ISSN 22799362

Machine Learning techniques in joint default assessment

Edoardo Fadda, Elisa Luciano, Patrizia. Semeraro

No. 723
July 2024

Carlo Alberto Notebooks

www.carloalberto.org/research/workingpapers

Machine Learning techniques in joint default assessment.

E. FADDA
Department of Mathematical Sciences G. Lagrange,
Politecnico di Torino.
E. LUCIANO[1]
ESOMAS Department and Collegio Carlo Alberto, Universitá di Torino
P. SEMERARO
Department of Mathematical Sciences G. Lagrange,
Politecnico di Torino.

June 3, 2024

[1] Corresponding author: Elisa Luciano, ESOMAS Department and Collegio Carlo Alberto, Universitá di Torino. Email: elisa.luciano@unito.it

Abstract

This paper empirically compares logistic regression with machine learning techniques in order to estimate the default risk measures and their bounds in large portfolios of identically distributed obligors. The methods compute different predictions of the probability of individual default and default correlation, so they are compared in various settings using increasing amounts of information: first the marginal probability, then the marginal probability and correlation, and lastly a specific model, the beta-binomial distribution. We make this evaluation using Value at Risk as well as Expected Shortfall in two settings: one synthetic and one real. In the synthetic setting, we construct portfolios of up to 10,000 obligors and test the performance of each method on 200 datasets. In the real setting, we use a publicly available credit card dataset of 30,000 obligors.

Keywords: Model Risk; Risk analysis; Bernoulli mixture model; ML methods; credit cards.

Classcode: JEL Classification codes: G32, C52, C60

Introduction

Pricing and hedging multi-name default products, such as Collateralized Debt Obligations (CDOs), heavily rely on assessing the joint default probability of several obligors. The number of obligors underlying these products is usually a hundred or more. Over the great recession, it became clear that a precise assessment of the joint default probability in credit portfolios was theoretically welcome, practically relevant for consumer protection, and needed for market stability. Nevertheless, the vast majority of the machine learning (ML) methods focus on the estimation of single default probabilities (see for instance Desai et al. (1996), Fitzpatrick and Mues (2016), Barbaglia et al. (2021) regarding loans and mortgages, and Khandani et al. (2010) for credit cards). In contrast to this stream of literature, the goal of this paper is to evaluate how traditional and ML-based methods behave in estimating the risk of portfolios of credits in terms of Value at Risk (VaR_α) and Expected Shortfall (ES_α).

In principle, to compute VaR_α and ES_α, it is enough to know the joint distribution of defaults, which is very difficult to estimate. Usually, the available information is the marginal individual default probability p and some partial information on the dependence structure, as the correlation among defaults ρ (see e.g. Embrechts et al. (2013), Bernard et al. (2023), and Barrieu and Scandolo (2015)). Nevertheless, by exploiting the estimation of p (or p and ρ), and by assuming exchangeable defaults (i.e., defaults with a joint distribution invariant under permutation of the default indicators) it is possible to analytically compute sharp risk bounds (i.e., there is a portfolio in the class reaching the risk bounds, Fontana et al. (2021), Fontana and Semeraro (2024)). The default exchangeability assumption is usually done when, as in this case, we consider homogeneous portfolios (i.e., defaults are identically distributed and exposures are equal). A standard exchangeable model is the exchangeable Bernoulli mixture model, where the joint distribution of defaults is inherited from the distribution of the one-dimensional mixing variable, that represents the distribution of the individual default probability.

Exploiting these results, we compare VaR_α and ES_α by using p and p, ρ estimated using Logistic Regression (LR), Multi-layer Perceptron (MLP), Random Forest (RF), Ada Boost (AB), and K-Nearest Neighbors (KNN). First, we consider the minimum and maximum VaR_α in the class of all portfolios with the estimated p. Second, we do the same in the class of all portfolios with the estimated p and ρ. The sensitivity of the bounds computed in the two settings for the selected techniques is a measure of the robustness of the estimations. Finally, we consider the VaR_α of a specific parametrical model to see if the results reflect the result obtained with partial information. We perform these tests in two settings: one synthetic and one real. In the synthetic setting, we construct homogeneous, exchangeable portfolios of 500 up to $10,000$ obligors, assume five given linear relationships among the covariates, and extract repeatedly from the covariates, to obtain 200 datasets. This analysis allows us to investigate the performance of the different methods when the true model is known. Instead, in the real setting, we use a publicly available credit card dataset and compare the results obtained for each method.

