The application of machine learning and challenger models in IRB Credit Risk modelling

The use in risk driver selection

MAKING AN IMPACT THAT MATTERS since 1845

Introduction

The recent surge in data availability and storing capacity, combined with increased computing power, creates the opportunity for machine learning (hereafter: "ML") to be applied in credit risk modelling. ML models are considered better equipped than "traditional" models to keep up with the emergence of Big Data and to detect complicated patterns and dependencies, thereby potentially leading to greater risk differentiation and higher predictive power.

The main challenge is that ML models are more complex, making their results less transparent to interpret, justify and explain to management functions and supervisors. Therefore, the incorporation of ML models in the internal ratings-based (hereafter: "IRB") model landscape has been limited.

Acknowledging that ML may play an important role in shaping credit risk modelling within financial services in the future, the European Banking Authority (hereafter: "EBA") recently published a discussion paper EBA discussion paper on machine learning for IRB models (EBA/DP/2021/04) (hereafter: "EBA ML discussion paper"). The paper aims to provide the supervisor's expectations and recommendations for a possible and prudent use of ML models in the context of the IRB framework.

As identified in the EBA ML discussion paper, one of the use cases for ML in IRB modelling are challenger models. Challenger models are models applied in parallel to traditional models, to benchmark model performance, explore alternative modelling assumptions and identify data patterns that may not be captured by traditional models.

This blog series aims to provide insights on how ML can be incorporated as challenger models in the context of IRB modelling. This blog specifically focuses on the application of ML in challenger models and the process of risk driver selection.

Traditional methods for risk driver selection

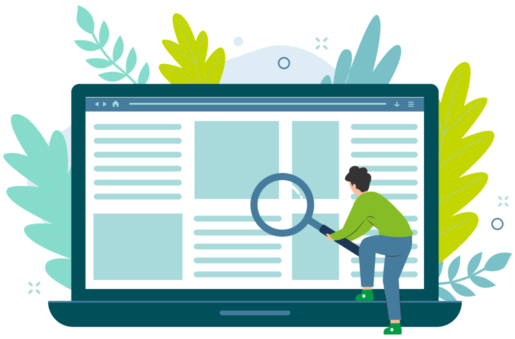
When creating a PD, LGD or EAD/CCF model, financial institutions have a longlist of potential risk drivers that may have predictive power on the output variable. The EBA guidelines on PD and LGD estimation (EBA-GL-2017-16) (paragraph 57) indicate that certain risk drivers have to be considered in this long list, such as obligor characteristics, financial information, trend information and behavioural information. The goal of risk driver selection is to reduce this longlist and include a subset of the most influential risk drivers in the model.

Market best practice is to conduct a single factor analysis (hereafter: "SFA"), also referred to as a univariate analysis, to reduce the long list to a short list of risk drivers. The SFA is conducted by studying (data) quality of individual risk drivers and the strength of its statistical relationship to the target variable. The statistical relationship is often evaluated using metrics such as correlation, chi-squared value, p-value and/or the Gini. Risk drivers are correspondingly placed in the short list if the predefined metric is above or below a specified threshold.

Using the short list of risk drivers, a multi-factor analysis (hereafter: "MFA"), also referred to as a multivariate analysis, is conducted to find the best combination of risk drivers to predict the output variable. The multi-factor analysis is often performed using the following two techniques:

- Stepwise regression
  Stepwise regression iteratively selects or removes a risk driver from the model if it yields an improvement of the regression model. Examples of stepwise regression algorithms are: forward selection, backward elimination, and bidirectional elimination.

- Lasso regression
  Lasso regression is an extension of a regression model that adds a penalty term based on the size of the risk driver coefficients when estimating the model. By doing so, risk drivers that do not have sufficient explanatory power will get a risk driver coefficient of zero. As a result, these risk drivers can be disregarded for the model.



AI based methods

The power of ML models is to detect complicated (non-linear) patterns and dependencies (interactions) between risk drivers. However, the strength of finding complex relationships comes at a cost. The procedures need to be designed in such a way that overfitting is prevented. Additionally, some ML models require so-called hyperparameter tuning. Hyperparameters are parameters which design the model structure. In contrast to the model parameters, whose values are trained from the data, the values of hyperparameters have to be determined outside of the model for which several hyperparameter tuning methods exist.

