$$BrierScore = \frac{1}{N}\sum_{t=1}^{N}(f_t - o_{t)})^2$$

Where *N* is the number of observations, *f* is the predicted probability of default, and *o* is the class of the observation (1 if default, 0 otherwise). We perform the same two exercises as with classification: we compute the Brier score for different sample sizes (from 1,000 to 75,000) and for different number of features (from 125 to 375). **Figure 5** shows for each model the resulting box plots from 400 simulations with different sample sizes. It can be seen that for most of the simulations and models, the Brier score is within a range of 3% and 4.4%. Differences among models are very small. Brier score values are small and similar among models due to the fact that there are only 3.95% defaulted loans in the whole sample. For illustrative purposes, if we were to consider the whole sample, and we used a model that assigns probability of default equal to zero for all the loans, the Brier score of that simple model would be 3.95%. Still, we can see in **Figure 5** that the models with the lowest average Brier score are Random Forest and XGBoost. The six models have a Brier score of 3.7% for Logit, 3.6% for Lasso, 3.5% for the CART and Deep Neural Network, and 3.4% for XGBoost and Random Forest when the whole sample is used.

Figure 5. Sensitivity of Brier score to sample size

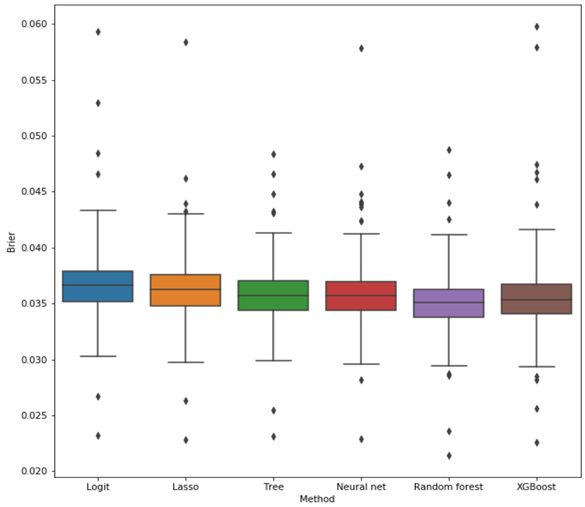
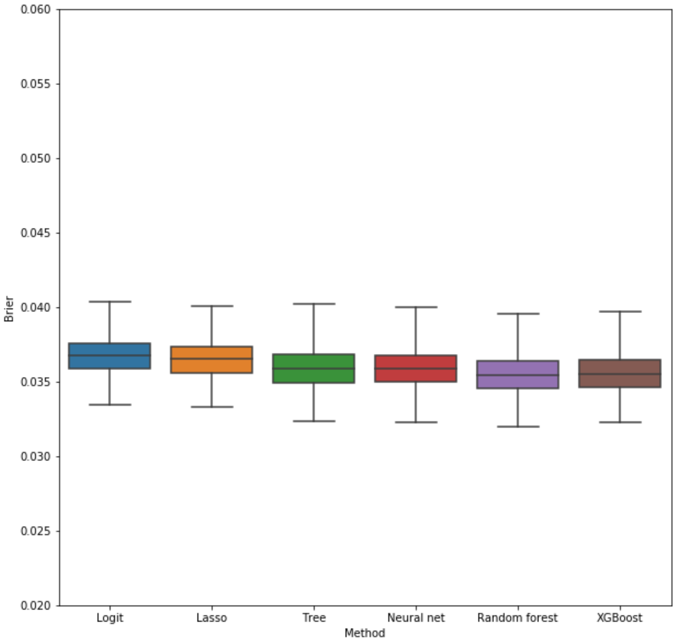


Figure 6 shows for each model the resulting box plots from 400 simulations with different number of features available. It can be seen that for most of the simulations and models, Brier score is within a range of 3.3% and 4%. We observe that Brier Scores are more homogeneous across simulations when changing the number of features than when changing the number of observations. This might be because now each simulation has all observations available, so the amount of defaulted credits is the same across simulations,

while in **Figure 5** the percentage of defaulted credits might differ from one simulation to another. In any case, XGBoost and Random Forest also achieve the lowest Brier scores when changing the number of features available, reinforcing the existence of a model advantage from a calibration point of view.

Figure 6. Sensitivity of Brier score to number of features



Since differences in Brier Score are so small among models, we propose the additional calibration exercise. We run 200 simulations for each model, only changing the train-test partitions, with all observations and all features available for each simulation. We have grouped the predictions of each model into 13 buckets[17], depending on the estimated probability of default. **Figure 7a** shows reliability curves for each of the models. The *x* axis represents the estimated probability of default of each bucket, and the *y* axis has the proportion of defaulted loans over total loans for each bucket. The 45 degrees line represents a perfect calibration. For example, a perfect calibration would imply that a bucket with 20% of estimated probability of default should contain a 20% of defaulted loans. All models seem to perform very similarly for the first two buckets. However, for predicted probabilities above 10%, the performance of the models differs. For predicted probabilities between 10% and 20%, Logit (blue line) and Lasso (yellow line) underestimate the probability of default with respect to the observed default. For estimated probabilities above 20%, all models overestimate the probability of default with respect to the observed default rate. But Logit and Lasso are the models that overestimate the most. On the other hand, XGBoost and Random Forest are the models that are closer to the 45 degrees line. Since these results are based on multiple simulations, we must take into account the variance of the observations. **Figure 16** in the Appendix shows the same results of **Figure 7a** but in 6 subplots (one for each model) in which we display the 95% confidence intervals. This way we can understand better the accuracy of the calibration. It can be seen that for Logit and

[17] The bucket distribution is as follows: Bucket 1 has loans with PD between 0% and 5%, bucket 2 PD between 5% and 10%, bucket 3 PD between 10% and 15%, bucket 4 PD between 15% and 20%, bucket 5 PD between 20% and 25% and bucket 6 PD between 25% and 30%. Buckets seven and above contains intervals of 10% of PD each, up to PD 100%.

Lasso, the 45 degrees line lays out of the calibration points' confidence interval from the third bucket (around 12% of probability of default) onwards. The rest of ML models perform better, especially Deep Neural Network, Random Forest and XGBoost, for which the 45 degrees line always lays on the confidence interval for all buckets.

