OR Spectrum (2023) 45:251–287
https://doi.org/10.1007/s00291-022-00689-6

ORIGINAL ARTICLE

Heterogeneities among credit risk parameter distributions: the modality defines the best estimation method

Marc Gürtler[1] · **Marvin Zöllner**[1]

Received: 6 July 2021 / Accepted: 9 August 2022 / Published online: 8 September 2022

Abstract

Comparative studies investigating the estimation accuracy of statistical methods often arrive at different conclusions. Therefore, it remains unclear which method is best suited for a particular estimation task. While this problem exists in many areas of predictive analytics, it has particular relevance in the banking sector owing to regulatory requirements regarding transparency and quality of estimation methods. For the estimation of the relevant credit risk parameter loss given default (LGD), we find that the different results can be attributed to the modality type of the respective LGD distribution. Specifically, we use cluster analysis to identify heterogeneities among the LGD distributions of loan portfolios of 16 European countries with 32,851 defaulted loans. The analysis leads to three clusters, whose distributions essentially differ in their modality type. For each modality type, we empirically determine the accuracy of 20 estimation methods, including traditional regression and advanced machine learning. We show that the specific modality type is crucial for the best method. The results are not limited to the banking sector, because the present distribution type-dependent recommendation for method selection, which is based on cluster analysis, can also be applied to parameter estimation problems in all areas of predictive analytics.

Keywords Risk Management · Parameter Estimation · LGD Distributions · Machine Learning · Global Credit Data

✉ Marc Gürtler
marc.guertler@tu-bs.de

Marvin Zöllner
marvin.zoellner@tu-bs.de

[1] Department of Finance, University of Braunschweig - Institute of Technology, Abt-Jerusalem-Straße 7, Braunschweig 38106, Germany

1 Introduction

Companies collect and generate large amounts of data, which are analyzed with methods of predictive analytics and used for forecasting and estimation purposes to draw conclusions for business decisions. In particular in the context of such estimation problems, companies and data scientists have a large number of methods at their disposal, ranging from traditional linear regression to advanced methods of machine learning. Due to the large number of methods, the question immediately arises as to which method should be used for a specific estimation problem. Several studies have already dealt with this question, but they come to different conclusions.

For example, in a comprehensive study King et al. (1995) apply 16 methods, including traditional and advanced methods, to 12 real-world estimation problems in the fields of image processing, medicine, engineering, and finance. They find that there is no consistently superior method and that the respective estimation performance depend critically on the datasets used. More recently, Baumann et al. (2019) perform a detailed comparison of 14 machine learning methods, which they analyzed with regard to the estimation accuracy using 20 datasets from different fields such as life sciences, physical sciences, engineering, social sciences, economics and others. Again, it becomes apparent that the best method in each case changes depending on the dataset.

The optimal choice of estimation method is particularly relevant in the banking sector. One reason for this is a regulatory requirement as per which banks must provide their own estimates of risk parameters when using the internal ratings-based approach. In addition to the regulatory requirement, accurate predictions of risk parameters are relevant for the risk-adjusted pricing of loans. Estimation methods with high predictive accuracy offer banks a competitive advantage, whereas weak predictions can lead to adverse selection. In this study, we focus on the loss given default (LGD), which is one of three relevant parameters to estimate the risk associated with a credit product. Also with regard to LGD, several studies exist in the literature examining different methods in terms of their estimation accuracy. Because these studies show considerable differences in results, it also remains unclear for LGD estimation, which method has the highest estimation accuracy. For example, Hurlin et al. (2018) base their analysis on defaulted customers in Brazil, finding that the random forest mostly outperforms other advanced methods and that the regression tree shows low estimation accuracy. Kaposty et al. (2020) arrive at a comparable conclusion. Using a dataset of defaulted corporate leases in Germany, they find that more sophisticated methods, especially the random forest, lead to remarkable increases in the prediction accuracy. In contrast, Yao et al. (2017) examine data on U.K. bank credit cards and conclude that a combination of least squares support vector regression and ordinary least squares regression leads to the best out-of-sample estimation accuracy compared to 11 alternative (and combined) methods. Similarly, Loterman et al. (2012) compare 24 methods and find that support vector regression and an artificial neural network perform significantly better than other methods do, including

the regression tree. However, in their examination of U.S. revolving credit loans, Tobback et al. (2014) find that the forecasting accuracy of support vector regression is lower than that of a regression tree. Consistent with this finding, Bastos (2010), Qi and Zhao (2011), and Hartmann-Wendels et al. (2014) recommend using a regression tree to predict the recovery rates of Portuguese, U.S., and German loans, respectively. By using recovery rates of U.S. non-financial corporations, Altman and Kalotay (2014) show that mixture regressions provide more accurate out-of-sample estimates than other regression-based methods. The high predictive accuracy of mixture regressions is confirmed by a study of Min et al. (2020), who conduct a method comparison (including regression tree and neural network) based on recovery rates of small- and medium-sized entities in the U.S. More recently, Bellotti et al. (2021) compare different methods for the prediction of European non-performing loans and find that rule-based algorithms such as cubist regression model, boosted trees, and random forest perform significantly better than other approaches. Consequently, most studies show that advanced methods have higher estimation accuracies than traditional methods, such as the ordinary least squares regression. However, based on a dataset of U.K. defaulted credit cards, Bellotti and Crook (2012) find that the ordinary least squares regression is superior to advanced methods. Similarly, Sopitpongstorn et al. (2021) find that advanced methods such as artificial neural network and regression tree show lower predictive performance than the (local) logit regression in the prediction of loan recovery rates.

In summary, views on how well various LGD estimation methods perform are mixed. The different results can be attributed, in particular, to the different countries in which the loan portfolios are located (see, e.g., Bastos (2010)). More specifically, Grunert and Weber (2009) note that most studies focus on the U.S. banking sector but there may be national differences in bankruptcy law or the characteristics of borrowing companies. In addition, Grippa et al. (2005) and Querci (2005) observe differences in LGD characteristics across geographic regions in their investigation of Italian accounts. The results indicate that the LGD distributions in credit portfolios seem to differ between regions and countries.

It remains unclear which characteristics of an LGD distribution are responsible for the different performance results in the literature. Against this background, the present study aims to identify the distributional features relevant to the quality of LGD estimation methods, and subsequently, determine the methods that have the highest estimation accuracy for the relevant distribution types. In this way regulators, for example, obtain simple rules for LGD estimation in the banking environment without having to rely on the specific loan portfolio of a bank. This is relevant insofar as the recently introduced constraints on the use of internal ratings-based approach demonstrate that the regulators are basically striving to simplify and standardize the estimation of credit risks (see Basel Committee on Banking Supervision (2016, 2017)).

To this end, we must achieve the following objectives: consider a broad international loan portfolio; identify heterogeneities among LGD distributions, that is, the relevant characteristics and types of distributions; and compare the estimation methods individually for each LGD distribution type. To meet the first objective, we

base our analysis on an international dataset of European defaulted loans of small and medium-sized enterprises (SMEs) from 16 countries, provided by Global Credit Data. To comply with the second objective, we investigate the natures of LGD distributions using cluster analysis. Specifically, we aggregate the LGDs of all loans belonging to one country and then cluster the resulting 16 country-specific LGD distributions. The cluster analysis leads to three relevant distribution clusters, which differ mainly in their modality: a (nearly) symmetric bimodal distribution, an asymmetric (positively skewed) bimodal distribution, and a (positively skewed) unimodal distribution.[1] Finally, we apply 20 different estimation methods to the subsamples belonging to the respective distribution cluster to achieve the third objective.

The considered methods can be categorized as traditional and advanced methods. Under the traditional methods, we include ordinary least squares regression, ordinary least squares regression with backward elimination, least-angle regression, three penalized regressions (ridge, lasso, and elastic net regression), and fractional logit regression. As advanced methods, we apply six rule-based algorithms (regression tree, random forest, gradient boosting, adaptive boosting, conditional inference tree and cubist regression model), an artificial neural network, two different types of vector regression (support and relevance vector regression), a Gaussian process regression, the k-nearest neighbors method, the multivariate adaptive regression splines and a finite mixture model.

The quality of the methods undoubtedly depends on the set of variables available to explain the LGD. In this context, we use various credit characteristics, such as the EAD, the number of collaterals and guarantees, and seniority. We also incorporate macroeconomic data, including the return of the STOXX 600 Index, the return of the six-month EURIBOR and the gross domestic product (GDP) in Europe. Because the literature already provides a good understanding of the determinants of the LGD, our initial selection of relevant variables is based on Krüger and Rösch (2017).

We conduct an out-of-sample analysis for each estimation method and each LGD distribution type. In summary, for loan portfolios with a symmetric bimodal LGD distribution, the random forest offers the highest estimation accuracy. For loan portfolios with an asymmetric bimodal LGD distribution, the gradient boosting method shows the most promising results. For a unimodal LGD distribution, the finite mixture model leads to the best performance. These results clearly show that, on the one hand, the specific type of LGD distribution is crucial for the optimal choice of the estimation method. On the other hand, the results suggest that the relevant heterogeneity among LGD distributions lies in the difference in modality. These results are confirmed by a series of robustness checks.

Thus, it seems that the best method depends on the distribution type of the parameter to be estimated. This result is of interest for the whole field of predictive analytics, because a distribution type-dependent recommendation for method selection can also be applied to other parameter estimation problems.

[1] Most studies only note the bimodality of LGD distributions; see, for example Yashkir and Yashkir (2013) and Krüger and Rösch (2017).

The remainder of this article is structured as follows. Section 2 introduces the data, including the descriptive statistics, describes the estimation methods used in the comparative analysis, presents the cluster analysis, and identifies the three relevant LGD distribution types. Section 3 contains the comparative analysis for each of the three distribution types and the resulting recommendations for LGD estimation. In Section 4 several robustness checks are performed. Section 5 concludes the paper.

2 Data and LGD estimation approach

To conduct the study, we use the database of Global Credit Data[2], which contains detailed information on loan defaults of 55 banks. In this section, we first introduce the data used and provide descriptive statistics. Afterwards, we explain the theoretical background of the LGD estimation methods used in the comparative analysis. Finally, we describe the cluster analysis and identify heterogeneities among LGD distributions.

2.1 Data description

The analysis is based on a dataset of resolved defaulted loans by SMEs from 16 European countries. The estimation of the LGD is based on workout recovery rates, which are calculated as the difference between all discounted post-default incoming cash flows (F^+) and all discounted post-default costs (C^-), divided by the EAD. That is,

$$\text{LGD} = 1 - \frac{\sum F^+ - \sum C^-}{\text{EAD}} \tag{1}$$

Incoming cash flows comprise principal, interest, and post-resolution payments, the recorded book value of collateral, received fees, and commissions. Costs include legal expenses, administrator and receiver fees, liquidation expenses, and other external workout costs. All cash flows are discounted using the three-month EURIBOR of the respective default date.

In the following, we briefly describe the restrictions we apply to the raw dataset, which comprises 38,166 defaulted loans. The filter rules are based on Gürtler and Hibbeln (2013), European Banking Authority (2016), Krüger and Rösch (2017), and Betz et al. (2018).

First, we restrict the sample to all defaults since 2000 and do not include defaults after 2016. The lower time limit is selected to ensure the consistent default definition of Basel II and thus prevent biased estimation results. The upper time limit is selected for two reasons. In the subsample of recently defaulted loans (with

[2] Global Credit Data supplies the world's largest database for LGD modeling, and is internationally recognized as the standard for LGD data collection. See https://www.globalcreditdata.org/ for further information.

completed workout processes), short workout periods are obviously overrepresented. Because, in turn, loans with shorter workout periods tend to be associated with lower LGDs, this subsample can lead to a sample selection bias. In addition, workout processes of recent defaults are not necessarily completed. By limiting the time span of the dataset, we remove 2,177 observations.

Second, cures are not considered, because they do not provide default data with actual losses (see Krüger and Rösch (2017)). By excluding cures, we drop 1,147 observations.

Third, in the Global Credit Data database, the default losses range from zero (e.g., for uncalled contingent facilities) to several hundred million euros. To satisfy a materiality threshold, we remove loans with an EAD of less than €500. Using this threshold, we exclude 978 observations.

Fourth, to correct input errors and to ensure consistent and plausible data, we eliminate loans with an abnormally low or high LGD, i.e., smaller than −100% and higher than 200%, respectively. We drop 265 observations.

Finally, loans with incomplete observations are excluded. We remove 748 observations. Overall, a dataset of 32,851 loans remains.

Table 1 presents the descriptive statistics; in particular, it shows the means and quantiles of the LGDs of selected loan categories. We separate the loans by the availability of guarantees, collateral type, facility type, seniority type, and industry type. The table is a further indicator of the plausibility of the dataset. For example, the existence of guarantees or securities reduces LGDs. Conversely, non-senior and short-term loans lead to higher LGDs. Interestingly, loans from the "finance, insurance, real estate" sector have the lowest LGDs.