The paper is structured as follows: Section 1 presents the model for the Bernoulli variables which represent defaults; Section 2 contains the in-vitro and real analysis, Section 3 reports the conclusion of the work. Some technical results are in the Appendices.

1 The model

This section sets up the portfolio model, discusses the exchangeability assumption, its consequences, and presents the bounds of the risk measures.

1.1 From the marginal to the joint default probability

Let the random vector $\boldsymbol{Y} = (Y_1, \ldots, Y_d)'$ be the vector of default indicators of a set of d obligors or credit card owners over a fixed time horizon T. We assume that the vector $\boldsymbol{Y}$ is exchangeable, i.e. its joint distribution of defaults is invariant under permutations. This assumption implies that obligors have the same default probability and they are equicorrelated. Formally, let $P = (w_1, \ldots, w_d)$ be the percentage weights that represent a credit risk portfolio at time T associated with the d obligors, where $w_i \in (0, 1]$ and $\sum_{i=1}^{d} w_i = 1$. To model the loss of the portfolio P, we consider the sum of the percentage individual losses L, given by:

$$L = \sum_{i=1}^{d} w_i Y_i, \tag{1.1}$$

Since we assume exchangeability among defaults, the Y_i are identically distributed and we can consider the case of homogeneous exposures, meaning that the portfolio weights are assumed to be equal: $w_i = \frac{1}{d}$, $i = 1, \ldots, d$. In this case, the relevant quantity to asses the portfolio risk is the number of defaults,

$$S = \sum_{i=1}^{d} Y_i. \tag{1.2}$$

In the following, we model defaults using the exchangeable Bernoulli mixture models, a class of exchangeable models widely used in credit risk (see McNeil et al. (2005)).

Definition 1.1. *Given a random variable Q, the random vector $\boldsymbol{Y} = (Y_1, \ldots, Y_d)'$ follows an exchangeable Bernoulli mixture model with mixing variable Q with support on $[0, 1]$, if conditional on Q the default indicator $\boldsymbol{Y}$ is a vector of independent Bernoulli random variables with $\mathbb{P}(Y_i = 1|Q) = Q$.*

For a $\boldsymbol{Y}$ following an exchangeable Bernoulli mixture model, the unconditional - common - marginal default probability is

$$\mathbb{P}(Y_i = 1) = \int_0^1 q dG(q), \tag{1.3}$$

where $G(q)$ is the distribution of Q. Moreover, the unconditional probability mass function (pmf) $p_{\boldsymbol{Y}}(\boldsymbol{y})$ of $\boldsymbol{Y}$ is:

$$p_{\boldsymbol{Y}}(\boldsymbol{y}) = \mathbb{P}(\boldsymbol{Y} = \boldsymbol{y}) = \int_0^1 q^{\sum\limits_{i=1}^{d} y_i}(1-q)^{d-\sum\limits_{i=1}^{d} y_i} dG(q). \tag{1.4}$$

We define the cross moments of $\boldsymbol{Y}$ as

$$\pi_k \doteq E[Y_{i_1} \cdots Y_{i_k}], \;\; \{i_1, \ldots, i_k\} \subset \{1, \ldots, d\} \quad 1 \leq k \leq d. \tag{1.5}$$