Figure 7a. Reliability curve

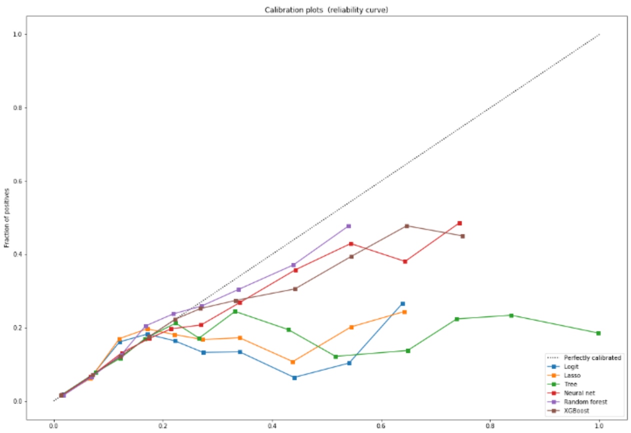
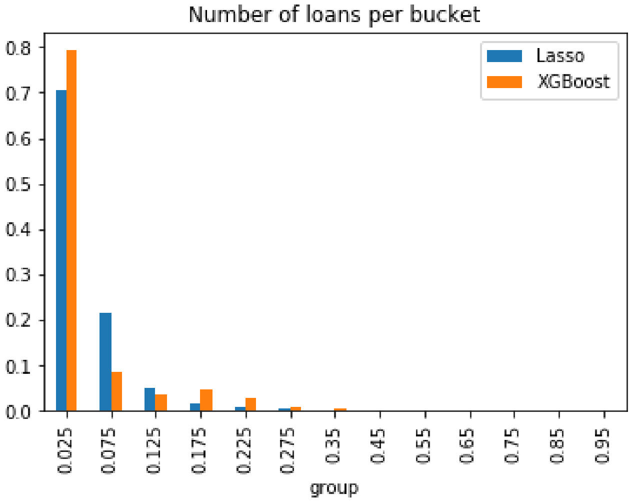


Figure 7b. Distribution of loans



Looking at **Figure 7a** it might seem that the difference in calibration power between XGboost and Lasso or Logit is higher than what their Brier scores suggest. This is explained by the fact that most observations have a probability of default below 10%. In **Figure 7b** we show the amount of credits in each bucket for Lasso and XGBoost. It can be seen that 80% of the credits have probabilities of default below 10%. Therefore we must acknowledge that XGBoost outperforms Lasso and Logit especially for probabilities above 10%, but there are fewer amount of credits in those buckets.

Therefore, as most of the predictions have a probability of default below 30%, we propose another calibration plot in which we group the predictions into more granular buckets, focusing only in probabilities below this threshold[18]. This way we can assess the performance of the models for lower and more common probabilities in the field of credit default prediction. The results are shown in **Figure 8a.** It can be seen that Lasso and Logit tend to underestimate the probability of default for predictions up to around 3%, then overestimate the probability of default for probabilities from 5% to 7%, underestimate again for probabilities between 10% and 20% (as suggested as well by **Figure 7a**), and overestimate for probabilities above 20%. The rest of ML models are closer to the 45 degrees line. In the appendix we show in **Figure 17** the results of **Figure 8a** but in six subplots (one for each model) and with 95% confidence intervals. It confirms that ML models like Deep Neural Network, Random Forest and XGBoost calibrate better. For

[18] The new bucket distribution is as follows: Bucket 1 has loans with PD between 0% and 1%, bucket 2 contains PD between 1% and 2%, bucket 3 contains PD between 2% and 4%, bucket 4 contains PD between 4% and 6%, bucket 5 contains PD between 6% and 8%, bucket 6 contains PD between 8% and 10%%, bucket 7 contains PD between 1% and 12%, bucket 8 contains PD between 12% and 15%, bucket 9 contains 15% and 20%, bucket 10 contains PD between 20% and 30%, and bucket 11 has up to PD 100%.

referential purposes, in **Figure 8b** we show the amount of credits in each bucket for Lasso and XGBoost with this new categorization of buckets.

Taking everything into account, we can conclude that XGBoost and Random Forest clearly outperform the other models, especially in classification, but also in calibration, although this is definitely a more difficult task for all the models. Finally, CART and Deep Neural Network have similar performances, always above Lasso and Logit. The main conclusion of this section is that ML models outperforms Logit both in classification and in calibration, existing a model advantage that can be statistically isolated from an information advantage. Nevertheless, most complex models like Deep Neural Networks, do not necessarily predict better neither in terms of classification nor calibration.

Figure 8a. Reliability curve

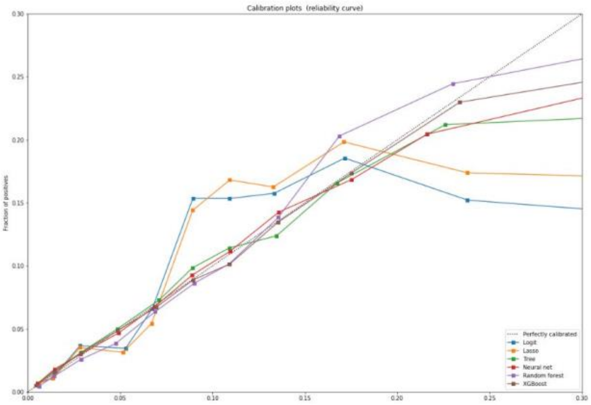
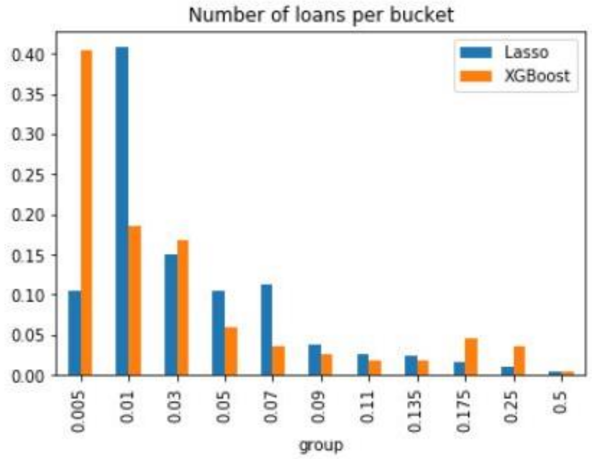


Figure 8b. Distribution of loans



5. Economic impact of using machine learning

In section 4 we found that ML models have better predictive power both in terms of classification and calibration than Logit or Lasso, regardless of the sample size and number of available features. In this section we wonder how to translate this statistical result into real business metrics. In particular, we aim to answer: which is the potential economic impact for a credit institution of using one of these ML models instead of traditional quantitative methods in credit default prediction in real business conditions?