In addition to the loan-specific properties, we consider macroeconomics variables to improve the prediction of the LGD, as suggested in the literature.[3] Therefore, we use various macroeconomic control variables. For the overall real and financial environment in Europe, we use the return of the STOXX 600 Index. Because Qi and Zhao (2011) and Chava et al. (2011) identify the three-month treasury bill as a significant variable to consider expectations of future financial conditions in the U.S., we use the six-month EURIBOR as a significant driver for LGDs in Europe. Following Mora (2015) and Yao et al. (2015), we also use the GDP to measure the market value of all final goods and services produced in the considered period in Europe. Specifically, we consider a dummy variable that indicates whether the GDP has increased from the previous quarter. We also tested other popular macroeconomic variables, such as the 10-year euro area yield, unemployment rate, and economic sentiment index. However, they were excluded owing to their strong correlations with other macroeconomic factors and thus, lower explanatory power.

2.2 LGD estimation methods

The challenge in LGD estimation lies in providing estimated LGDs that are close to and highly correlated with the true LGDs. As such, there is a wide range of

[3] See, for example, Tobback et al. (2014) and Nazemi et al. (2017). New technical standards emphasize the importance of using economic factors; see European Banking Authority (2017).

estimation methods used in the literature. For a comprehensive analysis, all methods used in the literature must be included in the comparison. In selecting the procedures, we followed Bellotti et al. (2021) who examined 18 LGD estimation procedures. By considering two further procedures, we finally use 20 different methods, categorized as either traditional or advanced methods, as noted in Section 1. In the following, we briefly introduce the competing methods (summarized in Table 2) as well as the main references in each case.

We use linear regression as the first traditional method because it is usually used as a reference method in other LGD studies. For instance, the linear regression has been implemented in a comparative context by Loterman et al. (2012) and Krüger and Rösch (2017). From a statistical perspective, linear regression has restrictions that may make it less suitable for LGD estimation. For this reason, we include other traditional methods that address these restrictions.

First, linear regression requires exogenously identifying the best subset of the variables to include in the model. The wrong choice of variables can induce problems such as biased regression coefficients (if relevant variables are omitted) or a decrease in estimation precision (if irrelevant variables are included in the model). To overcome the difficulties in variable selection, we use ordinary least squares regression with backward elimination and least angle regression. We use backward elimination instead of simple forward selection, which has the disadvantage of neglecting variable interactions (see, e.g., Smith (2018)). The purpose of both applied methods is to build a multiple regression model that includes a parsimonious set of variables, without compromising the estimating ability of the model. The use of variable selection methods in LGD estimation is proposed, for instance, by Hartmann-Wendels et al. (2014) and Ye and Bellotti (2019).

Second, linear regression models that contain multiple variables are susceptible to overfitting and may reveal a high variance of the model estimators, which typically results in a high expected mean squared error (hereafter referred to as "estimation error"). To reduce the parameter variance, we apply penalized regressions (i.e., ridge regression, lasso regression and elastic net regression), which introduce constraints that limit the model parameters.[4] In the LGD estimation, penalized regressions are implemented by Loterman et al. (2012), among others.

Third, the values predicted by linear regression can theoretically range from minus infinity to infinity. Because LGDs are restricted by a lower limit (close to zero) and an upper limit (close to one), linear regression will not meet this restriction. To consider the LGD boundaries, we use fractional logit regression. Because we use data with plausible values out of [0, 1][5], we transform the observed LGD using the equation below (as proposed by Krüger and Rösch (2017)) before performing the fractional logit regression:

[4] Note that a lasso regression shrinks some coefficients and sets other coefficients to zero (Tibshirani, 1996). Thus, it is also a tool for variable selection.

[5] LGDs greater than one can occur, for example, owing to administrative, legal, or liquidation expenses or financial penalties. LGDs below zero can occur, for example, as a result of high collateral recoveries.

Table 1 Descriptive statistics

	Quantiles					Mean	Obs.
	0.05	0.25	0.50	0.75	0.95		
$LGD_{overall}$	−7.92	0.08	5.45	61.36	100.00	28.86	32851
log(*EAD*)	8.11	10.00	11.36	12.64	14.39	11.33	32851
Number of collaterals	0.00	0.00	1.00	1.00	3.00	1.18	32851
Number of guarantors	0.00	0.00	0.00	0.00	2.00	0.31	32851
LGD *conditional to guarantee availability:*							
No guarantee	−7.91	0.08	5.55	65.73	100.00	29.89	28774
Guarantee	−7.93	0.04	5.04	33.94	100.00	21.57	4077
LGD *conditional to collateral type:*							
No collateral	−2.62	1.88	24.09	98.22	100.00	43.37	9008
Real estate	−8.47	−0.55	2.12	26.37	99.45	18.13	7188
Other	−9.71	−0.09	4.61	51.09	100.00	25.64	16655
LGD *conditional to facility type:*							
Medium term	−3.99	0.25	4.69	48.95	100.00	26.26	19463
Short term	−15.54	−0.39	8.30	82.77	100.00	32.72	12658
Other	−6.80	0.08	4.43	79.98	100.00	31.35	730
LGD *conditional to seniority type:*							
Pari-passu	−8.47	0.01	5.64	60.03	100.00	28.63	27013
Super senior	−4.67	0.99	4.10	62.91	100.00	29.04	5301
Non-senior	−9.81	0.73	20.65	84.76	100.00	38.40	537
LGD *conditional to industry type:*							
Finance, insurance, real estate	−8.74	−0.48	1.95	45.30	100.00	23.18	4726
Agriculture, forestry, fishing, hunting	−6.17	−0.66	2.43	66.13	100.61	27.20	1486
Mining	−5.52	0.65	1.13	53.88	100.00	26.82	120
Construction	−10.42	−0.02	3.54	62.49	100.00	27.74	3720
Manufacturing	−10.29	0.01	5.37	64.12	100.00	29.35	4707
Transp., commu.,elec., gas, sani. serv.	−6.76	0.61	2.47	51.53	100.00	25.08	2587
Wholesale and retail trade	−7.82	0.08	6.28	74.09	100.00	32.42	5483
Services	−6.85	0.82	14.41	76.85	100.00	34.12	5877
Other	−5.18	0.84	10.27	45.26	100.00	26.64	4145

Note. This table presents the means and quantiles of empirical LGDs (in %) for different loan categories

$$\mathrm{LGD}_{0,1} = \frac{\mathrm{LGD} - \min(\mathrm{LGD})}{\max(\mathrm{LGD}) - \min(\mathrm{LGD})} \tag{2}$$

After the estimation, we re-transform the predicted values to the previous LGD scale. Fractional logit regressions are used for LGD estimation by Dermine and de Carvalho (2006) and Chava et al. (2011), among others.

In addition to the traditional methods, we use various advanced methods. These are assumed to have an improved estimation accuracy because they do not require a specific functional form or distribution assumptions (see Nazemi et al. (2017),

Table 2 Competing methods

Method	Exemplary literature
Ordinary Least Squares (*OLS*)	Qi and Zhao (2011), Krüger and Rösch (2017)
Variable selection methods:	
(1) OLS with Backward Elimination (*bOLS*) (2) Least Angle Regression (LAR)	Hartmann-Wendels et al. (2014), Ye and Bellotti (2019)
Penalized Regressions:	
(1) Ridge Regression (*RR*) (2) Lasso Regression (*LR*) (3) Elastic Regression (*ER*)	Loterman et al. (2012)
Fractional Logit Regression (*FLR*)	Dermine and de Carvalho (2006), Chava et al. (2011)
Regression Tree (*RT*)	Matuszyk et al. (2010), Hurlin et al. (2018)
Conditional Inference Tree (*CIT*)	Hothorn et al. (2006), Bellotti et al. (2021)
Random Forest (*RF*)	Miller and Töws (2018), Hurlin et al. (2018)
Boosting Methods:	
(1) Adaptive Boosting (*ADA*), (2) Gradient Boosting (*GB*)	Tanoue and Yamashita (2019)
Cubist Regression Model (*CUB*)	Kuhn and Quinlan (2018), Bellotti et al. (2021)
Artificial Neural Network (*ANN*)	Qi and Zhao (2011), Hurlin et al. (2018)
Support Vector Regression (*SVR*)	Yao et al. (2015, 2017), Nazemi et al. (2017)
Relevance Vector Regression (*RVR*)	Karatzoglou et al. (2004), Bellotti et al. (2021)
Gaussian Process Regression (*GAPR*)	Bellotti et al. (2021)
K-nearest Neighbors (*KNN*)	Yang and Tkachenko (2012), Hartmann-Wendels et al. (2014)
Multivariate Adaptive Regression Splines (*MARS*)	Loterman et al. (2012), Bellotti et al. (2021)
Finite Mixture Model (*FMM*)	Krüger and Rösch (2017), Min et al. (2020)

Miller and Töws (2018), and Yao et al. (2015)). At the same time, advanced methods are more prone to overfitting than the traditional methods, which may, in turn, lead to inferior estimation accuracy (see Qi and Zhao (2011)). Therefore, hyperparameter tuning and a good understanding of the methods' functioning are required when using the advanced methods. A description of the methods used in the comparative analysis is given below.

As the first advanced methods, we use various rule-based methods because they allow nonparametric representations of the relationships between the dependent and explanatory variables. The most basic method in this context is the regression tree, popularized by Breiman (1984). The method recursively splits the data into groups and uses the group averages of the dependent variable as its mean prediction. This approach has been applied to LGD estimation by, for example, Matuszyk et al. (2010) and Hurlin et al. (2018). However, regression trees often suffer from variable selection bias (see Strobl 2005), that is, predictor variables with a higher number of possible realizations (and thus, a higher number of possible cut points) have a higher probability of being chosen in the

tree-growing step. Thus, selecting variables with low importance, that is, with low information content for predicting the LGD, in this way may lead to worse trees with higher estimation errors. To overcome this limitation, we also use a conditional inference tree by Hothorn et al. (2006). This algorithm separates the variable selection process from the splitting procedure. Moreover, regression trees aim to minimize the omitted variable bias, which can be achieved by trees that are grown very deep. However, deep-grown trees tend to overfit the training data, leading to poor out-of-sample estimation accuracy. To overcome these challenges, we also use the random forest algorithm by Breiman (2001). It is a bootstrap aggregation method of de-correlated regression trees that are independently built using random subsets of variables and trained on different parts of the same training set (see, for example, Hastie et al. (2017, chapter 15). After the random forest algorithm has grown an ensemble of regression trees, an average is formed over all regression trees to establish the estimation. Because the trees in a random forest are de-correlated, the method is less prone to overfitting. By averaging across trees, the variance is also reduced, which, in general, leads to a higher estimation accuracy. The use of a random forest for LGD estimation is proposed by, for example, Bastos (2014) and Hurlin et al. (2018).

However, in addition to the above-mentioned advantages, the random forest has one decisive disadvantage. When learning imbalanced data (e.g., in a loan portfolio, where most defaulted credits have LGDs close to zero and fewer loans have LGDs close to one), there is a significant probability that the bootstrap samples contain few data of the minority class (i.e., loans with LGDs close to one). This results in biased trees that perform poorly when estimating the minority class (see Chen and Breiman (2004)). Thus, by averaging over all trees, including the biased trees, the estimation accuracy of the random forest can be reduced. Therefore, we also apply boosting-based algorithms, because they focus on incorrectly estimated samples. In a random forest, the trees are built in parallel. In boosting, the trees are built sequentially, and each tries to reduce the bias of the preceding tree. Therefore, using boosting, we build a model in a non-random way that is less susceptible to imbalanced data and makes fewer estimation errors as more trees are added.

For boosting, we use two algorithms. First, we use the adaptive boosting method of Freund and Schapie (1996), which adapts the trees by differently weighting the incorrectly and correctly estimated samples. Second, we use the gradient boosting method of Friedman (2001), which fits each new tree to the residual errors made by the previous tree. The use of boosting methods for LGD estimation is proposed by, for example, Hurlin et al. (2018) and Tanoue and Yamashita (2019). Note that the rule-based methods do not require separate and prior variable selection, because their strategies automatically rank variables by their contribution to the decrease in the estimation error. As an additional rule-based method, we use the cubist regression model by Quinlan (1993). The algorithm uses two ways for improving a modified version of regression tree prediction. First, the cubist model uses a boosting-like framework called “committees” in which iterative model trees are created in sequence to correct for estimation errors. Second, it uses a weighted average of nearest sample neighbors to adjust the predictions.

Another considered advanced method is an artificial neural network, proposed by, for example, Bishop (1995), because it can describe non-relationships in coefficients. Simply put, it is a computational model that consists of several highly interconnected processing elements that process information by their dynamic state response to external inputs. In particular, we use a multilayer perceptron, which consists of a three-layer network (an input layer, a hidden layer, and an output layer). To calculate the artificial neural network, we use a resilient backpropagation algorithm that guarantees an approximation of the estimation value through iterative model updates (see Hastie et al. (2017) for a more detailed description of artificial neural networks). Despite their above-mentioned advantages, artificial neural networks have two disadvantages: Owing to their complexity, they are prone to overfitting, and thus, require intensive hyperparameter tuning, leading to a greater computational cost. Artificial neural networks also have been used for LGD estimation by, for instance, Qi and Zhao (2011).