We focus on the marginal default probability π_1, that we call p and on the equicorrelation, that is

$$\rho = \frac{\pi_2 - p^2}{p(1-p)}. \tag{1.6}$$

As a consequence of the exchangeability of the vector $\boldsymbol{Y}$, the unconditional distribution S of the number of defaults is entirely determined by the joint distributions of the default indicators (McNeil et al., 2005). In fact, if $\boldsymbol{Y}$ is exchangeable there is a one-to-one correspondence between the distribution of the number of defaults and the joint distribution of defaults (Fontana et al., 2021). In particular, in the Bernoulli mixture model the unconditional distribution $p_S(k)$ of the number of defaults S is

$$p_S(k) = \binom{d}{k} \int_0^1 q^k (1-q)^{d-k} dG(q). \tag{1.7}$$

It can be proved that the cross moments of $\boldsymbol{Y}$ are the moments of the mixing variable Q (McNeil et al., 2005), i.e.,

$$\pi_k = \mathbb{E}[Q^k]. \tag{1.8}$$

Comparing Eq. (1.8) with Eq. (1.5) it follows that the moments of the mixing variable Q completely determine all the moments and the joint distribution of defaults. Consequently the mixing variable Q determine the default probability, the equicorrelation and the distribution of the number of defaults. Different estimates of Q lead to different risk valuations.

We assume Q is a function h of random observable covariates $\boldsymbol{X} = (X_1, \ldots, X_n)$, representing the obligors' characteristics. Formally,

$$Q = h(\boldsymbol{X}). \tag{1.9}$$

In other words, the realizations of Q are functions of the realizations of $\boldsymbol{X}$ (i.e., $q = h(\boldsymbol{x})$). Therefore, if obligor j is characterized by $\boldsymbol{x}_j$ as covariates realizations, her default probability is $q_j = h(\boldsymbol{x}_j)$. This makes the realized default probability of each single obligor different.

The goal of the different techniques used to predict the default probability is to estimate as well as possible the function h. We call $\hat{h}$ the estimation of h. Different estimation techniques lead to different $\hat{h}$ and then to different estimates of the sample default probabilities and of its moments. Moreover, since it can be proved that both VaR_α and ES_α (as well as their bounds) depend on p and ρ (Fontana et al., 2021; McNeil et al., 2005), the choice of $\hat{h}$ affect also both the risk measures and their bounds. With this in mind, we can now state our main research question: what are the consequences on the portfolio risk of estimating p and ρ using different techniques?

To answer this question we consider two classes of exchangeable Bernoulli variables: The class $\mathcal{E}(p)$ of exchangeable Bernoulli distributions with p and the class $\mathcal{E}(p, \rho)$ of exchangeable Bernoulli distributions with the estimated mean p, and the estimated correlation ρ.

To find the risk corresponding to a specific model estimated with the different methods, we assume that Q follows a beta distribution with parameters a and b, i.e. $Q \sim \beta(a, b)$. In this setting, the number of defaults S in Eq. (1.7) is a beta-binomial distribution of parameters d, a and b (McNeil et al., 2005). Using Eq.(1.9), we estimate the beta parameters using the different techniques and measure their impact on the risk measures.

1.2 Bernoulli variables with given p and with given p and ρ

This section summarizes the geometrical structure of the class of distributions of the number of defaults S. We consider the family of distributions of the sums of the components of Bernoulli exchangeable vectors with distribution in $\mathcal{E}(p)$, $\mathcal{S}_d(p)$ and with distribution in $\mathcal{E}(p, \rho)$, named $\mathcal{S}_d(p, \rho)$. In Fontana et al. (2021) further details can be found including the proof of the results we report below for our convenience.

We recall that a polytope (or more specifically a d-polytope) is the convex hull of a finite set of points in $\mathbb{R}^d$ called the extremal points of the polytope.

The class $\mathcal{S}_d(p)$ $[\mathcal{S}_d(p, \rho)]$ is a d-polytope, i.e., for any $S \in \mathcal{S}_d(p)$ $[\mathcal{S}_d(p, \rho)]$ there exist $\lambda_1, \ldots, \lambda_{n_p} \geq 0$ summing up to 1 and $\boldsymbol{r}_j \in \mathcal{S}_d(p)$ $[\boldsymbol{r}_j \in \mathcal{S}_d(p, \rho)]$ such that

$$\boldsymbol{p}_Y = \sum_{j=1}^{n_p} \lambda_j \boldsymbol{r}_j, \tag{1.10}$$

where $\boldsymbol{r}_j = (r_j(0), \ldots, r_j(d))$, $j = 1, \ldots, n_p$ are the extremal points or the extremal densities of $\mathcal{S}_d(p)$ $[\mathcal{S}_d(p, \rho)]$.