One approach to measure the economic impact of better predictions is finding which loans could have been granted in case of counting with a better predictive model. This means either working out-of-sample, or using a subset of the portfolio and thinking retrospectively. This last approach is followed in Khandani et al (2010) and Albanessi and Vamossy (2019), who estimate the Value Added (VA) of using ML models by comparing the impact of true positive (TP) and false negative (FN) estimates, on the outstanding portfolio. The VA would be therefore the sum of the savings gained due to a correct decision not to grant a loan based on the TP rate, offset by the opportunity costs due to the lost return on those rejected loans because our model incorrectly expected them to default (FN rate). In this sense, it could be computed the VA in relative terms, comparing the savings when using a predictive model to the case of not using any model at all.

We understand that this approach, while valuable, has some limitations, as its computation relies on the assumption of working in-sample retrospectively, which means that no institution could anyhow materialize the process in the real world, as the VA is a backward looking metric. Therefore, we propose a novel approach to estimate the economic impact of applying ML in credit risk, which consists of calculating the potential savings in regulatory capital derived from using ML instead of a more traditional quantitative technique. This approach would complement the VA, and it has the advantage of being implementable after the loan has been granted, so credit institutions would be able to benefit from it immediately in real business conditions. As mentioned before, our dataset consists of consumer loans that have been already granted. These loans represent a credit risk exposure to the institution, with its corresponding cost in terms of regulatory capital. Assuming that the institution follows an IRB approach[19], we can calculate the difference in terms of regulatory capital between using a commonly used model nowadays like Lasso compared to using XGBoost, the model we found to be the most efficient in terms of predictive performance in our dataset. This measure would act as a floor or lower bound in the overall economic impact of using ML, assuming that at least any institution could benefit from reducing the capital requirements on their outstanding credit exposure, on top of which they could add the VA as estimated for instance by Khandani et al (2010), if any institution decides to implement a better predictive model on its new business strategy (out-of-sample).

5.1. Savings on regulatory capital

The pre-crisis regulatory framework provided credit institutions with a large degree of discretion in determining their capital requirements. This resulted in excessive variability in banks' capital requirements, which ultimately undermined the credibility of the risk-weighted capital framework at the peak of the global financial crisis. As stated in Bastos e Santos et al (2020), the Basel III post-crisis reforms developed by the Basel Committee sought to reduce this variability. To check this prerogative, the authors assess the degree of difference in modelled capital requirements across banks and over time. They observe that those credit institutions whose capital is closer to the minimum Tier1 ratios might be using more precise quantitative models to estimate their risk-weighted assets (RWA).[20]

In this sense, in Baena et al. (2005) it is explained how theoretically statistical models with better predictive power could yield a better outcome in terms of regulatory requirements. They showed that the Basel's risk weighted function for credit risk in the IRB approach is concave in the PD. This implies that the capital requirement for a group of assets increases as its PD increases, but each time less and less. If this holds true, a more granular classification of credit ratings should yield a lower overall capital requirement, and consequently the use of a statistical model with more predictive power could yield some

[19] Following CRE 30.42 "*For retail exposures, banks must provide their own estimates of PD, LGD and EAD*". Notwithstanding this, in our exercise we will analyze the impact of calculating the PD as the only risk factor to be estimated, using a standard value for the LGD, and leaving the EAD out of the scope of this work, as mentioned in the following Section.

[20] This result is consistent with previous findings, which in fact partly led to the Targeted Review of Internal Models (TRIM) back in 2015 from the European Central Bank (ECB), which derived lately in the IRB repair program performed by the European Banking Association (EBA), known as IRB roadmap (EBA, 2019).

savings. We are going to test this idea by performing a step-by-step computation of the capital requirements for our dataset, using both Lasso and XGBoost for estimating the PD.[21]

Before starting the exercise, we summarize the key formulas needed to compute the capital requirements. The Basel framework specifies different formulas depending on the nature of the underlying assets which represent the credit exposure[22]. Since our data consists of consumer loans, we will use the formula of capital requirement *K* for retail exposures, which is calculated as follows (**Equation 1**):

$$Capital\ requirement = K = \left[LGD \cdot N\left[\frac{G(PD)}{\sqrt{(1-R)}} + \sqrt{\frac{R}{1-R}} \cdot G(0.999) \right] - PD \cdot LGD \right]$$

Equation 1

Where *LGD* stands for Loss Given Default[23], *G* is the inverse cumulative distribution function for a standard normal random variable, *PD* is the average probability of default of the group of assets, and *R* is the correlation. The formula for the correlation *R* is given by:

$$Correlation = R = 0.03 \cdot \frac{(1-e^{-35 \cdot PD})}{(1-e^{-35})} + 0.16 \cdot \left(1 - \frac{(1-e^{-35 \cdot PD})}{(1-e^{-35})}\right)$$

Equation 2

The Basel framework suggests different correlation formulas and values depending on the nature of the assets they refer to. We will use equation (2) for the computation of the correlation in our benchmark exercise, but we will consider additionally other possibilities for illustrative purposes like *R*=0.04 (for revolving retail exposures) up to *R*=0.15 (retail residential mortgage exposures), as mentioned further in this section, in order to account for the uncertain nature[24] of the retail type exposure in our dataset. Finally, the amount of risk weighted asset (*RWA*) can be computed as follows:

$$RWA = K \cdot 12.5 \cdot EAD$$

Equation 3

[21] We will compute the K function using PD *point-in-time*, although the regulation CRD 2013/36 and CRR 575/2013 requires that the ratings represent a long term assessment of the risk of the underlying loans.

[22] These assets could be corporate, sovereign, bank or retail exposures.

[23] We assume that the bank's estimate for LGD is 0.45 as baseline scenario. This is a standard value for any senior claims on sovereigns, banks, securities firms and other financial institutions that are not secured by recognized collateral (CRE32.6). In any case, different LGD values would not affect our comparison between XGBoost and Lasso, since changes in LGD would affect their capital requirement equally (see equation 1).