Additional advanced methods are two different types of vector regressions. More precisely, we consider the support vector regression introduced by Vapnik (1995), and the relevance vector regression by Tipping and Smola (2001). Both methods extend the linear regression by considering relationships that are not linear in the coefficients and are supposed to offer improved accuracy in LGD estimation. The idea of vector regressions is to map the data into a higher dimensional space using a mapping function before performing the linear regression.[6] In the comparative analysis, we choose a radial-basis function kernel for both methods to map the data into a higher dimensional space. Similar to artificial neural networks, vector regressions are prone to overfitting, and thus, need extensive computing requirements for hyperparameter tuning. Nevertheless, some studies illustrate the good predictive performance of vector regressions for LGD estimation (see Yao et al. (2015, 2017) and Nazemi et al. (2017)).

We also apply a Gaussian process regression by Williams and Rasmussen (1996), which is already proposed in the context of LGD estimation by, for example, Bellotti et al. (2021), and can be considered as a nonparametric generalization of the relevance vector regression. Instead of calculating the probability distribution of the coefficients of the regression function, the Gaussian process directly imposes a prior (Gaussian) distribution on the functional values. In the present study, we implement the Gaussian process by using a radial basis kernel.

Next, we consider the k-nearest neighbors algorithm, owing to its simplicity in dealing with nonlinear data. The algorithm uses "variable similarities" to estimate the values of any new data point. The new point is assigned a value based on the k-nearest points of a neighborhood in the Euclidean space. Because the k-nearest neighbors algorithm simply chooses the neighbors based on distance criteria, it is highly sensitive to outliers, which can lead to an inferior estimation accuracy. The k-nearest neighbors algorithm has been used for LGD estimation by, for instance, Yang and Tkachenko (2012) and Hartmann-Wendels et al. (2014).

[6] See Cheng et al. (2007) for a more detailed description of relevance vector regressions.

As an additional nonparametric method, we use multivariate adaptive regression splines by Friedman (1991), which is an extension of linear regression. In this method, the training data are first partitioned into separate piecewise linear segments (splines) with different gradients. After partitioning, the splines are connected smoothly together, resulting in a flexible model that can handle both linear and nonlinear relationships between the dependent and independent variables. The use of multivariate adaptive regression splines for LGD estimation is proposed by, for example, Loterman et al. (2012) and Bellotti et al. (2021).

Finally, we apply a finite mixture model by Leisch (2004) owing to its promising results in LGD estimation. The algorithm uses probabilistic clustering and applies an individual linear regression model for each cluster (referred to as component). The use of the finite mixture model in the LGD estimation is proposed by, for example, Krüger and Rösch (2017) and Min et al. (2020).

2.3 Clustering and LGD distribution analysis

As noted above, the main motivation for this study is based on two findings from the literature. First, existing LGD studies identify different LGD estimation methods as having the highest estimation accuracy. Second, the LGD distributions in credit portfolios seem to differ by country. This, in particular, may explain the mixed results on the accuracy of LGD estimation methods, as each study uses data of a specific country (or region), which have a specific LGD distribution.

Against this background, we identify types of LGD distributions from an international portfolio of European loans and compare the estimation quality of the methods for each type of LGD distribution. Specifically, we identify country-specific types of LGD distributions from 16 European countries and apply cluster analysis to identify relevant types of distributions. Accordingly, we build country-specific subsamples for the respective distribution clusters (in the present subsection). Subsequently, in Section 3, we compare the methods to identify the LGD estimation method with the highest accuracy for each subsample. The procedure and the results of the cluster analysis are summarized as follows.

For each country in the dataset, we aggregate the LGDs of all defaulted loans based on the LGD quantiles in a range from 1% to 100% with a stepwise increase of 1%. We then cluster the resulting 16 country-specific LGD distributions using the agglomerative hierarchical clustering of Ward (1963) and the k-means clustering of MacQueen (1967). For both approaches, the Euclidean distance was chosen as the distance measure. The final number of clusters is given if the distance between the clusters proposed by the algorithm exceeds a predefined threshold value. Thus, both approaches lead to the same results where the country-specific LGD distributions are assigned to three clusters.[7]

[7] There exists an online appendix for this article. All tables listed in this online appendix are cited accordingly below. The dendrogram of the agglomerative hierarchical clustering is shown in Fig. OA.1 of online appendix.

The results are shown in Fig. 1. By aggregating the LGDs from the countries belonging to cluster 1, we see a (nearly) symmetric bimodal LGD distribution, with two extreme events (total losses and total recoveries) being equally likely. If we aggregate the LGDs of all loans from countries belonging to the cluster 2, most of the LGDs characterize (nearly) total losses or total recoveries, also yielding a strong bimodality of the distribution. In contrast to cluster 1, total recoveries in cluster 2 are more likely than total losses (approximately in the ratio 2:1). More precisely, we can identify an asymmetric (positively skewed) bimodal distribution. Finally, if we consider the LGDs from the countries in cluster 3, we find a (positively skewed) unimodal LGD distribution that differs significantly from the LGD distributions in the other two clusters because most of the LGDs characterize total recoveries.

In summary, the cluster analysis identifies heterogeneities among the LGD distributions that differ particularly in terms of their modality. To examine the quality of the clustering result, we apply two different test statistics. Both, the paired t-test and the Mann–Whitney U test confirm the dissimilarity among the three resulting country distributions.[8] Considering the effect of insolvency legislation on LGD, the resulting distribution clusters are economically understandable. According to a study by the European Commission (2016), the time to resolve insolvency and the cost of resolving insolvency is higher in the countries of cluster 1 compared with the countries in clusters 2 and 3. Because an increase in both components (ceteris paribus) implies higher LGDs, it is understandable that the share of high LGDs is higher in countries of cluster 1 than in countries of clusters 2 and 3. This, in turn, explains the second mode ($LGD = 1$) in the first cluster.

3 Comparative analysis

In this section, we first introduce the procedure and measures used to compare the predictive performances of the LGD methods. Subsequently, we describe the procedure for determining appropriate hyperparameter values for the competing advanced methods and present the selected values for these methods in each cluster. Finally, we state and discuss the cluster-specific results of the comparative analysis.

3.1 Model comparison procedure

First, we split the dataset[9] into a subsample for training (in-sample calibration) and a subsample for testing (out-of-sample estimation), which is a common approach in LGD studies (see Gürtler and Hibbeln (2013) or Hartmann-Wendels et al. (2014)). For robustness, the results should be independent of the specific split of the dataset into training and test data. To achieve this, we split the dataset randomly[10] according to

[8] The results are shown in Table OA.1 in online appendix.

[9] In the following, the term dataset refers to a dataset of one cluster. Of course, the analysis is conducted for each cluster.

[10] A random split of a dataset is commonly used for comparing the predictive performance of LGD estimation methods and is applied both by academics and banks (see Hurlin et al. (2018)).

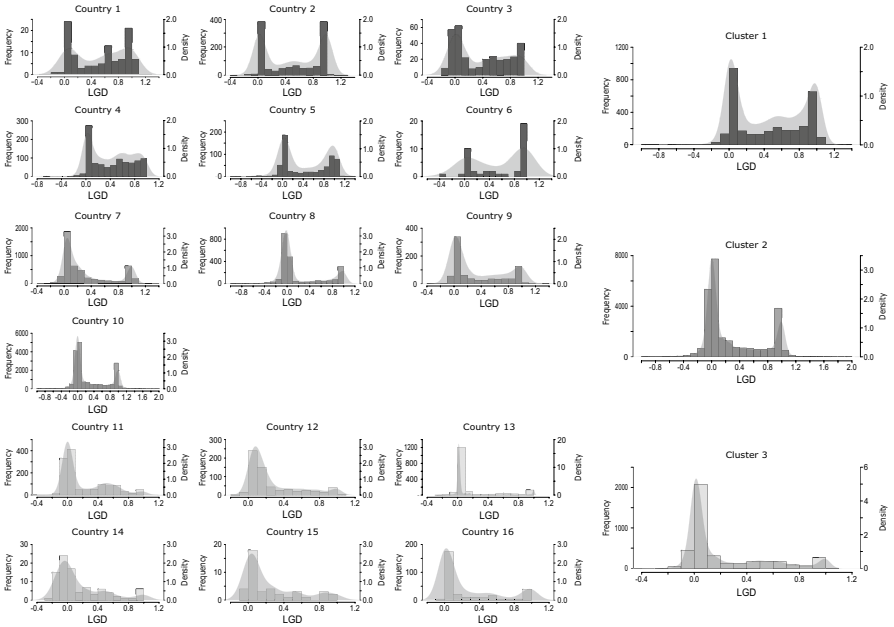

Fig. 1 European credit portfolios: LGD frequency and approximated density distributions. Note. The country names are made anonymous because of a non-disclosure agreement

different split ratios. Specifically, we split the datasets according to a (60/40), (70/30), (80/20), and (90/10) training/test split ratio. For each sample split, the LGD estimation methods are in-sample calibrated on the training dataset. The calibrated methods are applied to the test dataset to estimate out-of-sample LGDs. To measure the estimation accuracy, we use three popular out-of-sample criteria (see Hurlin et al. (2018), Krüger and Rösch (2017), or Qi and Zhao (2011)): the mean absolute error (*MAE*), the mean squared error (*MSE*), and the coefficient of determination, which are defined as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |LGD_i - \widehat{LGD}_{i,m}| \tag{3}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (LGD_i - \widehat{LGD}_{i,m})^2 \tag{4}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(LGD_i - \widehat{LGD}_{i,m})^2}{\sum_{i=1}^{n}(LGD_i - \overline{LGD})^2} \tag{5}$$

where n corresponds to the number of observations in the respective dataset; LGD_i denotes the true LGD value of the i^{th} credit, $\widehat{LGD}_{i,m}$ denotes the corresponding LGD estimation using method m, and $\overline{LGD}$ corresponds to the arithmetic mean of the

true LGD values. Finally, the mean of each criteria calculated over all sample splits denotes the estimation accuracy of the respective LGD estimation method.

3.2 Hyperparameter tuning

To provide the best out-of-sample estimation accuracy for each method, it is partially necessary to determine a suitable set of hyperparameter values, a process called hyperparameter tuning. We determine hyperparameter values for the advanced methods, as well as the penalized regressions for each cluster and each sample split, using a five-fold cross-validation (see Nazemi et al. (2017) and Hurlin et al. (2018)) and a random search algorithm. We use the random search algorithm instead of the grid search algorithm because a random search leads to significantly shorter runtimes, and delivers accurate results if the number of iterations is sufficiently high (see Bergstra and Bengio (2012)). The final choice of hyperparameter values in the tuning process is based on the same criteria used in the model comparison procedure.

The hyperparameter tuning process is based on the respective training subsamples and can be described as follows. We first separate the respective training dataset into five subsamples. Of the five subsamples, four are used for (in-sample) calibration, while the remaining set is used for out-of-sample testing. This procedure is carried out five times with a changing test dataset. Within this process, the random search algorithm trains the considered LGD estimation method based on 1,000 different hyperparameter settings, where the hyperparameters are chosen randomly from a predefined hyperparameter set. The number of hyperparameter settings follows Bergstra and Bengio (2012). Finally, the random search algorithm chooses the hyperparameter values with the highest estimation accuracy (e.g., the smallest *MSE* on the test set). For each estimation method and cluster, the chosen hyperparameters, corresponding sets, and final choice of hyperparameter values are given in Table 3. For clarity, we limit the presentation of the hyperparameter tuning results in each cluster to the (70/30) sample split (with the MSE as evaluation criterion), which is often used in other LGD studies (see Qi and Zhao (2011) and Gürtler and Hibbeln (2013)).[11] Below, we briefly summarize the main results.

First, for the lasso and elastic regressions, we find the same optimal hyperparameter values ($\lambda = 0.001$ in cluster 1 and $\lambda = 0.0001$ in clusters 2 and 3), whereas the ridge regression deviates significantly from these values $\lambda = 100$ in each cluster).

Second, for the rule-based methods, the results are as follows. For the regression tree, the trees in each cluster are similar in terms of size (6–7), while the tree in cluster 2 has a lower minimum "node size" (9 instead of 14 or 17). For the conditional inference tree and random forest, the tuning process shows that the number of splitting variables is similar in each cluster. The random forest also builds higher tree sizes than those of the regression tree in each cluster (9 instead of 6 or 7) and

[11] The chosen hyperparameter values for the other sample splits ((60/40), (80/20), and (90/10)) and performance measures are available from the authors upon request.