The extremal points of $\mathcal{S}_d(p)$ and $\mathcal{S}_d(p, \rho)$ in (1.10) have been analytically found in Fontana et al. (2021), where the authors also provide the number of extremal points. For large d there are roughly $d^2 p(1-p)+1$ extremal points. For example for $d = 1000$ and $p = 0.1$ there are 90.000 extremal points.

1.3 Risk measure bounds

The importance of the geometrical representation is that several functionals reach their bounds at the extremal pmfs. In particular, Fontana et al. (2021) proves that the VaR_α bounds on classes of distributions that are convex polytopes are at the extremal points. Moreover, Fontana and Semeraro (2024) proves that for any functional Φ defined on a class of distributions that is a convex polytope the bounds of $\Phi(f) = E[\phi_d(X_f)]$ are reached on the extremal points.

As measures of portfolio risk, we consider the Value-at-Risk (VaR_α) and the Expected Shortfall (ES_α) that is a convex expectation measure of the number of defaults, S. We recall their definition for a general random variable Y.

Definition 1.2. *Let Y be a random variable representing a loss with a finite mean. Then the VaR_α at level α is defined by*

$$VaR_\alpha(Y) = \inf\{y \in \mathbb{R} : P(Y \leq y) \geq \alpha\}$$

and the expected shortfall at level α is defined by

$$ES_\alpha(Y) = \frac{1}{1-\alpha}(E[Y; Y \geq VaR_\alpha(Y)] + VaR_\alpha(Y)(1 - \alpha - P(Y \geq VaR_\alpha(Y)))).$$

Although both these measures reach their sharp bounds at the extremal points of $\mathcal{S}_d(p)$ and $\mathcal{S}_d(p, \rho)$, finding them by enumerating the values at each extremal point becomes infeasible for high dimension due to the huge number of extremal points. It is possible to to overcome this issue for the class $\mathcal{S}_d(p)$ due to the following Theorem (Proposition (5.4) in Fontana et al. (2021)) which provides an analytical expression for the bounds.

Theorem 1.1. *Let j_1^M be the largest integer smaller than pd, j_2^m be the smallest integer greater than pd and $j_1^p = \frac{(p-(1-\alpha))d}{\alpha}$. Let $MVaR = \max_{Y \in \mathcal{S}_d(p)} VaR_\alpha(Y)$ and $mVaR := \min_{Y \in \mathcal{S}_d(p)} VaR_\alpha(Y)$ It holds that:*

1. *if $p < 1-\alpha$, $mVaR = 0$ and $MVaR = [\frac{pd}{1-\alpha}]$ if $\frac{pd}{1-\alpha}$ is not integer and $MVaR = \frac{pd}{1-\alpha} - 1$ if it is integer;*

2. *if* $1-\alpha \le p \le 1-\alpha+\frac{\alpha}{d} j_1^M$, $mVaR = j_1^*$, *where* j_1^* *is the smallest integer greater or equal to* j_1^p *and* $MVaR = d$;

3. *if* $p > 1-\alpha+\frac{\alpha}{d} j_1^M$, $mVaR = j_2^m = j_1^M + 1$ *and* $MVaR = d$. *In this case, if* pd *is integer* $j_1^M + 1 = pd$.

Moreover, Proposition (5.2) in Fontana and Semeraro (2024) proves that the lower bounds for convex risk measures are reached at the specific extremal distribution in $\mathcal{S}_d(p)$ with support on the biggest integer lower than pd and the lowest integer higher than pd. This distribution corresponds to the safest dependence structure among the indicators of default, see e.g. Dhaene and Denuit (1999). Finally, it is possible to prove that the upper bound is reached on the extremal distribution with support on the two points 0 and d, that corresponds to the maximal dependence among the Bernoulli variables, i.e. among defaults (see e.g. Bernard et al. (2017) and Kaas et al. (2000)).