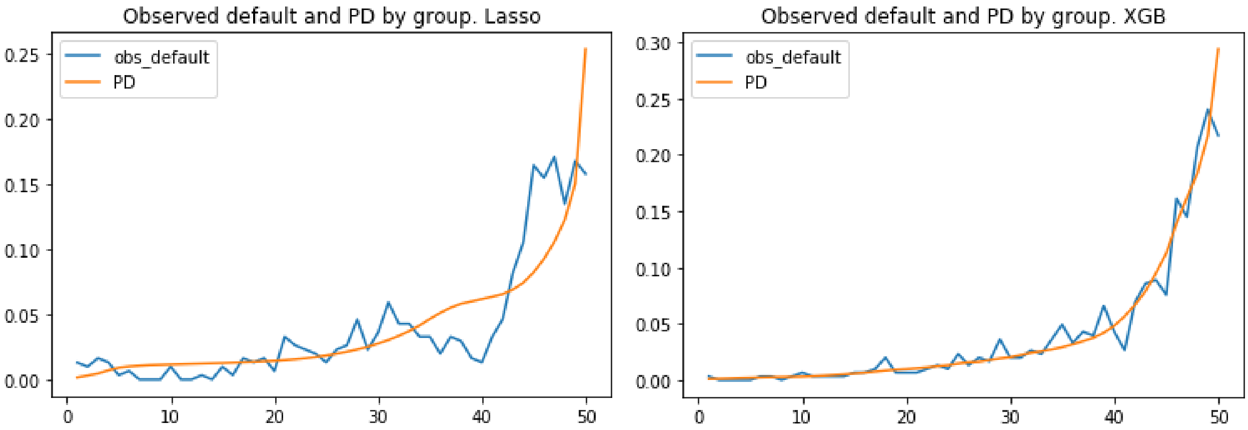
[24] We ignore certain characteristics of the underlying credit, like either if there is any guarantee or collateral or the potential revolving structure of the loans.

Where *EAD* is exposure at default measured in euros. We cannot compute this measure, as we don't know which feature corresponds to the outstanding credit balances or Exposures At Default (EAD) [25]. Therefore, we will focus on computing the savings of capital requirement *K* in relative percentage terms.

Step 1 – Discriminate between risk buckets.

Out of nearly 75,000 loans we use around 60,000 to train the models and we make predictions over the remaining 15,000 loans. We first rank those 15,000 loans by their perceived credit risk. To this purpose we estimate the PD using both Lasso and XGBoost, and we order the predictions proportionally in 50 buckets, from lower to higher values of PD[26]. The results are displayed in **Figure 9**, on the left hand side for Lasso and on the right for XGBoost. For both methods we show the average PD (orange line) and the observed default rate (blue line) for all 50 buckets. It can be seen that the estimated probability complies with the desired property of increasing monotonically in order to demonstrate discriminatory power. However, the divergence with the default rate per bucket suggests that a calibration process needs to be performed. This divergence is more significant for Lasso, especially between buckets 30 and 50. These results are in line with our findings in the calibration analysis of section 4.2, where we showed in **Figure 8a** that Lasso first tends to underestimate the default rate when PD is around 3% and 4% (which corresponds to buckets 25 to 30 of Lasso in **Figure 9**), then overestimates for PD from 5% to 7% (buckets from 32 to 42 of Lasso), underestimates again for PD from 10% to 20% (buckets from 42 to 48 of Lasso) and finally overestimates when PD is higher than 20% (buckets 49 and 50 of Lasso). While XGBoost seem to adjust better the PD to the default rate in each bucket, its fit is not perfect and therefore needs further calibration as well.

Figure 9. Ranking PDs per model



[25] As stated before, we do not know the labels of our features

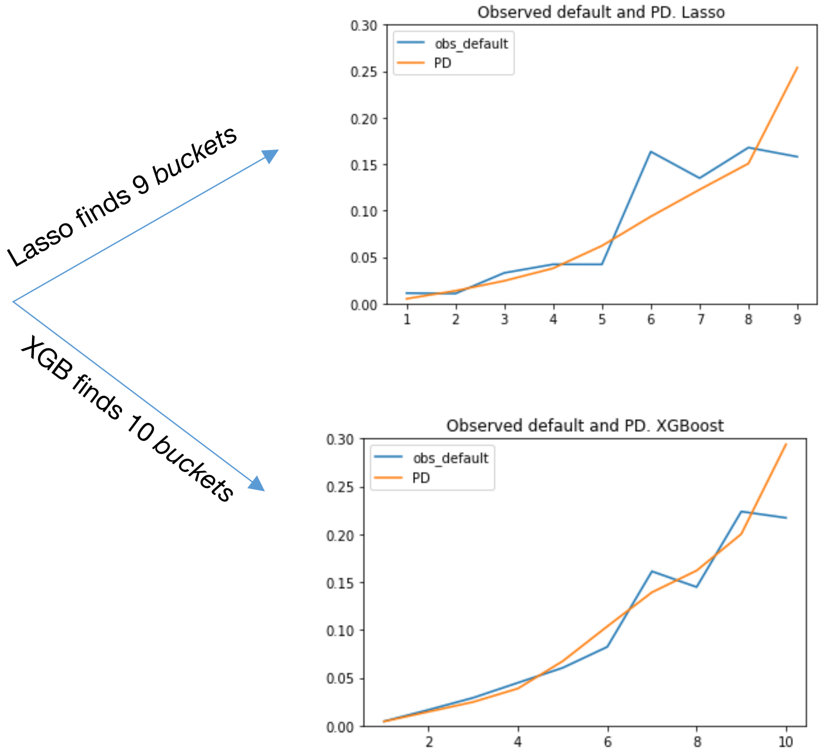
[26] There are around 300 loans per bucket.

Step 2 – Calibration process.

In order to get approval from a supervisor, the classification resulting from the model must resemble the observed default rate. We propose in **Figure 10** an initial set of 10 rating grades based on the PD estimated, in order to fine-tune the calibration.

Figure 10. Initial distribution in rating buckets.

1. Lower than 1% - **AAA**
2. From 1% to 2% - **AA**
3. From 2% to 3% - **A**
4. From 3% to 5% - **BBB**
5. From 5% to 8% - **BB**
6. From 8% to 12% - **B**
7. From 12% to 15% - **CCC**
8. From 15% to 18% - **CC**
9. From 18% to 25% - **C**
10. Higher than 25% - **D**