Table 3 Hyperparameter choice for various LGD methods in each cluster for the (70/30) sample split

Method	Hyperparameter	Description of Hyperparameter	Hyperparameter Set	Set inspired by	Hyperparameter Choice		
					Cluster 1	Cluster 2	Cluster 3
RR *LR* *ER*	λ	Lagrangian parameter that control the amount of shrinkage of the model parameters: The larger the value of λ, the greater is the amount of shrinkage.	$\{10^{-6}, 10^{-5}, ..., 10^{6}\}$	Zou and Hastie (2005)	100 0.001 0.001	100 0.0001 0.0001	100 0.0001 0.0001
RT	Tree size	Tuning parameter that controls the tree's complexity and indicates how deep the tree is allowed to be.	$\{3, 4, ..., 10\}$	Qi and Zhao (2011)	6	7	6
	Min node size	The minimum number of observations required to split an internal node. By increasing the node size, the tree becomes more constrained because it has to consider more samples at each node.	$\{5, 6, ..., 20\}$	Qi and Zhao (2011)	14	9	17
CIT	Tree size	Same description as for Regression Tree.	$\{3, 4, ..., 10\}$	Talaba (2019)	8	8	8
	Mincriterion	1 - p-value that must be exceeded in order to implement a split.	$\{0.01, 0.02, ..., 1\}$	Talaba (2019)	0.65	0.6	0.15
RF	Tree size	Same description as for Regression Tree.	$\{3, 4, ..., 10\}$	Qi and Zhao (2011)	9	8	9
	Min node size	Same description as for Regression Tree.	$\{5, 6, ..., 20\}$	Qi and Zhao (2011)	6	5	7
	# splitting variables	The number of variables to consider when looking for the best split. It handles the so-called bias-variance trade-off.	$\{1, 6, ..., 10\}$	Breiman (2001)	8	9	9
	# trees	This parameter specifies the number of trees in the forest of the model and thus controls the model's complexity.	$\{100, 101..., 1000\}$	Hastie et al. (2017)	833	873	777

Table 3 (continued)

Method	Hyperparameter	Description of Hyperparameter	Hyperparameter Set	Set inspired by	Hyperparameter Choice		
					Cluster 1	Cluster 2	Cluster 3
ADA	Tree size	Same description as for Regression Tree.	{3, 4, ..., 10}	Qi and Zhao (2011)	4	4	6
	Min node size	Same description as for Regression Tree.	{5, 6, ..., 20}	Qi and Zhao (2011)	13	12	11
	# splitting variables	Same description as for Random Forest.	{1, 6, ..., 10}	Breiman (2001)	6	6	8
	# trees	Same description as for Random Forest.	{100, 101, ..., 1000}	Hastie et al. (2017)	112	134	152
GB	Tree size	Same description as for Regression Tree.	{3, 4, ..., 10}	Qi and Zhao (2011)	5	5	4
	Min node size	Same description as for Regression Tree.	{5, 6, ..., 20}	Qi and Zhao (2011)	8	9	8
	# splitting variables	Same description as for Random Forest.	{1, 6, ..., 10}	Breiman (2001)	6	5	7
	# trees	Same description as for Random Forest.	{100, 101, ..., 1000}	Hastie et al. (2017)	123	137	168
CUB	# committees	Tuning parameter that controls the number of boosting iterations.	{1, 2, ..., 100}	Kuhn and Johnson (2016)	23	62	64
	k	The Number of nearest training set instances that are used to adjust the model prediction.	{1, 2, ..., 9}	Kuhn and Johnson (2016)	8	7	7

Table 3 (continued)

Method	Hyperparameter	Description of Hyperparameter	Hyperparameter Set	Set inspired by	Hyperparameter Choice		
					Cluster 1	Cluster 2	Cluster 3
ANN	# hidden neurons	The number of neurons in the first and second hidden layer. The parameter affects the network's complexity.	({1, 2, ..., 10}; {0, 1, ..., 10})	Heaton (2008)	(6,0)	(5,0)	(5,0)
	Activation	The function is attached to each neuron in the network, and, determines whether a neuron should be activated, based on whether its input is relevant to the model's prediction.	{logistic, tanh, relu}	Hastie et al. (2017)	Logistic	Logistic	Logistic
	Error function	The function that optimizes the weights of the network.	{stochastic gradient -based optimizer (sgo)}	Kingma and Ba (2014)	sgo	sgo	sgo
SVR	C	The so-called cost parameter controls the penalty imposed on observations that lie outside a defined margin of tolerance. Larger values of C focus attention on (correctly classified) points near the decision boundary, while smaller values involve data further away.	$\{0.001, 0.01, 0.1, 1, 5, 10, 100\}$	van Gestel et al. (2004)	1	1	1
	ϵ	The parameter controls the width of the so-called ϵ-insensitive zone, used to fit the training data. The value of ϵ can affect the number of support vectors used to construct the regression function. A bigger value of ϵ indicates that, fewer support vectors are selected.	$\{0, 0.1, ..., 1\}$	van Gestel et al. (2004)	0.2	0.3	0.4

Table 3 (continued)

Method	Hyperparameter	Description of Hyperparameter	Hyperparameter Set	Set inspired by	Hyperparameter Choice		
					Cluster 1	Cluster 2	Cluster 3
RVR	Kernel	The kernel function used in training and predicting.	{radial basis function (rbf)}	Karatzoglou et al. (2004)	rbf	rbf	rbf
	σ	Tuning parameter that determines the inverse kernel width for the radial basis function.	{0.01, 0.02, ..., 1}	Karatzoglou et al. (2004)	0.06	0.09	0.10
GAPR	Kernel	Same description as for Relevance Vector Regression.	{radial basis function (rbf)}	Karatzoglou et al. (2004)	rbf	rbf	rbf
	σ	Same description as for Relevance Vector Regression.	{0.01, 0.02, ..., 1}	Karatzoglou et al. (2004)	0.03	0.05	0.06
KNN	k	Number of nearest neighbors.	{1, 2, ..., 100}	Hastie et al. (2017)	22	20	11
MARS	Degree	Complexity parameter that controls the maximum degree of input terms in the regression function.	{1, 2, ..., 5}	Boehmke and Greenwell (2020)	2	4	3
	nprune	Number of terms to retain in the final regression function.	{2, 3, ..., 100}	Boehmke and Greenwell (2020)	18	32	15
FMM	k_c	Number of mixture components.	{1, 2, ..., 10}	Min et al. (2020)	5	5	5

Note. The names of the chosen hyperparameters and a description of the estimation methods can be found in Hastie et al. (2017)

produces a high number of trees (777–873). Compared with the random forest, both boosting methods build smaller trees in each cluster ("tree size" $\in \{4, 5, 6\}$ instead of "tree size" $= 9$) and produce distinctly fewer trees (112–168 instead of 777–873). For the boosting methods, we use a constant learning rate that corresponds to the speed with which the error is corrected from each tree to the next. A high learning rate requires a lower number of trees, and, conversely, a low learning rate requires a higher number of trees. As we test the number of trees for a predefined set, a constant learning rate is appropriate. The trees in the cubist regression model are (nearly) similar in clusters 2 and 3, while the first cluster produces a lower number of committees (23 instead of 62 and 64).

Third, for the artificial neural network, in each cluster, the second[12] hidden layer is eliminated and the number of neurons in the first hidden layer varies from five to six. In each cluster, the logistic function is preferred as the activation function. As a solver for the weight optimization, we use the stochastic gradient-based optimizer (with a learning rate of 0.001) proposed by Kingma and Ba (2014) because it is recommended for large datasets.

Fourth, in the support vector regression, the cost parameter value is the same for each cluster, while the number of selected support vectors increases from cluster 1 to cluster 3. For the relevance vector regression and the Gaussian process regression, the inverse kernel width increases from cluster 1 to 3. As a kernel function, we use the radial basis function for each of the three last-mentioned methods because of its good overall performance for vector machines (see Baesens et al. (2000)).

Fifth, in the k-nearest neighbors method, the number of nearest neighbors in cluster 3 ($k = 11$) deviates from those of clusters 1 and 2 ($k \in \{20, 22\}$).

Sixth, for the multivariate adaptive regression splines, both tuning parameters (maximum degree of input parameters and number of terms to retain in the final regression function) increase from cluster 1 to cluster 2 and decrease to cluster 3.

Finally, in the finite mixture model, the number of mixture components are identical in each cluster.

3.3 Out-of-sample results

In this subsection, we present the results of our comparative analysis. Because the cluster-specific best estimation methods are identical for all selected performance measures, we only present the results based on the MSE and MAE in detail for reasons of clarity.[13] Tables 4 and 5 show the out-of-sample estimation accuracies of the LGD methods. The resulting MSEs and MAEs are shown separately for the different clusters and split ratios. The final assessment of the methods is based on their mean MSE and mean MAE, respectively.

In cluster 1, where the LGDs are symmetrically bimodally distributed, the results can be summarized as follows. First, the traditional methods are similar in terms

[12] The number of hidden layers is inspired by Hurlin et al. (2018), who apply networks in LGD estimation with one hidden layer. We consider more than one but not more than two hidden layers because adding further layers leads to considerable long computation times.

[13] The results based on R^2 are shown in Table OA.4 in online appendix.

of their mean MSE, with the ordinary least squares regression having the lowest (0.1331) and the fractional logit regression leading to the highest mean MSE (0.1335). In terms of their mean MAEs, the results are comparable with lasso regression showing the worst performance (0.3278). Second, not all advanced methods outperform the traditional methods. In particular, the k-nearest neighbors method and the artificial neural network exhibit relatively weak performance, with a mean MSE of 0.1431 and 0.1336, respectively. Considering the mean MAE, the neural network outperforms the traditional methods, with a mean MAE of 0.3107, and also shows better performance compared to other advanced methods such as the Gaussian process regression (0.3192) and the conditional inference tree (0.3160). Again, the k-nearest neighbors method performs worst with a mean MAE of 0.3343. Furthermore, the traditional methods have the highest mean MSE and mean MAE for the 70/30 sample split, whereas the performance of the rule-based methods improves with the size of the training sample. Each model extension of the simple regression tree also shows an improvement in MSE and MAE (decrease in MSE and MAE by at least 0.0094 and 0.0107, respectively). Finally, the decisive result is that the random forest distinctly outperforms the other methods – even in each sample split – with a mean MSE of 0.1241 and a mean MAE of 0.3067. Considering the mean MSE, it is followed by the gradient boosting method (0.1305) and support vector regression (0.1322). For the mean MAE, the cubist regression model (0.3068) and the relevance vector regression (0.3095) are the next best methods.

In cluster 2, where the LGDs follow an asymmetric (positively skewed) bimodal LGD distribution, the results are slightly different. First, considering the mean MSE, all of the traditional LGD estimation methods show lower estimation accuracies than in cluster 1. On average, the mean MSE has increased by 0.0041 for the traditional methods. In contrast, for the advanced methods, the mean MSE increased by 0.0014 on average. Considering the mean MAEs, the results are different. Most of the methods (except adaptive boosting method and finite mixture model) show higher estimation accuracies than in cluster 1. Second, while the fractional logit regression performs poorly in cluster 1 for both criteria, it outperforms the other traditional methods in cluster 2. Third, most of the advanced methods (except the adaptive boosting method) outperform the traditional methods, and the performance of each method increases with the size of the training sample. Finally, the gradient boosting method shows the lowest MSEs and MAEs for each sample split and leads to the lowest mean MSE of 0.1276 and the lowest mean MAE of 0.2712. For the mean MSE, it is followed by the random forest (0.1319) and the Gaussian process regression (0.1329). For the mean MAE, the cubist regression model and the support vector regression perform second best and third best, respectively.

In cluster 3, the case of (positively skewed) unimodally distributed LGDs, the MSEs of the advanced methods have evidently been reduced by about half, and the mean MSE ranges from 0.0455 to 0.0788. In contrast, the traditional methods show a reduction in the MSEs of about one third and the mean MSE ranges from 0.0807 to 0.0840. Obviously, the MAEs show equivalent results. That is, all methods can handle unimodal distributions better than bimodal distributions. While the ordinary least squares regression proves to be the best of the traditional methods (as in cluster 1) for the MSEs, it is lasso regression for the MAEs. The artificial neural network

Table 4 European credit portfolios: Out-of-sample estimation accuracies (*MSE*)