Traditional methods can potentially exclude risk drivers that exhibit a non-linear relation to the target variable or exclude risk drivers that are individually not predictive, but can be predictive when interacting with other variables. An example of an ML model that can be employed in risk driver selection are tree-based models. Post-hoc explainers are model-agnostic methods to analyze dynamics following from the model output (after training the model and therefore post-hoc). Post-hoc explainers can provide local and global explanations:

Tree based models

Tree based models use (multiple) decision trees to generate predictions. A decision tree is built up using a series of if-then rules along which observations move down a tree until a final prediction is reached. Decision tree models can be used both for regression (predicting continuous numerical values) and classification (predicting discrete "categorical" values, e.g. 0 or 1) problems.

Two of the most popular tree-based ML methods are:

- Random Forest
  Random Forest is one of the most well-known ML algorithms. It is built on the foundation that one single decision tree will not be able to fully detect complex patterns in the data and therefore is unable to produce adequate predictive power. The Random Forest is constructed by combining multiple decision trees, i.e. an ensemble of decision trees, to generate predictions. Each tree in the Random Forest is trained using a random subset of training data and a random subset of input variables.

- CatBoost
  CatBoost (categorical boosting) algorithm is an alternative to the Random Forest and has gained popularity within the ML community. Similarly as the Random Forest, the CatBoost algorithm uses multiple trees to generate predictions, but whilst constructing the trees, each tree learns from the mistakes from the previous tree.

After a tree based ML model is fitted on the dataset, feature (i.e., risk driver) importance can be extracted from the model. Feature importance provides the overall weight of each risk driver in determining the output and thus provides insights into which risk drivers influence the model prediction the most. Feature importance values are calculated based on Gini. From the feature importance values, risk drivers can be selected or disregarded for the model.

(Dis)advantages of different methods to select risk drivers

The below table gives an overview of risk driver selection methods including their advantages and disadvantages.

Traditional methods → Machine learning methods

	Sequential feature selection Sequentially add/remove features to the feature set	Penalized regression models Regression with penalty on risk driver coefficient size	Tree based methods Based around tree models, using tree feature importance to select features
Example techniques	• Sequential forward selection • Sequential backward elimination • Bi-directional selection	• Lasso • Ridge • Elastic-net	• Random forest • CatBoost • XGBoost
Advantages	• Transparent and easily explainable • No hyperparameter tuning	• Transparent and easily explainable • Automatic threshold • Relatively quick to run	• Able to capture complex (e.g., non-linear, interaction effects) data patterns • Ability to handle highly dimensional data
Disadvantages	• Ineffective in highly dimensional datasets • Do not capture non-linear relationships and interaction effects	• Do not capture non-linear relationships and interaction effects • Requires (limited) hyper parameter tuning	• Risk of overfitting • Relatively slow to run • Requires extensive hyper parameter tuning

TABLE 1. COMPARISON OF RISK DRIVER SELECTION METHODS RANGING FROM TRADITIONAL METHODS TO MACHINE LEARNING METHODS.

Challenger models for risk driver selection in practice

An advantage of using challenger models is that they provide new insights into risk drivers that need further investigation (i.e. supporting analysis, discussion with business experts).

To illustrate this, an example is provided in Table 2. From the table, it can be concluded that risk drivers A and C are selected by both methods, providing further support to include these risk drivers in the final model. For risk drivers B and D, different outcomes are observed for the models. Risk driver B is not selected by the traditional model; however it is selected by the challenger model. This could indicate that the traditional model is not able to use the full potential of the data; for example, because risk driver B is related to the target variable in a non-linear way or via interaction effects. The opposite holds for risk driver D, which is selected by the traditional method, but not by the challenger model. This could indicate that the risk driver is not as predictive as the traditional model suggests.

Risk driver	Selected by traditional selection method	Selected by challenger model
Risk driver A	✓	✓
Risk driver B	✗	✓
Risk driver C	✓	✓
Risk driver D	✓	✗

TABLE 2. ILLUSTRATIVE EXAMPLE OF RISK DRIVER SELECTION OUTCOMES BY TRADITIONAL AND CHALLENGER MODEL.

Model agnostic comparison

When using challenger models, comparability of the model outputs is crucial. For example, traditional risk driver selection processes may use metrics such as the p-value and/or Bayesian information criterion (BIC) for selecting risk drivers, while tree-based ML models may use the Gini. Comparing the outcomes of the two approaches is not straightforward since the models use different logic and metrics to select risk drivers, which creates a situation of comparing apples to pears; requiring the need for a model agnostic comparison.

The SHAP framework has been developed to explain and gain insight on the output of any type of (ML) model. An important aspect of the SHAP framework is the SHAP value. The SHAP value is the contribution of a risk driver value to the difference between the actual prediction and the mean prediction. SHAP values are computed after a model has been estimated and can be applied, or 'wrapped around', many algorithms like logistic regressions, random forest and neural networks, and are thus model agnostic.