For the class $\mathcal{S}_d(p, \rho)$ there are not similar results and the only way to find the bounds is to proceed by enumeration. Actually, for the ES$_\alpha$ we can use a wiser method than complete enumeration by exploiting the convexity of ES$_\alpha$. In particular, let us consider a random variable having a probability mass function as:

$$X = \begin{cases} s & \text{with probability } \pi^s \\ 0 & \text{otherwise,} \end{cases} \qquad s \in \mathcal{S}$$

where $\mathcal{S} = \{1, \dots, S\}$ are the possible realizations. In Rockafellar and Uryasev (2000) the authors prove that the ES$_\alpha$ of X can be computed by solving:

$$\min \qquad \phi + \frac{1}{1-\alpha} \sum_s \pi^s z^s \tag{1.11}$$

$$\text{s.t.} \qquad z^s \ge s - \phi \qquad \forall s \in \mathcal{S} \tag{1.12}$$

$$\phi \in \mathbb{R}, z^s \in \mathbb{R}^+ \tag{1.13}$$

In our application, $\mathcal{S}$ represents the number of defaults (i.e., the support of the distribution). Since we aim to find a pmf with a given first and second moment, we can rewrite model (1.11)-(1.13), with π^s as a variable, obtaining:

$$
\begin{aligned}
\min \quad & \phi + \frac{1}{1-\alpha}\sum_s \pi^s z^s && && (1.14)\\
\text{s.t.} \quad & z^s \geq s - \phi && \forall s \in \mathcal{S} && (1.15)\\
& \sum_s \pi^s = 1 && && (1.16)\\
& \sum_s s\pi^s = pd && && (1.17)\\
& \sum_s s^2\pi^s = pd + d(d-1)\mu_2 && && (1.18)\\
& \phi \in \mathbb{R}, z^s \in \mathbb{R}^+, \pi^s \in [0,1] && \forall s \in \mathcal{S}, && (1.19)
\end{aligned}
$$

where μ_2 is the estimation of $\mathbb{E}[X^2]$. The objective function (1.14), together with the constraints (1.15) model the ES_α, while the constraints (1.16), (1.17), and (1.18) describe the characteristics of the probability distribution. Finally, Eq. (1.19) specifies the type of variables considered. Model (1.14)-(1.19) is not convex due to the product $\pi^s z^s$ in the objective function. Nevertheless, it can be solved to optimality (at least for small-size instances) by the commercial solver such as Gurobi (Gurobi Optimization, LLC, 2023).

Within this framework, we evaluate how different ML techniques impact these bounds. In particular, selecting different ML techniques leads to different models $\hat{h}$ which compute different realizations of $Q = \hat{h}(\mathbf{x})$. Then, we use the realizations of Q to estimate its moments $\hat{p}$, $\hat{\rho}$ (estimation of p and ρ) and $\hat{a}$ and $\hat{b}$ (estimation of the beta-parameters), and we can compute the bounds on the risk measure either by using the formula in Theorem 1.1 or the mathematical model.

2 Numerical experiments

In this section, we explore how different machine learning methods behave in different settings. Since an exhaustive list of experiments would require too much space, we provide the open-source code at `https://github.com/EdoF90/XXXX`, to properly guarantee reproducibility and enable the interested reader to carry out further experiments. The code has been developed in Python 3.10 and all the ML methods use the implementation of the scikit-learn python library (Pedregosa et al., 2011).
We split the section into two subsections: Section 2.1 considers the in-vitro analysis and Section 2.2 considers the Kaggle real dataset on credit card defaulters (`https://www.kaggle.com/uciml/default-of-credit-card-clients-dataset`).