Split	*OLS*	*bOLS*	*LAR*	*RR*	*LR*	*ER*	*FLR*	*RT*	*CIT*	*RF*	*ADA*	*GB*	*CUB*	*ANN*	*SVR*	*RVR*	*GAPR*	*KNN*	*MARS*	*FMM*
Cluster 1: (nearly) symmetric bimodal LGD distribution																				
60/40	0.1338	0.1344	0.1336	0.1349	0.1330	0.1316	0.1330	0.1350	0.1379	0.1268	0.1361	0.1321	0.1345	0.1374	0.1354	0.1372	0.1357	0.1463	0.1360	0.1350
70/30	0.1359	0.1355	0.1362	0.1332	0.1347	0.1348	0.1343	0.1353	0.1320	0.1235	0.1347	0.1320	0.1314	0.1346	0.1321	0.1363	0.1323	0.1442	0.1337	0.1327
80/20	0.1316	0.1332	0.1317	0.1329	0.1329	0.1346	0.1341	0.1333	0.1294	0.1238	0.1315	0.1308	0.1322	0.1314	0.1326	0.1311	0.1317	0.1419	0.1342	0.1320
90/10	0.1311	0.1299	0.1309	0.1324	0.1326	0.1326	0.1324	0.1302	0.1309	0.1221	0.1303	0.1270	0.1317	0.1311	0.1285	0.1282	0.1309	0.1399	0.1282	0.1314
Mean	0.1331	0.1332	0.1331	0.1334	0.1333	0.1334	0.1335	0.1335	0.1326	0.1241	0.1332	0.1305	0.1324	0.1336	0.1322	0.1332	0.1327	0.1431	0.1330	0.1328
Cluster 2: asymmetric (positively skewed) bimodal LGD distribution																				
60/40	0.1388	0.1383	0.1388	0.1382	0.1383	0.1380	0.1369	0.1357	0.1352	0.1330	0.1474	0.1287	0.1351	0.1356	0.1348	0.1358	0.1340	0.1358	0.1351	0.1354
70/30	0.1376	0.1376	0.1376	0.1376	0.1376	0.1378	0.1357	0.1344	0.1340	0.1324	0.1462	0.1282	0.1341	0.1346	0.1343	0.1345	0.1335	0.1345	0.1344	0.1349
80/20	0.1366	0.1366	0.1381	0.1373	0.1376	0.1378	0.1359	0.1339	0.1334	0.1325	0.1443	0.1273	0.1335	0.1340	0.1332	0.1343	0.1331	0.1344	0.1354	0.1334
90/10	0.1353	0.1361	0.1364	0.1360	0.1361	0.1366	0.1360	0.1331	0.1317	0.1296	0.1438	0.1260	0.1305	0.1335	0.1303	0.1337	0.1309	0.1335	0.1318	0.1316
Mean	0.1371	0.1371	0.1377	0.1373	0.1374	0.1375	0.1361	0.1343	0.1336	0.1319	0.1454	0.1276	0.1333	0.1344	0.1331	0.1346	0.1329	0.1345	0.1342	0.1338
Cluster 3: (positively skewed) unimodal LGD distribution																				
60/40	0.0828	0.0815	0.0857	0.0810	0.0817	0.0819	0.0847	0.0682	0.0658	0.0603	0.0777	0.0607	0.0642	0.0805	0.0642	0.0660	0.0644	0.0646	0.0689	0.0496
70/30	0.0813	0.0812	0.0825	0.0809	0.0807	0.0799	0.0825	0.0675	0.0649	0.0551	0.0773	0.0576	0.0599	0.0791	0.0603	0.0636	0.0627	0.0627	0.0689	0.0412
80/20	0.0795	0.0817	0.0837	0.0804	0.0801	0.0802	0.0829	0.0677	0.0646	0.0549	0.0761	0.0568	0.0591	0.0787	0.0598	0.0656	0.0636	0.0630	0.0682	0.0455
90/10	0.0790	0.0808	0.0843	0.0809	0.0803	0.0807	0.0805	0.0674	0.0647	0.0540	0.0720	0.0536	0.0570	0.0768	0.0565	0.0636	0.0624	0.0642	0.0674	0.0456
Mean	0.0807	0.0813	0.0840	0.0808	0.0807	0.0807	0.0826	0.0677	0.0650	0.0561	0.0758	0.0572	0.0600	0.0788	0.0602	0.0647	0.0633	0.0636	0.0683	0.0455

Note. This table reports the *MSE* of the out-of-sample estimation for the considered LGD methods. High values for *MSE* imply a bad fit. The methods are abbreviated as follows: Ordinary Least Squares (*OLS*); *OLS* with Backward Elimination (*bOLS*); Least Angle Regression (*LAR*); Ridge Regression (*RR*); Lasso Regression (*LR*); Elastic Regression (*ER*); Fractional Logit Regression (*FLR*); Regression Tree (*RT*); Conditional Inference Tree (*CIT*); Random Forest (*RF*); Adaptive Boosting (*ADA*); Gradient boosting (*GB*); Cubist Regression Model (*CUB*); Artificial Neural Network (*ANN*); Support Vector Regression (*SVR*); Relevance Vector Regression (*RVR*); Gaussian Processes (*GAPR*); K-nearest Neighbors (*KNN*); Multivariate Adaptive Regression Splines (*MARS*); Finite Mixture Model (*FMM*)

Table 5 European credit portfolios: Out-of-sample estimation accuracies (*MAE*)

Split	*OLS*	*bOLS*	*LAR*	*RR*	*LR*	*ER*	*FLR*	*RT*	*CIT*	*RF*	*ADA*	*GB*	*CUB*	*ANN*	*SVR*	*RVR*	*GAPR*	*KNN*	*MARS*	*FMM*
Cluster 1: (nearly) symmetric bimodal LGD distribution																				
60/40	0.3225	0.3234	0.3244	0.3217	0.3278	0.3244	0.3107	0.3178	0.3296	0.3128	0.3354	0.3134	0.3099	0.2969	0.3279	0.3209	0.3221	0.3391	0.3271	0.3206
70/30	0.3250	0.3253	0.3254	0.3237	0.3296	0.3284	0.3311	0.3247	0.3147	0.3075	0.3339	0.3133	0.3057	0.3140	0.3236	0.3106	0.3182	0.3354	0.3210	0.3419
80/20	0.3188	0.3221	0.3201	0.3208	0.3276	0.3286	0.3253	0.3156	0.3097	0.3077	0.3327	0.3115	0.3065	0.3090	0.3242	0.3052	0.3178	0.3352	0.3199	0.3205
90/10	0.3176	0.3169	0.3172	0.3196	0.3261	0.3245	0.3298	0.3115	0.3099	0.2989	0.3323	0.3129	0.3053	0.3229	0.3170	0.3014	0.3186	0.3278	0.3118	0.3197
Mean	0.3210	0.3219	0.3218	0.3214	0.3278	0.3265	0.3242	0.3174	0.3160	0.3067	0.3336	0.3128	0.3068	0.3107	0.3232	0.3095	0.3192	0.3343	0.3200	0.3257
Cluster 2: asymmetric (positively skewed) bimodal LGD distribution																				
60/40	0.2963	0.2953	0.2958	0.2953	0.2975	0.2959	0.3017	0.2844	0.2843	0.2887	0.3430	0.2731	0.2736	0.2940	0.2780	0.2826	0.2861	0.2807	0.2867	0.3470
70/30	0.2950	0.2950	0.2950	0.2950	0.2970	0.2966	0.2929	0.2817	0.2815	0.2884	0.3434	0.2745	0.2732	0.2659	0.2779	0.2811	0.2869	0.2795	0.2868	0.3467
80/20	0.2924	0.2924	0.2951	0.2940	0.2971	0.2962	0.2954	0.2806	0.2789	0.2881	0.3388	0.2697	0.2705	0.2604	0.2751	0.2796	0.2859	0.2776	0.2869	0.3450
90/10	0.2923	0.2939	0.2945	0.2939	0.2954	0.2956	0.2570	0.2785	0.2764	0.2853	0.3394	0.2676	0.2678	0.2847	0.2712	0.2794	0.2829	0.2780	0.2845	0.3411
Mean	0.2940	0.2942	0.2951	0.2946	0.2968	0.2961	0.2867	0.2813	0.2803	0.2876	0.3411	0.2712	0.2713	0.2763	0.2756	0.2807	0.2854	0.2790	0.2862	0.3450
Cluster 3: (positively skewed) unimodal LGD distribution																				
60/40	0.2130	0.2077	0.2238	0.2067	0.2091	0.2108	0.2217	0.1641	0.1679	0.1518	0.2116	0.1599	0.1603	0.2228	0.2143	0.1691	0.1679	0.1633	0.1704	0.1499
70/30	0.2123	0.2112	0.2162	0.2111	0.2091	0.2067	0.2162	0.1621	0.1661	0.1498	0.1961	0.1569	0.1521	0.1944	0.2097	0.1648	0.1669	0.1498	0.1687	0.1493
80/20	0.2047	0.2121	0.2202	0.2079	0.2045	0.2051	0.2163	0.1636	0.1675	0.1440	0.2288	0.1527	0.1528	0.1940	0.2095	0.1699	0.1706	0.1563	0.1737	0.1470
90/10	0.1997	0.2046	0.2183	0.2068	0.2022	0.2036	0.2408	0.1666	0.1763	0.1496	0.2318	0.1574	0.1557	0.2305	0.2071	0.1684	0.1684	0.1787	0.1725	0.1460
Mean	0.2074	0.2089	0.2196	0.2081	0.2062	0.2065	0.2238	0.1641	0.1694	0.1488	0.2171	0.1568	0.1552	0.2104	0.2101	0.1680	0.1685	0.1620	0.1713	0.1480

Note. This table reports the *MAE* of the out-of-sample estimation for the considered LGD methods. High values for *MAE* imply a bad fit

and the adaptive boosting method perform worst for both criteria. In accordance with clusters 1 and 2, the random forest and the gradient boosting show good performance, while their absolute difference in mean MSE is slightly less than in the other two clusters. The decisive result is that the finite mixture model unmistakably outperforms the other methods—even in each sample split—with a mean MSE of 0.0455 and a mean MAE of 0.1480. For the mean MSE, it is followed by the random forest (0.0561) and the gradient boosting method (0.0572). For the mean MAE, the random forest (0.1488) and the cubist regression model (0.1552) are the next best methods.

To exclude the possibility that some superiority may have occurred by chance, we also perform a paired t-test in each cluster to compare the mean MSE and mean MAE of the five best methods (similar to Yao et al. (2017) and Hurlin et al. (2018)).[14] The key insights of the pairwise tests are as follows. Considering cluster 1, the differences in mean MSE between the random forest and the next best methods are always negative at the 5% significance level. In contrast, the differences in the mean MAE are negative at the 10% significance level. That is, the random forest shows (marginal) significant superiority. In the context of clusters 2 and 3, the two best-performing methods (i.e., the gradient boosting method and the finite mixture model) outperform all other models significantly. Precisely, the differences between the corresponding mean MSEs and mean MAEs are always negative at the 5% significance level.

Broadly, the results indicate that the advanced methods outperform the traditional methods overall. However, the relatively weak performance of the artificial neural network shows that, even with a systematic choice of hyperparameters, overfitting remains a challenging issue when applying advanced methods to an LGD estimation. Further, the level of estimation accuracy is related to the respective LGD distribution. For bimodal distributions, all methods show considerably worse performance than for unimodal distributions. This result is understandable because the methods (mostly) correctly estimate a low LGD for a unimodal distribution based on the randomly drawn training dataset. In the case of a bimodal distribution with two different modes, the estimation is discernibly more difficult. Moreover, the type of distribution is crucial for the best-performing method. For symmetrically bimodally distributed LGDs, the random forest implies the highest estimation accuracy and the paired t-test shows its significant superiority compared with the other next best methods. For the asymmetric bimodal LGD distribution, the gradient boosting method shows the best performance, which is also significant at the 5% level. This result is understandable and can be explained as follows. A central difference between the random forest and gradient boosting method is the simultaneous or iterative construction of individual trees. Because of the high probability of small LGDs in the case of asymmetric bimodal distributions, the random forest creates a high proportion of trees that belong to low LGDs. Here, the "learning effect" of the method is missing because of its simultaneous structure. In contrast, the iterative

[14] Again, the results are shown in detail in Table OA.2 and Table OA.3 in online appendix.

approach of gradient boosting leads to better identification of high LGDs, even if high LGDs only have a small proportion. That is, this method has advantages for asymmetrical bimodal distributions. Finally, in the case of a unimodal LGD distribution, the finite mixture model shows the highest estimation accuracy, which is also significant compared to the other methods. Because of the simple nature of the distribution, the finite mixture model can easily identify suitable components for which the separate linear regressions then lead to good estimation results. Although some other methods also use partitioning strategies, the finite mixture model seems to be particularly suitable for unimodal LGD distributions.

4 Robustness checks

To investigate the robustness of the performance results, we consider four modifications of the estimation approach: First, we extend the methods by including additional explanatory variables. Second, we change the clustering procedure by clustering the LGD distributions based on a loan-specific variable rather than country. Third, we apply a logarithmic transformation to the positively skewed unimodally distributed LGDs to get a more normal-like distribution. Fourth, we change the dataset and use non-European credit portfolios that are characterized by the same three types of LGD distributions.

In each robustness check, we rerun our method comparison procedure, that is, the methods are re-calibrated, optimal hyperparameter values are re-determined and the out-of-sample model comparisons and significance tests are re-performed. Before we present the detailed results, it can already be stated at this point that the best methods remain the same for the three distribution types, regardless of the performance measure. For this reason, we show only the results based on the mean MSE for reasons of clarity.[15]

4.1 Inclusion of enterprise-specific variables

In this subsection, we test how additional explanatory variables affect the estimation results. Specifically, we include the following three enterprise-specific (logarithmic) variables: the reported sales in the 12-month period before default, the reported total assets on default, and the total amount of interest-bearing debt. Because of a non-disclosure agreement, this information is not available for all enterprises in the credit portfolio. For this reason, the robustness check is based on a reduced (but sufficiently large) dataset of 4,268 defaulted loans. The LGD distributions of the loans are still characterized by the three distribution types.[16]

Table 6 shows the results in terms of the *MSE*s. It is noteworthy that the estimation errors are reduced for each method in each cluster, that is, the newly added variables

[15] The results based on the MAE and R^2 are shown in online appendix. The results of the hyperparameter tuning processes and significance tests are available upon request.