In Figure 1, an example is shown in which SHAP values are calculated for risk drivers using a ML CatBoost algorithm and a logistic regression, which were fitted to the open source LendingClub dataset. Figure 1 shows that the models have overall comparable SHAP values for the risk drivers. However, the SHAP value for risk driver 'unemployment' differs significantly between the two methods. The logistic regression method gives higher relative importance to the 'unemployment' risk driver than the CatBoost method, as the SHAP value from the first mentioned method is higher. This difference can be reason to perform additional analysis and consult business experts to review the risk driver.

The example shows important considerations that need to be taken into account before using challenger models in practice. The following questions can be raised:

- For what purpose are challenger models used in the process?
- Does the challenger model provide more accurate results and can the outcomes of the traditional model be overruled?
- What supporting analysis is required to reach a conclusion when the challenger model has a different outcome than the traditional model?

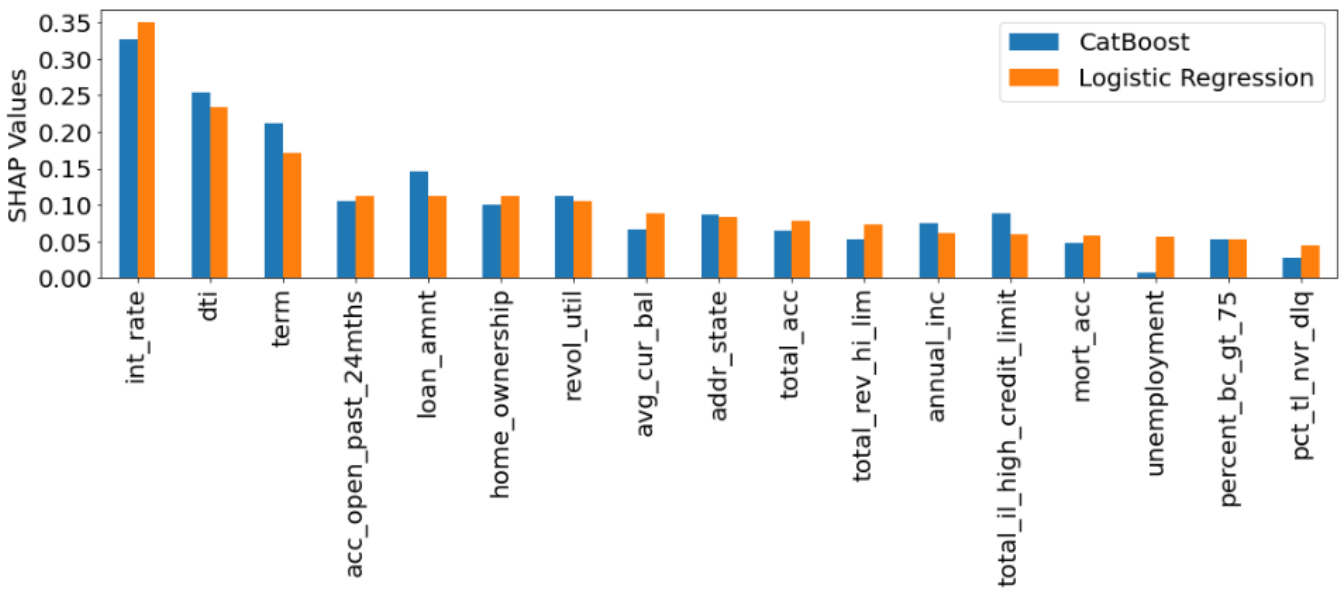


FIGURE 1. ILLUSTRATION OF SHAP VALUES FOR RISK DRIVERS BASED ON CATBOOST ALGORITHM AND LOGISTIC REGRESSION APPLIED TO THE OPEN SOURCE LENDINGCLUB DATASET

Conclusion and considerations

Challenger models can be created to challenge existing models and explore alternative approaches and assumptions. There are numerous ML models with favourable properties which could be considered in challenger models. Regulators increasingly acknowledge the added value of ML in challenger models, as can be read in the recently published EBA ML discussion paper. The advantages of ML methods over traditional methods can be found in the ability to detect complex data patterns and the ability to handle highly dimensional data. These points make ML based challenger models suitable to enhance the risk driver selection process.

However, there are important considerations that need to be taken into account before using challenger models in practice. For example, comparability of the model outputs is crucial to draw conclusions and define next steps, such as supporting analysis or discussion with business experts. The SHAP framework is a useful model agnostic tool to gain insight on model outcomes and make a fair comparison between two different selection methods.

Authors

- Marco Folpmers
- Reinout Kool
- Arwyn Goos
- Charlotte Prins
- Wouter Hottenhuis