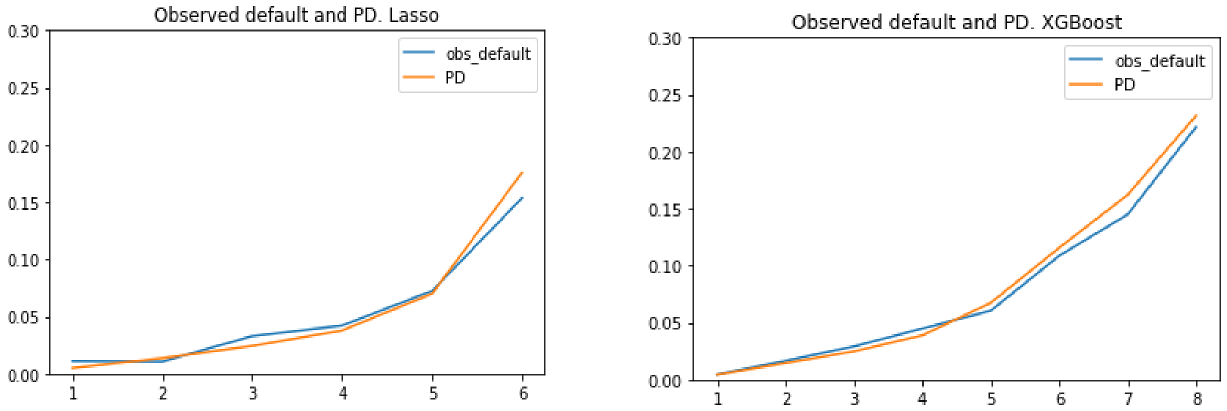


For these rating notches to be approved by the supervisor, they must comply with two criteria: (i) heterogeneity between risk buckets, and (ii) homogeneity within risk buckets. This implies that risk categories must be different from each other (in our case, finding a PD which is monotonically increasing fulfils this requirement), while keeping consistency of risk level within each group. In **Figure 10** it is evident that the homogeneity criterion does not apply, as the difference between default rate and PD in each bucket is too high[27].

In order to accomplish the two criteria, we reduce sequentially the number of buckets, until we find the first set of ratings for each model which satisfies them. The result is shown in **Figure 11**, below:

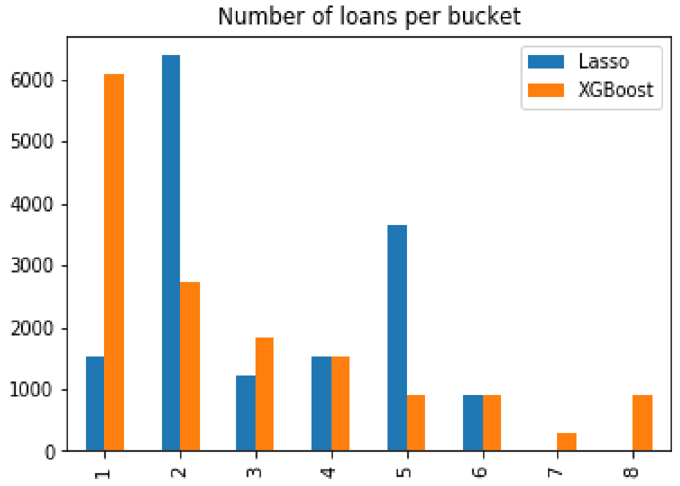
[27] We set the threshold of the homogeneity criterion in a maximum of 2% of difference between the PD and the default rate.

Figure 11. Final distribution of rating buckets.



While Lasso allows us to identify 6 different risk buckets, XGBoost allows a more granular classification, up to 8 risk buckets. As shown in **Figure 12**, the distribution of loans per bucket differ between each model. It can be seen that XGboost has a more granular and smooth distribution over buckets. XGBoost allocates a significant bigger amount of loans, in bucket 1, and then the number of loans per bucket decreases, stretching overall the distribution of loans between buckets. Lasso, on the other hand, accumulates more loans in buckets 2 and 5.

Figure 12. Distribution of loans per final rating buckets.



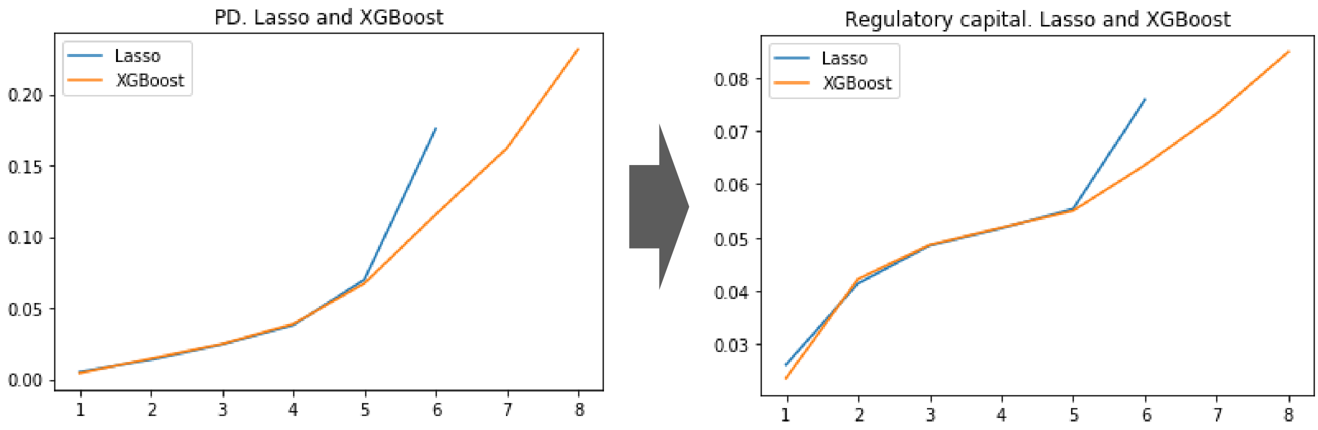
Step 3 – Calculation of capital requirements.

We can now assume that both rating scales would pass the supervisory test, allowing us to calculate the capital requirements per bucket. The capital requirement of each bucket is a function of its average PD, as it was shown in **Equation (1)**. In **Figure 13** we plot on the left hand side the average PD in each bucket for both XGBoost and Lasso, and on the right we plot the corresponding capital requirement. First of all, it can be seen that the higher the PD of a bucket, the higher would be the regulatory capital. But, as suggested by Baena et al, (2005), the relationship between the regulatory capital and the PD is concave. This can be appreciated specially for buckets 1 to 5.

Interestingly, bucket (6) in Lasso has a significantly higher PD than the bucket (6) in XGBoost. In fact, this latter model is able to achieve more granularity, as loans allocated by Lasso to a high PD (above 15%) in bucket (6), are allocated by XGBoost between bucket (6) with a PD ca. 10%, plus two additional buckets (7-8) with higher PD, ca. 15% and above 20% respectively. Since the capital requirement is a concave function of PD, the fact that XGBoost achieves a smoother stratification of buckets allows it to deliver a lower weighted capital requirement. In fact, if we take the average capital requirement for each bucket, and we weight it by the amount of loans that are in the bucket (**Figure 12**), capital requirements are 12.5% lower under the XGBoost rating scale than under the Lasso one.