[16] The descriptive statistics and LGD distributions are shown in Table OA.5 and Fig. OA.2 in online appendix.

seem to be important for the estimation performances of the methods. For this reason, we exemplary analyze the variable importance of the random forest for the (nearly) symmetric bimodal distribution (cluster 1). The importance score of a variable is computed as the total reduction of the node impurity brought by that variable, averaged over all trees in the random forest. In this study, the decrease in node impurity is measured based on the difference between the MSE before and after splitting on a certain variable. The higher the impurity decrease, the higher the importance score of the respective variable as it indicates a higher contribution to reducing the MSE.

Figure 2 shows the relative variable importance[17] for each variable in the random forest. It turns out that the additional three variables (represented by dark bars) have a high importance score in the random forest, confirming the relevance of including enterprise-specific variables in the LGD estimation. However, because the inclusion of additional variables usually also carries the risk of overfitting, we would like to point out that such an approach does not necessarily lead to better estimation results in other estimation tasks.

As already mentioned, the main results of this robustness check are similar to those in the preceding subsection. First, for bimodal distributions, all methods show worse performances than for the unimodal distribution. Second, for symmetrically bimodally distributed LGDs, the random forest implies the highest estimation accuracy. Third, the gradient boosting method shows the best performance for the asymmetric bimodal distribution, followed by Gaussian process regression and random forest. Fourth, in the case of a unimodal distribution, the finite mixture model turns out to be best. For all best-performing methods, the paired t-tests confirm their significant superiorities. Therefore, we confirm that the level of estimation accuracy is related to the respective distribution type.

4.2 Clustering based on loan-specific variable

A key finding of our study is that the specific modality type of a distribution is crucial for the best-performing estimation method. To rule out that the identified heterogeneities among the distributions are not caused by the approach of clustering, in this robustness check we do not cluster the distributions by country, but by a loan-specific variable. Specifically, we use the number of collaterals deposited for a loan, which has emerged in the literature as one of the most important loan-specific variables for estimating LGDs (see, for instance, Dermine and de Carvalho (2006) or Krüger and Rösch (2017)). Moreover, the analysis of the variable importance of the random forest in the previous subsection also indicates the high relevance of this variable (see Fig. 2).

The clustering strategy is same as in Subsection 2.3: For each number of collateral in the dataset,[18] we aggregate the LGDs of all defaulted loans based on the LGD quantiles in a range from 1% to 100% with a stepwise increase of 1%. We then cluster the resulting ten loan-specific LGD distributions using the agglomerative hierarchical clustering. The results are shown in Fig. 3.[19] It turns out that clustering by a

[17] The relative variable importance is calculated by dividing each variable importance score by the sum of all variable importance scores.

[18] Loans with greater than or equal to ten collaterals are grouped together because there are few loans in the dataset that exceed this number of collaterals.

[19] The dendrogram of the agglomerative hierarchical clustering is shown in Fig. OA.3 in online appendix.

Table 6 Inclusion of enterprise-specific variables: Out-of-sample estimation accuracies (*MSE*)

Split	*OLS*	*bOLS*	*LAR*	*RR*	*LR*	*ER*	*FLR*	*RT*	*CIT*	*RF*	*ADA*	*GB*	*CUB*	*ANN*	*SVR*	*RVR*	*GAPR*	*KNN*	*MARS*	*FMM*
Cluster 1: (nearly) symmetric bimodal LGD distribution																				
60/40	0.1168	0.1182	0.1168	0.1173	0.1172	0.1166	0.1150	0.1181	0.1149	0.1085	0.1153	0.1121	0.1189	0.1163	0.1129	0.1196	0.1142	0.1249	0.1152	0.1168
70/30	0.1140	0.1168	0.1158	0.1157	0.1146	0.1155	0.1148	0.1182	0.1136	0.1062	0.1149	0.1134	0.1158	0.1161	0.1121	0.1177	0.1145	0.1233	0.1144	0.1158
80/20	0.1111	0.1135	0.1143	0.1153	0.1147	0.1156	0.1146	0.1163	0.1160	0.1052	0.1136	0.1173	0.1121	0.1141	0.1136	0.1143	0.1132	0.1204	0.1144	0.1121
90/10	0.1150	0.1089	0.1103	0.1108	0.1124	0.1116	0.1138	0.1121	0.1107	0.1045	0.1131	0.1063	0.1094	0.1127	0.1117	0.1062	0.1085	0.1185	0.1127	0.1107
Mean	0.1142	0.1143	0.1143	0.1148	0.1147	0.1148	0.1145	0.1162	0.1138	0.1061	0.1142	0.1123	0.1140	0.1148	0.1126	0.1144	0.1126	0.1218	0.1142	0.1139
Cluster 2: asymmetric (positively skewed) bimodal LGD distribution																				
60/40	0.1279	0.1260	0.1258	0.1268	0.1250	0.1257	0.1242	0.1260	0.1250	0.1238	0.1246	0.1233	0.1270	0.1270	0.1251	0.1277	0.1268	0.1247	0.1254	0.1252
70/30	0.1237	0.1244	0.1242	0.1255	0.1257	0.1252	0.1233	0.1172	0.1204	0.1216	0.1235	0.1210	0.1228	0.1232	0.1257	0.1237	0.1225	0.1227	0.1169	0.1218
80/20	0.1219	0.1217	0.1243	0.1235	0.1246	0.1242	0.1234	0.1173	0.1163	0.1099	0.1236	0.1074	0.1149	0.1184	0.1115	0.1188	0.1090	0.1219	0.1175	0.1182
90/10	0.1192	0.1197	0.1216	0.1191	0.1198	0.1198	0.1177	0.1176	0.1149	0.1088	0.1235	0.1058	0.1062	0.1171	0.1077	0.1179	0.1052	0.1205	0.1148	0.1061
Mean	0.1232	0.1230	0.1240	0.1237	0.1238	0.1237	0.1221	0.1195	0.1192	0.1160	0.1238	0.1144	0.1177	0.1214	0.1175	0.1221	0.1159	0.1225	0.1186	0.1178
Cluster 3: (positively skewed) unimodal LGD distribution																				
60/40	0.0789	0.0817	0.0835	0.0809	0.0805	0.0805	0.0835	0.0694	0.0664	0.0585	0.0780	0.0589	0.0624	0.0780	0.0592	0.0671	0.0639	0.0681	0.0705	0.0485
70/30	0.0811	0.0815	0.0838	0.0832	0.0791	0.0796	0.0829	0.0679	0.0646	0.0553	0.0764	0.0565	0.0592	0.0778	0.0582	0.0652	0.0649	0.0675	0.0688	0.0472
80/20	0.0769	0.0799	0.0820	0.0769	0.0790	0.0790	0.0818	0.0653	0.0628	0.0532	0.0741	0.0546	0.0575	0.0759	0.0567	0.0646	0.0617	0.0658	0.0663	0.0437
90/10	0.0788	0.0785	0.0812	0.0788	0.0797	0.0791	0.0762	0.0630	0.0621	0.0536	0.0725	0.0525	0.0566	0.0725	0.0563	0.0634	0.0621	0.0661	0.0647	0.0437
Mean	0.0789	0.0804	0.0826	0.0800	0.0796	0.0795	0.0811	0.0664	0.0640	0.0551	0.0753	0.0556	0.0589	0.0760	0.0576	0.0651	0.0631	0.0669	0.0676	0.0458

Note. This table reports the *MSE* of the out-of-sample estimation for the considered LGD methods. High values for *MSE* imply a bad fit

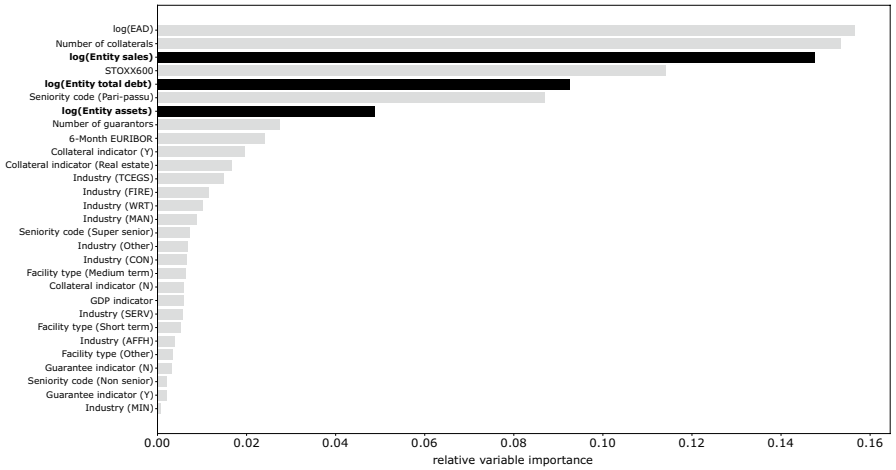

Fig. 2 Inclusion of enterprise-specific variables: Node impurity decrease measure in random forest (Cluster 1). Note. This figure shows the relative importance of the variables in the random forest using the node impurity measure. A high value indicates higher importance of a variable

loan-specific variable does not lead to any other specific distribution types. Again, we find three clusters, whose distributions essentially differ in their modality type.

Table 7 shows the out-of-sample estimation accuracies of the LGD methods. Overall, the results are comparable to those from Subsection 3.3 and can be summarized as follows: First, most of the advanced methods outperform the traditional methods for the three distribution types. Second, the performance of each method increases in each cluster with the size of the training sample. Third, all methods deal better with unimodal than with bimodal distributions, confirming that the level of estimation accuracy is related to the respective distribution. Finally, the superior methods are the same for each cluster, which confirms that the specific distribution type is crucial for the best-performing method.

This robustness check shows that the identified heterogeneities among the distributions even persist when the clustering approach is modified. Of course, it is conceivable that other distributions are relevant in a clustering approach based on other variables such as macroeconomic variables. In such a case, the best estimation procedure must be redetermined.

4.3 Logarithmic transformation of the positively skewed unimodally distributed LGDs

In this subsection, we apply a logarithmic transformation to the positively skewed unimodally distributed LGDs (country-specific LGD distribution; cluster 3), which leads to a more normal-like distribution.[20] We investigate whether this approach leads to improved estimation results or change the conclusions regarding the accuracies of the methods.

Table 8 shows the estimation performance of the methods in terms of the MSEs. The main results are similar to those for cluster 3 in Subsection 3.3. We find that the estimation errors of all methods are reduced due to the simple nature of the normal-like distribution. In accordance with the previous results, the random forest and the gradient boosting show good performance. However, the finite mixture model outperforms significantly the other methods in each sample split. Therefore, this robustness check provides two new insights: First, transforming the unimodal distribution improves the performance of the estimation methods. Second, the finite mixture model also seems to be particularly suitable for more normal-like distributions.

4.4 Non-European credit portfolios

In this subsection, we conduct a comparative analysis based on various non-European credit portfolios as a robustness check. Using 6,408 defaulted loans by Latin American, North American, and Oceanian SMEs, provided by Global Credit Data, we rerun our method comparison procedure. The restrictions we applied to the data are the same as those for the European data.[21]

The LGD distributions of the defaulted loans are shown in Fig. OA.4 in online appendix and are characterized by the distribution types identified previously. While the LGDs in the Latin American loan and the North American loan portfolios show a symmetric or asymmetric (positively skewed) bimodal distribution (clusters 1 and 2), the Oceanian LGDs are characterized by a (positively skewed) unimodal distribution shape (cluster 3). The LGD estimation methods are evaluated based on their out-of-sample performances. Table 9 shows the results in terms of the *MSE*s.[22] The main results are similar to those for the clusters in the preceding subsections.

First, for bimodal distributions, all methods show worse performances than for the unimodal distribution. Second, for symmetrically bimodally distributed LGDs, the random forest implies the highest estimation accuracy. Third, the gradient boosting method shows the best performance for the asymmetric bimodal distribution, followed by the random forest and support vector regression. Fourth, in the case of a unimodal distribution, the finite mixture model turns out to be best. Therefore, we confirm that the level of estimation accuracy is related to the respective LGD

[20] The descriptive statistics are shown in Table OA.11 in online appendix.

[21] The descriptive statistics are shown in Table OA.13 in online appendix.

[22] Again we also perform an analysis based on the R^2 and MAE. The results are available on request.