In both examples, each obligor is characterized by a set of covariates and is associated with a label equal to 1 if she defaults, and 0 otherwise. In the in-vitro dataset, defaults are simulated according to a Bernoulli with probability computed by the true model. Instead, in the real dataset defaults are observed. This splits each dataset into two classes of obligors: the defaulted and the non-defaulted ones.

We consider the following five ML methods (Brownlee, 2016):

- Logistic Regression (LR): it estimates $\mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}]$ The default threshold is 0.5, but we update it using the best threshold according to Youden's J statistic (Youden, 1950).
- Random Forest Classifier (RF): it is an ensemble method based on the bagging algorithm considering several Decision Trees. Each decision tree recursively partitions a data set to divide the two labels ($Y = 1$, and $Y = 0$) through a sequence of tests (e.g., if the j-entry in $x_i, j = 1, .., n, i = 1, .., d$ is greater or smaller than a threshold).
- Multi-Layer Perceptron Classifier (MLP): it is one of the simplest architectures of neural networks. If it has no hidden layer, a sigmoid activation function, and is trained with a cross-entropy loss function, it is equivalent to an LR.
- K-Neighbors Classifier (KNN): it is a clustering technique aiming at splitting the feature spaces into two regions corresponding to the two labels.
- Ada Boost Classifier (AB): it is an ensemble method based on the boosting algorithm.

To find the best hyperparameters for each method we use a grid search with cross-validation. We refer to the online repository for the details. We point out that MLP is trained with a cross-entropy loss function because we aim to estimate probabilities.

It is important to note that different techniques lead to different results as they may have different goals. For example, LR is trained to minimize the probability error, while RF is trained to minimize the classification error. Therefore, to estimate probabilities it is better to rely on methods that focus on that or to use proper techniques (e.g., calibration). In the following, we will validate this assertion.

2.1 In-vitro analysis

The purpose of the in-vitro analysis is to provide a setting where the real model is known. This allows us to compare the performance of the different ML techniques in estimating p, ρ, and the beta-binomial parameters. The results themselves are relevant only as an instrument to understand the ML techniques in a controlled setting. For this reason, we estimate p, ρ, the beta parameter and compare the performance of each model and then

we only consider the VaR_α as a measure of risk to have a preliminary idea of the effect of these estimates on the portfolio risk.
In the in-vitro analysis, we generate several datasets by assuming a given function h and different relationships between the covariates $X_1, \dots X_n$. This enables us to easily compute the default probability $p = E[Q] = E[h(\boldsymbol{X})]$ and thus compute the modeling error of the various ML techniques.

The datasets are simulated by setting h as a logistic function of parameters $\boldsymbol{\theta}$. We set $\boldsymbol{\theta}$ in such a way that 20% of the observations are defaults. We chose this percentage intending to mimic realistic datasets as the one considered in Section 2.2. Moreover, similar default percentages are present in credit datasets (see Liu et al. (2022)).

We assume five different relationships between the covariates, together with different values and dimensions of $\boldsymbol{\theta}$. In the first relation, we assume two independent covariates. The second one introduces non-linear dependence between the two covariates. The third and fourth are constructed using copulas to introduce dependence and the last increases the number of covariates. We select these types of relations to span through polynomial functions a reasonable subset of possible dependence structures among covariates. We sample the covariates according to a uniform distribution between 0 and 1. This choice reflects the values of the covariates after the application of *covariate scaling* which removes effects such as the different magnitudes of the covariates.

- Relationship `uniform_independent` (`UI`) is generated through a logit model having two covariates, X_1, X_2 i.i.d. $\mathcal{U}[0,1]$ and $\boldsymbol{\theta} = [-0.2, 1.5, -5.0]$.

- Relationship `2_squared` (`2S`) is generated through a logit model with two covariates X_1, X_2, where $X_1 \sim \mathcal{U}[0,1]$ and $X_2 = X_1^2 + U[-0.05, 0.05]$ and $\boldsymbol{\theta} = [-0.2, 0.7, -5.5]$.