We have performed our benchmark exercise under the assumption that the consumer loans of our data would fall into the category of "*other retail exposures*" according to the Basel framework. Nevertheless, as mentioned before the loans of our dataset could fall into other categories, like "retail exposure with mortgages collateral" or "revolving retail credit exposures".[28] Therefore, we have run a sensitivity analysis considering those possibilities, using their corresponding Basel formulas for the *K* function. The savings in terms of regulatory capital range are 14% and 17% respectively for those two alternative scenarios.

Figure 13. Computation of regulatory capital



Capital savings from the use of XGBoost comes from two sources. First, XGBoost has more loans with estimated probability lower than 5% than Lasso, buckets 1 to 4 (Figure 12). In those buckets, the estimated PD for both models is almost the same (Figure 13a). Second, Lasso has many more loans than XGBoost in bucket 5 and with a slightly higher PD. XGBoost allocates more loans into buckets 6, 7 and 8 than Lasso does to bucket 6, but the high PD in Lasso´s bucket 6 compensates that effect. Summarizing, the fact that XGBoost is able to deliver a more granular distribution of loans and a smother classification of PD allows it to deliver those capital savings. Ideally, we should weight the average capital requirement of each bucket by the loan balance of the bucket. Unfortunately, we do not know which feature of our dataset corresponds with the loan balance. Jiménez and Saurina (2004) pointed to an inverse relationship between the size of the loan and the probability of default because larger loans are more carefully screened. Therefore, we assume that 12.5%

[28] C.f. footnote 28.

is a conservative estimate of the savings in capital requirements, since the number of loans with default probabilities higher than 10% is higher for XGBoost than for Lasso.

6. Conclusions

While institutions have been using internal models in the context of regulatory capital for a long time, the predominant techniques have not evolved significantly. Multivariate analysis and logistic regressions, like Probit or Logit, are effective tools to predict probability of default (Trucharte et al 2015), being currently common in the industry evolutions like the Lasso penalization. However, nowadays ML tools have the potential to be a game changer, as the technological progress and financial innovation has opened the room for implementing more advanced predictive models, leveraged on big data, advanced analytics and fostered by the push of newcomers into the market, which are implementing these kind of technologies in online platforms (EBA, 2018 and Huang et al, 2020).

In this environment supervisors face the challenge of allowing credit institutions and individuals to benefit from innovation, while at the same time respecting technological neutrality and ensuring compatibility with the prudential regulation and supervisory process. In Alonso and Carbó (2020) we identified the existence of a trade-off between the predictive power of ML models, and the potential cost for supervisors to keep up with the evaluation and approval of these models at ease. With the aim to properly quantify this trade-off, we rely on the validation process of IRB systems as a tool to identify both the benefits of using ML and its potential limitations to comply with the current regulatory requirements.

In this article we partially[29] tackle the first of these challenges: the measurement of the benefits from using ML models in credit default prediction. We perform our analysis using a unique and anonymized database from a major Spanish bank. Our results show that ML models perform better than the traditional Logit model, both in classification and calibration terms. While calibration is clearly a more difficult task than classification, XGBoost and Random Forest seem to provide the best results in both measures, despite not being the most algorithmically complex models (for instance, when compared to Deep Neural Networks). In order to test the robustness of our results, we perform a sensitivity analysis, simulating how the results would change in case of different number of observations and features, demonstrating that statistically it exists a model advantage on top of an information advantage (as suggested by Huang et al, 2020).

Finally, we estimate the economic impact of being able to statistically classify and calibrate better. We propose a novel approach, in which we simulate the gains in terms of savings that an institution would achieve if they were to use XGBoost compared to a more common Lasso penalized logistic regression. We estimate that these savings could amount to up to 17% of capital requirements in our benchmark exercise, which is a significant figure that lead us to suggest that more research is needed to understand the supervisory cost to get a model approval, based on the risks embedded. As mentioned before, predictive performance comes at a price, in particular in terms of model evaluation, which should be properly quantified too in order to better inform credit institutions and supervisors on the optimal model selection. Additionally, further research is needed on how to integrate macro-

[29] In order to have a complete view of the potential economic gains of using this techniques other features should be considered, for example, the possibility of providing credit to more customers.

prudential effects of an industry wide implementation of ML models in credit risk management.

Bibliography

Albanesi, S., & Vamossy, D. F. (2019). Predicting consumer default: A deep learning approach (No. w26165). National Bureau of Economic Research.

Alonso y Carbó (2020). *Using machine learning tools in credit risk: how to measure benefits and costs from a supervisory standpoint*. Andrés Alonso, José Manuel Carbó. Banco de España. DT 2032.

Babaev, D., Savchenko, M., Tuzhilin, A., & Umerenkov, D. (2019). ET-RNN: *Applying Deep Learning to Credit Loan Applications*. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 2183-2190).

Baena et al (2005). *Aspectos críticos en la implantación y validación de modelos internos*. Raúl García Baena, Luis González Mosquera, María Oroz García. Revista de Estabilidad Financiera. Banco de España, 2005.

Butaru, F., Chen, Q., Clark, B., Das, S., Lo, A. W., & Siddique, A. (2016). *Risk and risk management in the credit card industry.* Journal of Banking & Finance, 72, 218-239.

BdF (2020). *Governance of Artificial Intelligence in Finance*. Laurent Dupont, Olivier Fliche, Su Yang, Fintech Innovation Hub ACPR. Banque de France. June 2020.

BIS (2001). *The Internal Ratings-Based Approach. Supporting Document to the New Basel Accord.* January 2001.

BIS (2005). *Studies on the validation of internal rating systems*. Working paper, 14.

BIS (2020). *Variability in Risk-Weighted Assets: What Does the Market Think?* Edson Bastos e Santos, Neil Esho, Marc Farag and Christopher Zuin. No 844.

BoE (2019). *Machine learning in UK financial services*. Bank of England

Cardoso, V. S., Guimarães, A. L., Macedo, H. F., & Lima, J. C. (2013). *Assessing corporate risk: a PD model based on credit ratings*. ACRN Journal of Finance and Risk Perspectives

Cheng y Xiang (2017). *The Study of Credit Scoring Model Based on Group Lasso.* Procedia Computer Science.

Dastile, X., Celik, T., & Potsane, M. (2020). Statistical and machine learning models in credit scoring: A systematic literature survey. Applied Soft Computing, 106263.

EBA (2017). *Report on innovative uses of consumer data by financial institutions.*

EBA (2018). *Report on the Prudential Risks and Opportunities arising for Institutions from Fintech*.