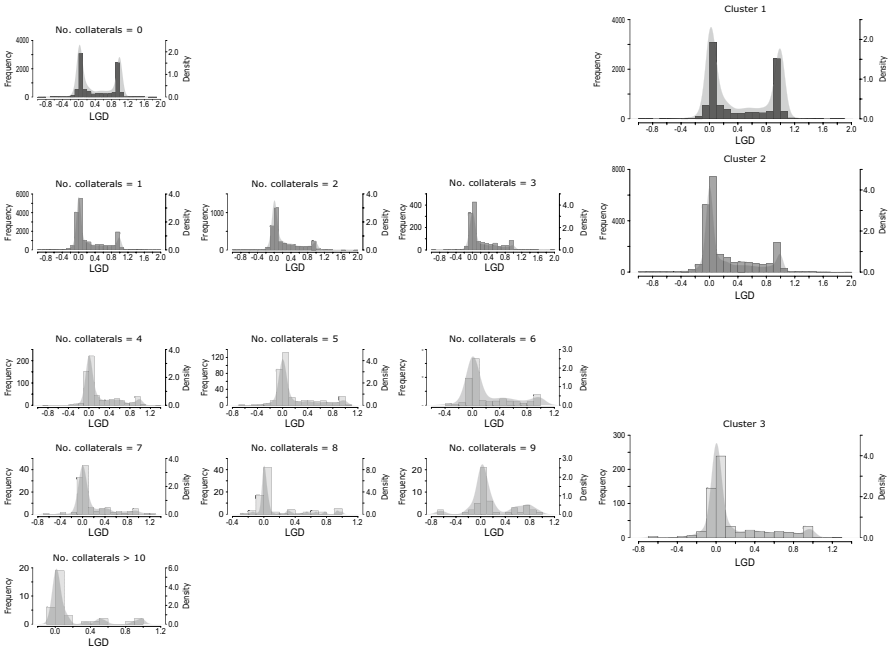

Fig. 3 Clustering based on loan-specific variable: LGD frequency and approximated density distributions

distribution type. It is also noteworthy that these results persist even with small loan portfolios such as the Latin American or the Oceanian portfolio.

5 Conclusion

The literature reveals mixed results on how well LGD estimation methods perform owing to each study using country-specific credit data that have a specific LGD distribution. In contrast, we compare various LGD estimation methods for a large class of LGD distributions. For a broad international loan portfolio, we first identify relevant types of LGD distributions by using cluster analysis and then compare the estimation methods individually for each LGD distribution type.

The cluster analysis leads to three types of distributions, which differ in their modality. We identify a (nearly) symmetric bimodal distribution, an asymmetric (positively skewed) bimodal distribution, and a (positively skewed) unimodal distribution. The estimation accuracies of 20 different methods are tested based on their out-of-sample performance, measured using MSE, MAE, and R^2. First, for loan portfolios with a symmetric bimodal LGD distribution, the random forest implies the highest estimation accuracy and should be preferred to other methods in an LGD estimation. Second, LGD estimations for loan portfolios with asymmetrically (positively skewed) bimodally distributed LGDs

Table 7 Clustering based on loan-specific variable: Out-of-sample estimation accuracies (*MSE*)

Split	*OLS*	*bOLS*	*LAR*	*RR*	*LR*	*ER*	*FLR*	*RT*	*CIT*	*RF*	*ADA*	*GB*	*CUB*	*ANN*	*SVR*	*RVR*	*GAPR*	*KNN*	*MARS*	*FMM*
Cluster 1: (nearly) symmetric bimodal LGD distribution																				
60/40	0.1366	0.1365	0.1366	0.1370	0.1372	0.1374	0.1369	0.1370	0.1357	0.1300	0.1367	0.1348	0.1353	0.1370	0.1353	0.1372	0.1362	0.1448	0.1360	0.1348
70/30	0.1362	0.1357	0.1362	0.1370	0.1369	0.1367	0.1366	0.1366	0.1356	0.1305	0.1357	0.1328	0.1357	0.1368	0.1343	0.1361	0.1359	0.1437	0.1353	0.1345
80/20	0.1355	0.1359	0.1360	0.1363	0.1362	0.1361	0.1360	0.1361	0.1347	0.1262	0.1353	0.1325	0.1340	0.1366	0.1346	0.1364	0.1359	0.1435	0.1344	0.1344
90/10	0.1357	0.1372	0.1361	0.1362	0.1363	0.1363	0.1361	0.1361	0.1354	0.1244	0.1364	0.1313	0.1328	0.1359	0.1331	0.1366	0.1357	0.1438	0.1345	0.1344
Mean	0.1360	0.1363	0.1362	0.1366	0.1367	0.1366	0.1364	0.1364	0.1353	0.1277	0.1360	0.1328	0.1345	0.1366	0.1343	0.1366	0.1359	0.1439	0.1350	0.1345
Cluster 2: asymmetric (positively skewed) bimodal LGD distribution																				
60/40	0.1254	0.1257	0.1258	0.1256	0.1257	0.1256	0.1255	0.1252	0.1235	0.1207	0.1293	0.1194	0.1241	0.1249	0.1216	0.1252	0.1222	0.1253	0.1249	0.1234
70/30	0.1253	0.1254	0.1257	0.1253	0.1253	0.1254	0.1257	0.1247	0.1232	0.1199	0.1302	0.1183	0.1242	0.1248	0.1211	0.1248	0.1225	0.1251	0.1245	0.1229
80/20	0.1252	0.1252	0.1253	0.1253	0.1255	0.1253	0.1249	0.1245	0.1220	0.1193	0.1280	0.1177	0.1239	0.1248	0.1207	0.1247	0.1210	0.1250	0.1239	0.1222
90/10	0.1251	0.1251	0.1254	0.1252	0.1253	0.1253	0.1243	0.1242	0.1213	0.1193	0.1273	0.1161	0.1234	0.1238	0.1196	0.1247	0.1193	0.1248	0.1235	0.1216
Mean	0.1252	0.1253	0.1255	0.1254	0.1254	0.1254	0.1251	0.1246	0.1225	0.1198	0.1287	0.1179	0.1239	0.1246	0.1207	0.1249	0.1213	0.1251	0.1242	0.1225
Cluster 3: (positively skewed) unimodal LGD distribution																				
60/40	0.0806	0.0802	0.0824	0.0797	0.0815	0.0815	0.0796	0.0764	0.0758	0.0708	0.0794	0.0735	0.0729	0.0785	0.0754	0.0742	0.0758	0.0754	0.0777	0.0695
70/30	0.0797	0.0790	0.0816	0.0796	0.0797	0.0798	0.0799	0.0757	0.0744	0.0702	0.0777	0.0729	0.0725	0.0787	0.0735	0.0749	0.0760	0.0751	0.0756	0.0674
80/20	0.0778	0.0780	0.0815	0.0802	0.0796	0.0795	0.0779	0.0760	0.0749	0.0695	0.0786	0.0733	0.0723	0.0787	0.0730	0.0738	0.0744	0.0751	0.0759	0.0669
90/10	0.0781	0.0783	0.0816	0.0813	0.0791	0.0800	0.0782	0.0758	0.0748	0.0677	0.0761	0.0726	0.0735	0.0765	0.0727	0.0729	0.0726	0.0756	0.0750	0.0637
Mean	0.0790	0.0788	0.0818	0.0802	0.0800	0.0802	0.0789	0.0760	0.0750	0.0695	0.0780	0.0731	0.0728	0.0781	0.0737	0.0739	0.0747	0.0753	0.0760	0.0669

Note. This table reports the *MSE* of the out-of-sample estimation for the considered LGD methods. High values for *MSE* imply a bad fit

Table 8 Logarithmic transformation of the positively skewed unimodally distributed LGDs: Out-of-sample estimation accuracies (*MSE*)

Split	*OLS*	*bOLS*	*LAR*	*RR*	*LR*	*ER*	*FLR*	*RT*	*CIT*	*RF*	*ADA*	*GB*	*CUB*	*ANN*	*SVR*	*RVR*	*GAPR*	*KNN*	*MARS*	*FMM*
60/40	0.0804	0.0809	0.0848	0.0811	0.0806	0.0817	0.0817	0.0659	0.0653	0.0556	0.0737	0.0618	0.0629	0.0806	0.0581	0.0694	0.0639	0.0761	0.0690	0.0444
70/30	0.0787	0.0776	0.0823	0.0782	0.0780	0.0776	0.0810	0.0652	0.0652	0.0553	0.0742	0.0617	0.0609	0.0768	0.0564	0.0655	0.0628	0.0751	0.0675	0.0434
80/20	0.0775	0.0789	0.0826	0.0779	0.0776	0.0776	0.0813	0.0651	0.0639	0.0549	0.0722	0.0597	0.0627	0.0760	0.0555	0.0655	0.0613	0.0746	0.0675	0.0441
90/10	0.0767	0.0800	0.0824	0.0772	0.0780	0.0775	0.0817	0.0652	0.0622	0.0531	0.0703	0.0618	0.0607	0.0757	0.0542	0.0656	0.0614	0.0698	0.0640	0.0444
Mean	0.0783	0.0793	0.0830	0.0786	0.0786	0.0786	0.0814	0.0653	0.0641	0.0547	0.0726	0.0612	0.0618	0.0773	0.0561	0.0665	0.0623	0.0739	0.0670	0.0441

Note. This table reports the *MSE* of the out-of-sample estimation for the considered LGD methods. High values for *MSE* imply a bad fit

Table 9 Non-European credit portfolios: Out-of-sample estimation accuracies (*MSE*)

Split	*OLS*	*bOLS*	*LAR*	*RR*	*LR*	*ER*	*FLR*	*RT*	*CIT*	*RF*	*ADA*	*GB*	*CUB*	*ANN*	*SVR*	*RVR*	*GAPR*	*KNN*	*MARS*	*FMM*
Latin American credit portfolio: (nearly) symmetric bimodal LGD distribution																				
60/40	0.1403	0.1465	0.1391	0.1390	0.1476	0.1478	0.1383	0.1458	0.1414	0.1256	0.1429	0.1491	0.1472	0.1448	0.1429	0.1421	0.1258	0.1444	0.1461	0.1338
70/30	0.1388	0.1467	0.1388	0.1371	0.1468	0.1458	0.1367	0.1421	0.1395	0.1110	0.1395	0.1126	0.1112	0.1434	0.1415	0.1384	0.1035	0.1433	0.1459	0.1298
80/20	0.1344	0.1449	0.1366	0.1374	0.1444	0.1450	0.1361	0.1440	0.1374	0.1072	0.1375	0.1196	0.1183	0.1409	0.1413	0.1368	0.1331	0.1443	0.1449	0.1267
90/10	0.1353	0.1438	0.1378	0.1303	0.1434	0.1443	0.1351	0.1420	0.1260	0.1081	0.1378	0.1123	0.1106	0.1398	0.1362	0.1263	0.1082	0.1373	0.1406	0.1098
Mean	0.1372	0.1455	0.1381	0.1359	0.1455	0.1457	0.1366	0.1435	0.1361	0.1130	0.1394	0.1234	0.1218	0.1422	0.1405	0.1359	0.1176	0.1423	0.1444	0.1250
North American credit portfolio: asymmetric (positively skewed) bimodal LGD distribution																				
60/40	0.1363	0.1367	0.1363	0.1363	0.1402	0.1381	0.1334	0.1391	0.1390	0.1265	0.1438	0.1272	0.1334	0.1381	0.1346	0.1384	0.1313	0.1387	0.1371	0.1342
70/30	0.1361	0.1362	0.1361	0.1361	0.1395	0.1377	0.1331	0.1387	0.1358	0.1251	0.1429	0.1249	0.1287	0.1378	0.1339	0.1379	0.1286	0.1382	0.1305	0.1337
80/20	0.1344	0.1336	0.1345	0.1344	0.1388	0.1371	0.1313	0.1351	0.1362	0.1228	0.1411	0.1179	0.1241	0.1348	0.1334	0.1359	0.1264	0.1380	0.1284	0.1268
90/10	0.1289	0.1284	0.1289	0.1289	0.1348	0.1365	0.1258	0.1332	0.1290	0.1198	0.1393	0.1185	0.1262	0.1351	0.1289	0.1351	0.1253	0.1342	0.1250	0.1248
Mean	0.1339	0.1337	0.1340	0.1339	0.1383	0.1373	0.1309	0.1365	0.1350	0.1236	0.1418	0.1221	0.1281	0.1364	0.1327	0.1368	0.1279	0.1373	0.1303	0.1299
Oceanian credit portfolio: (positively skewed) unimodal LGD distribution																				
60/40	0.0992	0.1019	0.1244	0.0994	0.0943	0.0927	0.1010	0.0951	0.1074	0.0764	0.0974	0.0893	0.0958	0.1062	0.0936	0.1094	0.0847	0.1263	0.1080	0.0651
70/30	0.1065	0.0951	0.1236	0.0951	0.0922	0.0933	0.0971	0.0918	0.1065	0.0801	0.0943	0.0888	0.0922	0.1058	0.0923	0.1077	0.0828	0.1254	0.1083	0.0560
80/20	0.1187	0.1007	0.1216	0.0977	0.0916	0.0922	0.1040	0.0919	0.1075	0.0748	0.0941	0.0885	0.0937	0.1054	0.0914	0.1016	0.0808	0.1262	0.1075	0.0514
90/10	0.1117	0.0964	0.1182	0.0959	0.0908	0.0914	0.0988	0.0904	0.1071	0.0764	0.0972	0.0882	0.0922	0.1014	0.0915	0.1023	0.0727	0.1241	0.1030	0.0410
Mean	0.1090	0.0985	0.1220	0.0970	0.0922	0.0924	0.1002	0.0923	0.1071	0.0769	0.0957	0.0887	0.0935	0.1047	0.0922	0.1052	0.0803	0.1255	0.1067	0.0534

Note. This table reports the *MSE* of the out-of-sample estimation for the Latin American, North American, and Oceanian credit portfolios. High values for *MSE* imply a bad fit

should be based on the gradient boosting method. Finally, in the case of a unimodal LGD distribution, the finite mixture model shows the best performance. The latter results are supported by a series of robustness checks.