- Relationship `normal_copula` (`NC`) is generated through a logit model having two covariates whose marginal distributions are $\mathcal{U}[0,1]$ and who are linked through a Normal copula; the covariance matrix of the latter is

  $$\begin{bmatrix} 0.5 & -0.2 \\ -0.2 & 0.5 \end{bmatrix}. \tag{2.1}$$

  Moreover, we set $\boldsymbol{\theta} = [-0.2, 0.5, -3.2]$. We illustrate the values of the h function as a function of the covariates and a resulting simulation of defaults/non-defaults in Figure 1.

- Relationship `t_copula` (`TC`) is generated through a logit model as above but with the two marginal distributions linked through a Student t copula with 2 degrees of freedom and $\boldsymbol{\theta} = [-0.2, 0.5, -3.2]$.

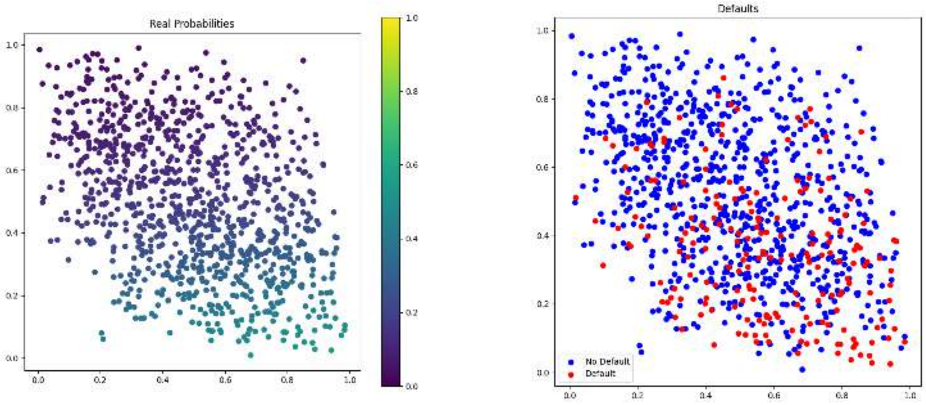

Figure 1: For the `2_squared` Relationship we show the default probability Q for each creditor (on the left) and a simulation of realizations of the individual defaults (on the right). On the right, the red dots are defaults. Both the probability and the default realizations are shown as a function of the covariates X_1 and X_2, which are reported on the axes.

- Relationship `5_non_linear` (`5NL`) is generated through a logit model where the covariates are: X_1, X_2, X_3 i.i.d. $\mathcal{U}[0,1]$ $X_1 \cdot X_2 + \mathcal{U}[-0.05, 0.05]$, $X_1 \cdot X_3 + \mathcal{U}[-0.05, 0.1]$, and $\boldsymbol{\theta} = [-0.1, -1, -0.5, -0.5, -1, -1]$.

For each relationship, we simulate 4 different portfolios with 500, 1000, 5000, and 10000 records, and we repeat the simulation 10 different times for each possible choice, resulting in a total of 5 (number of relationships) ×4 (number of portfolio sizes) ×10 (number of repetitions), namely 200 datasets.

We denote with $\hat{h} \in \{AB, LR, MLP, KNN, RF\}$ the ML method used to estimate the parameters on the synthetic datasets. We first use standard statistical methods to decide whether the datasets need to be balanced or calibration must be performed. Only once the corresponding choices are justified we report the moments of Q with the different ML techniques and comment on the corresponding VaRs.

According to Lessmann et al. (2015), the measure of the quality of an ML method depends on one of three possible goals: assessing the ability to discriminate between default and non-default, assessing the accuracy of the probability predictions, and assessing the correctness of the default predictions. We consider the AUC for the first, Brier Score for the second, and both precision and recall of the default class for the first, second, and third goals (see Geron (2022) for details on these measures). Note that we are interested in the default class, i.e., the set of obligors characterized by default, since they are more difficult to forecast (they are the 20% of the total observations). Since we know the true $h(\mathbf{X})$, instead of reporting the value of the indicators, we report the gap with respect to the true value achieved by each method. For example, calling AUC($\hat{h}$)

and AUC(h) the AUC of the ML method and the real model, we compute $GAPAUC[\%]$ as

$$\frac{\mathrm{AUC}(h) - \mathrm{AUC}(\hat{h})}{\mathrm{AUC}(\hat{h})}. \tag{2.2}$$

This quantity can be interpreted as the loss in the AUC caused by the modeling error. An analogous formula can be adopted to compute the gap for the other indicators.