EBA (2019). "*Progress Report on the IRB Roadmap*".

EBA (2020). *Report on Big Data and Advanced Analytics*.

Fawcett (2005). *An introduction to ROC analysis*. Pattern Recognition Letters, 27.

Fernández, Ana (2019). *Inteligencia artificial en los servicios financieros*. Boletín Económico 2/2019. Artículos Analíticos. Banco de España

Fuster et al (2018). *Predictably Unequal? The Effects of Machine Learning on Credit Markets*. Andreas Fuster, Paul Goldsmith-Pinkham, Tarun Ramadorai, and Ansgar Walther.

Guegan, D., & Hassani, B. (2018). Regulatory Learning: how to supervise machine learning models? An application to credit scoring. The Journal of Finance and Data Science, 4(3), 157-171.

Huang et al (2020) *Fintech Credit Risk Assessment for SMEs: Evidence from China*. IMF Working Paper 20/193. Yiping Huang, Longmei Zhang, Zhenhua Li, Han Qiu, Tao Sun, and Xue Wang. September 2020.

IIF (2019). *Machine Learning in credit risk.* Institute of International Finance.

Jiménez G. & Saurina J. (2006). *Credit cycles, credit risk, and prudential regulation*. International Journal of Central Banking, 2 (2), 65-98.

Jones, S., Johnstone, D., & Wilson, R. (2015). *An empirical evaluation of the performance of binary classifiers in the prediction of credit ratings changes*. Journal of Banking & Finance, 56, 72-85.

Khandani, A. E., Kim, A. J., & Lo, A. W. (2010). *Consumer credit-risk models via machine-learning algorithms*. Journal of Banking & Finance, 34(11), 2767-2787.

Kvamme, H., Sellereite, N., Aas, K., & Sjursen, S. (2018). *Predicting mortgage default using convolutional neural networks*. Expert Systems with Applications, 102, 207-217.

Moscatelli, M., Narizzano, S., Parlapiano, F., & Viggiano, G. (2019). *Corporate default forecasting with machine learning* (No. 1256). Bank of Italy, Economic Research and International Relations Area.

NP (2020). *Non-paper - Innovative and trustworthy AI: two sides of the same coin.* Position paper on behalf of Denmark, Belgium, the Czech Republic, Finland, France, Estonia, Ireland, Latvia, Luxembourg, the Netherlands, Poland, Portugal, Spain and Sweden on innovative and trustworthy AI.

Petropoulos, A., Siakoulis, V., Stavroulakis, E., & Klamargias, A. (2019). *A robust machine learning approach for credit risk analysis of large loan level datasets using deep learning and extreme gradient boosting.*

Sirigniano et al (2019). *Universal features of price formation in financial markets: perspectives from deep learning.* Journal of Quantitative Finance.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321-357.

Trucharte et al (2015). *Credit Portfolios and Risk Weigthted Assets: Analysis of European Banks.* Estabilidad Financiera, 29. Banco de España. November 2015.

Turiel, J. D., & Aste, T. (2019). P2P *Loan acceptance and default prediction with Artificial Intelligence*. ArXiv preprint arXiv: 1907.01800.

WB (2019). *Credit Scoring Approaches Guidelines*. World Bank Group.

Appendix

Data balancing techniques

Since defaults represent only 3.95% of observations in our data, we perform two data balancing exercises to test the robustness of our results. These balancing techniques are among the most popular ones in the literature (Dastile 2020). First we scale the calculated loss for each observation by assigning a higher weight in the loss function on the observed defaults. The weight is computed in a way that the statistical losses associated with defaults and not defaults are balanced. The performance of the models in terms of AUC and TPR with classifier threshold 50% are in **Table 2.** We use a classifier threshold of 50% because once we balance the observations, TPR for low classifier thresholds are above 90% for every model.

Table 2: Weighted data set: AUC and True Positive Rate for different classifier thresholds

Method	AUC	TPR, Classifier threshold = 50%
Logit	78%	73%
Lasso	79%	73%
Tree	82%	74%
Random Forest	81%	66%
XGBoost	83%	75%
Deep learning	81%	72%

We can see that gains in AUC of the ML models with respect to Logit are similar to the ones we observed in the benchmark exercise, show in **Figure 2**. The main difference with this weighted dataset is that the performance of Random Forest is slightly worse than in the benchmark exercise. Regarding TPR, while XGBoost has the best performance again, Lasso and Logit having similar performance to the ML models and Random Forest obtaining a lower TPR.

Secondly, we balance our dataset by oversampling defaults with the Synthetic Minority Oversampling Technique (SMOTE). This is one of the most common methods to solve imbalance problems. It balances the class distribution by generating new examples of the minority class (for more details, see Nitesh Chawla et al 2002). We oversample in a way that we end up with a database with 25% of loans defaulted. The results are in **Table 3.**

Table 3: SMOTE oversampled dataset: AUC and True Positive Rate for different classifier thresholds

Method	AUC	TPR, Classifier threshold = 10%	TPR, Classifier threshold = 20%	TPR, Classifier threshold = 30%
Logit	79%	71%	37%	17%
Lasso	79%	71%	41%	14%

Tree	81%	68%	46%	34%
Random Forest	82%	69%	49%	31%
XGBoost	83%	65%	48%	34%
Deep learning	80%	70%	51%	40%

Again the results in terms of AUC are very similar to the ones of our benchmark exercise. The ranking of the algorithms and the difference of ML models with respect to Logit is the same. Regarding TPR, ML models outperform clearly Logit, especially for thresholds above 20%. The fact that Deep learning is the best performer in terms of TPR with this oversampled dataset stands out.