This study makes two main contributions to the literature on LGD estimation. On the one hand, we show that different country-specific LGD distributions can be traced to three basic (modality) types, which determine the estimation method to be used. On the other hand, we identify methods that perform best, depending on the modality of the LGD distribution. These results provide general advice for banking practice and regulatory authorities. Instead of an extensive loan portfolio analysis, we recommend that only the LGD distribution type needs to be identified to select the best-performing estimation method.

Furthermore, our study also has relevance for forecasting and estimation problems outside the banking area, because the idea of clustering and identifying different parameter distribution types to determine the respective best estimation procedure is applicable in all areas of predictive analytics. In this way, we obtain a distribution-type-dependent recommendation for method selection. Of course, in case of additional identified distribution types, the performance measurement of the estimation procedures has to be repeated.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00291-022-00689-6.

Funding Open Access funding enabled and organized by Projekt DEAL. No funds, grants, or other financial support was received.

Data Availability Non-disclosure agreement with Global Credit Data (GCD). GCD is a non-profit association. See https://www.globalcreditdata.org/ for further information.

Code Availability The programming code in Python of the applied traditional and advanced methods is available.

Declarations

Conflicts of interest/Competing interests The authors have no conflicts of interest to declare that are relevant to the content of this article.

References

Altman EI, Kalotay E (2014) Ultimate recovery mixtures. J Bank Financ 40:116–129. https://doi.org/10.1016/j.jbankfin.2013.11.021

Baesens B, Viaene S, van Gestel T, et al. (2000) An empirical assessment of kernel type performance for least squares support vector machine classifiers. In: KES'2000 fourth international conference on knowledge-based intelligent engineering systems and allied technologies. Proceedings (Cat. No.00TH8516). IEEE, pp 313–316. https://doi.org/10.1109/KES.2000.885819

Basel Committee on Banking Supervision (2016) Reducing variation in credit risk-weighted assets – constraints on the use of internal model approaches. Consultative document. https://www.bis.org/bcbs/publ/d362.pdf

Basel Committee on Banking Supervision (2017) Basel iii: finalising post-crisis reforms. https://www.bis.org/bcbs/publ/d424.pdf

Bastos JA (2010) Forecasting bank loans loss-given-default. J Bank Financ 34(10):2510–2517. https://doi.org/10.1016/j.jbankfin.2010.04.011

Bastos JA (2014) Ensemble predictions of recovery rates. J Bank Financ Ser Res 46(2):177–193. https://doi.org/10.1007/s10693-013-0165-3

Baumann P, Hochbaum DS, Yang YT (2019) A comparative study of the leading machine learning techniques and two new optimization algorithms. Eur J Oper Res 272(3):1041–1057. https://doi.org/10.1016/j.ejor.2018.07.009

Bellotti A, Brigo D, Gambetti P et al. (2021) Forecasting recovery rates on non-performing loans with machine learning. Int J Forecast 37(1):428–444. https://doi.org/10.1016/j.ijforecast.2020.06.009

Bellotti T, Crook J (2012) Loss given default models incorporating macroeconomic variables for credit cards. Int J Forecast 28(1):171–182. https://doi.org/10.1016/j.ijforecast.2010.08.005

Bergstra J, Bengio Y (2012) Random search for hyper-parameter optimization. J Mach Learn Res 13(10):281–305

Betz J, Kellner R, Rösch D (2018) Systematic effects among loss given defaults and their implications on downturn estimation. Eur J Oper Res 271(3):1113–1144. https://doi.org/10.1016/j.ejor.2018.05.059

Bishop CM (1995) Neural networks for pattern recognition. Oxford University Press and Clarendon Press, Oxford

Boehmke BC, Greenwell B (2020) Hands-on machine learning with R. Chapman Hall/CRC the R series, New York

Breiman L (1984) Classification and regression trees. Chapman Hall/CRC, New York

Breiman L (2001) Random forests. Mach Learn 45(1):5–32. https://doi.org/10.1023/A:1010933404324

Chava S, Stefanescu C, Turnbull S (2011) Modeling the loss distribution. Manag Sci 57(7):1267–1287

Chen C, Breiman L (2004) Using random forest to learn imbalanced data. University of California, Berkeley, pp 1–12

Cheng H, Chen H, Jiang G et al. (2007) Nonlinear feature selection by relevance feature vector machine. In: Perner P (ed) Machine learning and data mining in pattern recognition. Springer, Berlin, Heidelberg, pp 144–159

Dermine J, de Carvalho CN (2006) Bank loan losses-given-default: a case study. J Bank Financ 30(4):1219–1243. https://doi.org/10.1016/j.jbankfin.2005.05.005

European Banking Authority (2016) Guidelines on PD estimation. LGD estimation and the treatment of defaulted exposures, Consultation Paper

European Banking Authority (2017) Impact assessment for the GLs on PD, LGD and the treatment of defaulted exposures based on the IRB survey results. EBA report on IRB modelling practices

European Commission (2016) Internal market, industry, entrepreneurship and SMEs. SBA fact sheets

Freund Y, Schapie RE (1996) Experiments with a new boosting algorithm. In: International conference on machine learning pp 148–156

Friedman JH (1991) Multivariate adaptive regression splines. Ann Stat 19(1):1–67

Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 29(5):1189–1232. https://doi.org/10.1214/aos/1013203451

Grippa P, Iannotti S, Leandri F (2005) Recovery rates in the banking industry: stylised facts emerging from the Italian experience. Next Chall Credit Risk Manag Recovery Risk 121141:121–141

Grunert J, Weber M (2009) Recovery rates of commercial lending: empirical evidence for german companies. J Bank Financ 33(3):505–513. https://doi.org/10.1016/j.jbankfin.2008.09.002

Gürtler M, Hibbeln M (2013) Improvements in loss given default forecasts for bank loans. J Bank Financ 37(7):2354–2366. https://doi.org/10.1016/j.jbankfin.2013.01.031

Hartmann-Wendels T, Miller P, Töws E (2014) Loss given default for leasing: parametric and nonparametric estimations. J Bank Financ 40:364–375. https://doi.org/10.1016/j.jbankfin.2013.12.006

Hastie T, Tibshirani R, Friedman JH (2017) The elements of statistical learning: data mining, inference, and prediction, second edition, corrected at 12th printing, 2017th edn. Springer, New York

Heaton J (2008) Introduction to neural networks with Java, 2nd edn. Heaton Research, St. Louis, Mo
Hothorn T, Hornik K, Zeileis A (2006) Unbiased recursive partitioning: a conditional inference framework. Journal of computational and graphical statistics 15(3):651–674. https://doi.org/10.1198/106186006X133933
Hurlin C, Leymarie J, Patin A (2018) Loss functions for loss given default model comparison. Eur J Oper Res 268(1):348–360. https://doi.org/10.1016/j.ejor.2018.01.020
Kaposty F, Kriebel J, Löderbusch M (2020) Predicting loss given default in leasing: a closer look at models and variable selection. Int J Forecast 36(2):248–266. https://doi.org/10.1016/j.ijforecast.2019.05.009
Karatzoglou A, Smola A, Hornik K (2004) kernlab – an s4 package for kernel methods in R. J Stat Softw. https://doi.org/10.18637/jss.v011.i09
King RD, Feng C, Sutherland A (1995) Statlog: Comparison of classification algorithms on large real-world problems. Appl Artif Intell 9(3):289–333. https://doi.org/10.1080/08839519508945477
Kingma D, Ba J (2014) Adam: a method for stochastic optimization. In: International conference on learning representations
Krüger S, Rösch D (2017) Downturn LGD modeling using quantile regression. J Bank Financ 79:42–56. https://doi.org/10.1016/j.jbankfin.2017.03.001
Kuhn M, Johnson K (2016) Applied predictive modeling, corrected 5th, printing. Springer, New York
Kuhn M, Quinlan R (2018) Rule- and instance-based regression modeling. R Package Vers 2:1–14
Leisch F (2004) Flexmix: a general framework for finite mixture models and latent class regression in R. J Stat Softw. 10.18637/jss.v011.i08
Loterman G, Brown I, Martens D et al. (2012) Benchmarking regression algorithms for loss given default modeling. Int J Forecast 28(1):161–170. https://doi.org/10.1016/j.ijforecast.2011.01.006
MacQueen J (1967) Some methods for classification and analysis of multivariate observations. Proceedings of the fifth berkeley symposium on mathematical statistics and probability. University of California Press, Berkeley, Calif, pp 281–297
Matuszyk A, Mues C, Thomas LC (2010) Modelling LGD for unsecured personal loans: decision tree approach. J Oper Res Soc 61(3):393–398. https://doi.org/10.1057/jors.2009.67
Miller P, Töws E (2018) Loss given default adjusted workout processes for leases. J Bank Financ 91:189–201. https://doi.org/10.1016/j.jbankfin.2017.01.020
Min A, Scherer M, Schischke A et al. (2020) Modeling recovery rates of small- and medium-sized entities in the US. Mathematics 8(11):1856. https://doi.org/10.3390/math8111856
Mora N (2015) Creditor recovery: the macroeconomic dependence of industry equilibrium. J Financ Stab 18:172–186. https://doi.org/10.1016/j.jfs.2015.04.004
Nazemi A, Farnoosh FP, Heidenreich K et al. (2017) Fuzzy decision fusion approach for loss-given-default modeling. J Oper Res Soc 262(2):780–791. https://doi.org/10.1016/j.ejor.2017.04.008
Qi M, Zhao X (2011) Comparison of modeling methods for loss given default. J Bank Financ 35(11):2842–2855. https://doi.org/10.1016/j.jbankfin.2011.03.011
Querci F (2005) Loss given default on a medium-sized Italian bank's loans: an empirical exercise. European financial management association
Quinlan JR (1993) Combining instance-based and model-based learning. In: Proceedings of the tenth international conference on machine learning, ICML'93. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp 236–243
Smith G (2018) Step away from stepwise. J Big Data. https://doi.org/10.1186/s40537-018-0143-6
Sopitpongstorn N, Silvapulle P, Gao J et al. (2021) Local logit regression for loan recovery rate. J Bank Financ 126(4):106,093. https://doi.org/10.1016/j.jbankfin.2021.106093
Strobl C (2005) Variable selection bias in classification trees based on imprecise probabilities. Discuss Pap 419:386. https://doi.org/10.5282/ubm/epub.1788
Talaba G (2019) Hyperparameter tuning with caret for author name disambiguation. In: Proceedings of the 18th international conference on informatics in economy education, research and business technologies. bucharest university of economic studies press, international conference on informatics in economy proceedings, pp 129–134. https://doi.org/10.12948/ie2019.03.08
Tanoue Y, Yamashita S (2019) Loss given default estimation: a two-stage model with classification tree-based boosting and support vector logistic regression. J Risk 21(4):19–37. https://doi.org/10.21314/JOR.2019.405
Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc Ser B (Methodological) 58(1):267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Tipping ME, Smola A (2001) Sparse bayesian learning and the relevance vector machine. Journal of Machine Learning Research 1(3). https://doi.org/10.1162/15324430152748236

Tobback E, Martens D, van Gestel T et al. (2014) Forecasting loss given default models: impact of account characteristics and the macroeconomic state. J Oper Res Soc 65(3):376–392. https://doi.org/10.1057/jors.2013.158

van Gestel T, Suykens JA, Baesens B et al. (2004) Benchmarking least squares support vector machine classifiers. Mach Learn 54(1):5–32. https://doi.org/10.1023/B:MACH.0000008082.80494.e0

Vapnik VN (1995) The nature of statistical learning theory. Springer, New York. https://doi.org/10.1007/978-1-4757-2440-0

Ward JH (1963) Hierarchical grouping to optimize an objective function. J Am Stat Assoc 58(301):236. https://doi.org/10.2307/2282967

Williams C, Rasmussen C (1996) Gaussian processes for regression. Advances in neural information processing systems 8. Max-Planck-Gesellschaft. MIT Press, Cambridge, MA, USA, pp 514–520

Yang BH, Tkachenko M (2012) Modeling exposure at default and loss given default: empirical approaches and technical implementation. J Credit Risk 8(2):81–102. https://doi.org/10.21314/JCR.2012.139

Yao X, Crook J, Andreeva G (2015) Support vector regression for loss given default modelling. Eur J Oper Res 240(2):528–538. https://doi.org/10.1016/j.ejor.2014.06.043

Yao X, Crook J, Andreeva G (2017) Enhancing two-stage modelling methodology for loss given default with support vector machines. Eur J Oper Res 263(2):679–689. https://doi.org/10.1016/j.ejor.2017.05.017

Yashkir O, Yashkir Y (2013) Loss given default modeling: a comparative analysis. Journal Risk Model Valid 7(1):25–59. https://doi.org/10.21314/JRMV.2013.101

Ye H, Bellotti A (2019) Modelling recovery rates for non-performing loans. Risks 7(1):1–17. https://doi.org/10.3390/risks7010019

Zou H, Hastie T (2005) Regularization and variable selection via the elastic net. J R Stat Soc Ser B (Statistical Methodology) 67(5):768. https://doi.org/10.1111/j.1467-9868.2005.00527.x