Figure 14. Simulation of AUC-ROC performance to sample size increase with 95% confidence intervals

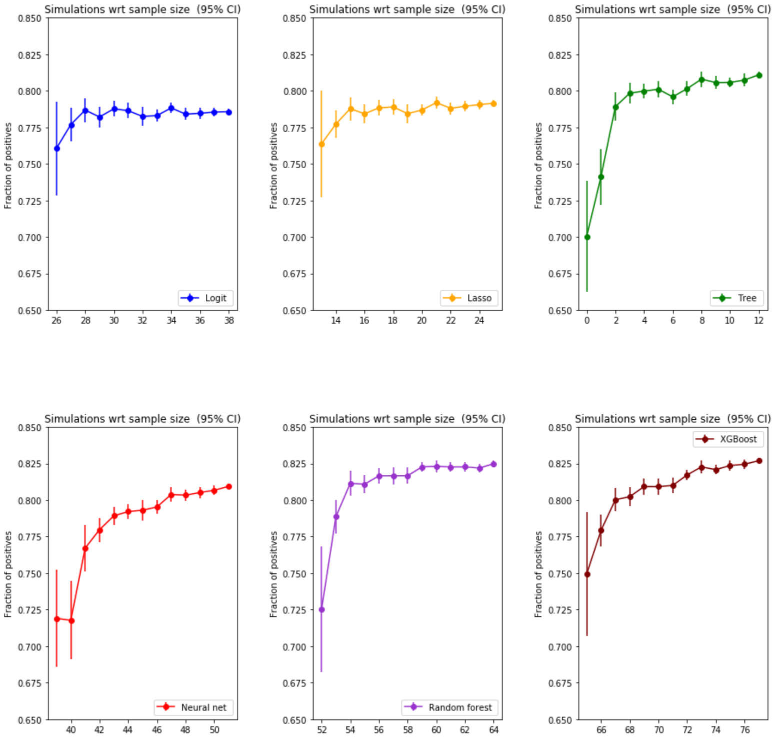


Figure 15. Simulation of AUC-ROC performance to number of features increase with 95% confidence intervals

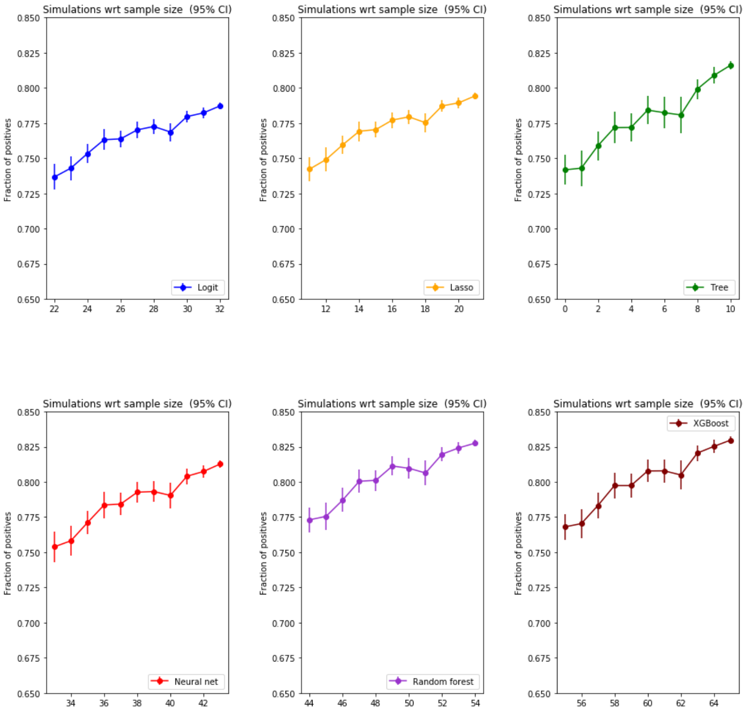


Figure 16. Calibration reliability curve with 95% confidence intervals

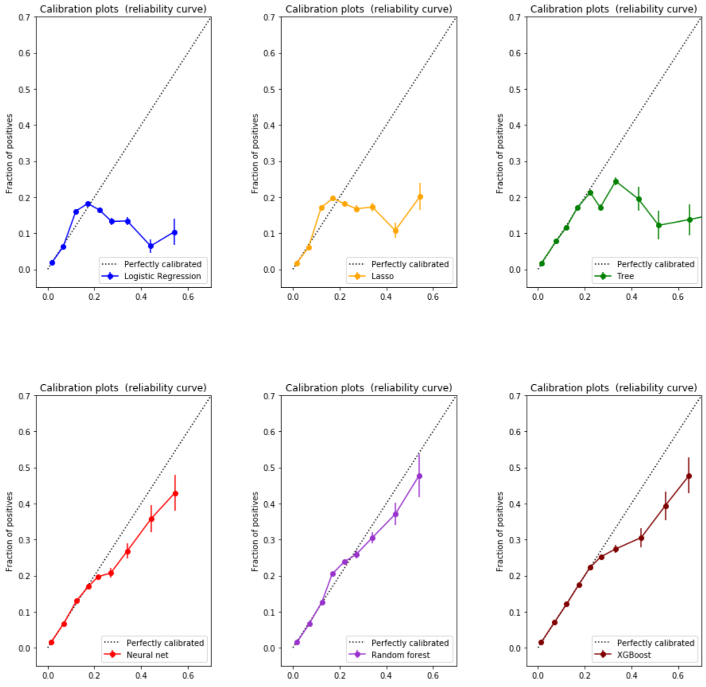
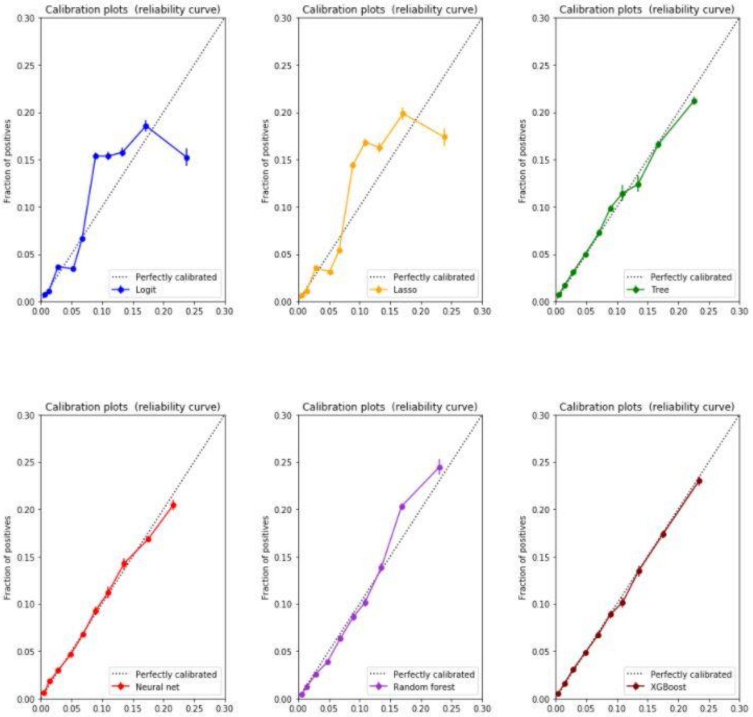


Figure 17. Calibration reliability curve with more granularity and with 95% confidence intervals


Calibration plots (reliability curve)
Fraction of positives
Perfectly calibrated
Logit
Lasso
Tree
Neural net
Random forest
XGBoost