Since the datasets are not balanced (they have 20% of the observations with the default labels and the remaining 80% with the non-default ones), we need to use the gaps to evaluate the impact of imbalance techniques. We focus on *SMOTE*, which is one of the most used imbalance techniques (similar results hold for other techniques such as *SMOTETomek*, *TomekLinks*, *RandomUnderSampler*, *RandomOverSampler*, *ClusterCentroids*, and *SMOTETomek*). We report the values of the percentage gap of AUC, Brier Score, precision, and recall of the default class (computed according to Eq. (2.2) or its analogous) for all the ML classifiers in all the datasets in Figure 2. The lower the gap values, the better the classifier is. Each boxplot collects the data on 5 ML algorithms, 5 relationships, and 10 repetitions (for a total of 250 data) for the balanced (named identity) and imbalanced (SMOTE) data, presented with different colors. In this figure, different portfolio sizes correspond to different couples of box plots.

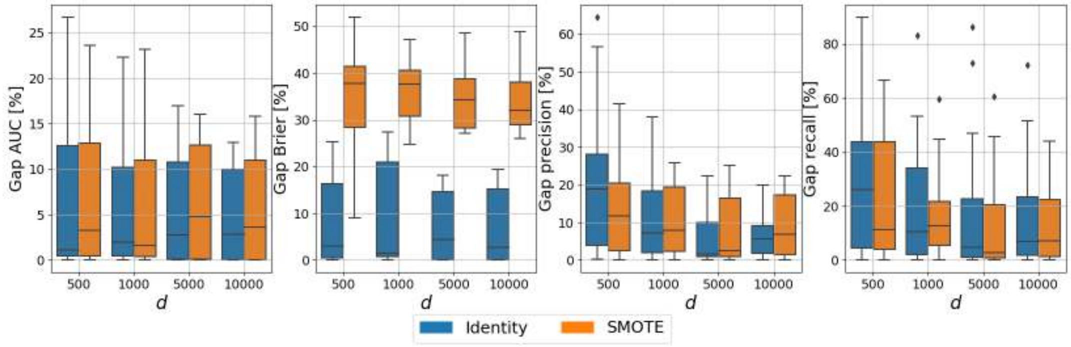

Figure 2: Boxplots of AUC, Brier Score, precision, and recall of the default class. Each boxplot collects the data on 5 ML algorithms, 5 relationships, and 10 repetitions (for a total of 250 data) for the balanced (named identity) and imbalanced (SMOTE) data, in blue and orange respectively. Different portfolio sizes d correspond to different couples of box plots.

Although imbalance techniques lead to higher precision and recall for datasets with $d = 500$, they do not have positive effects on datasets of higher dimensions. Moreover, independently of the size of d, *SMOTE* decreases the Brier Score. As a consequence, since our main interest is in default probabilities, we do not consider imbalance techniques.

As for calibration, several ML techniques focus on predicting the correct label (i.e., default/ no default) and do not aim to model default probabilities. When this is the case (as for the AB, RT, and KNN) it is important to calibrate the ML method (Brownlee, 2016). This procedure aims at minimizing the error of the forecasted probability. The two most commonly used techniques are the Sigmoid calibration (or Platt scaling), which is better if the calibration error is symmetric, and the isotonic calibration, which is a more powerful calibration method that can correct any monotonic distortion but that is prone to overfitting on small datasets (Brownlee, 2016). Since the smallest d is 500, we consider this last calibration method. We apply isotonic calibration to RF, AB, and KNN since both LR and MLP do not require it, as they minimize the log loss and the cross-entropy loss, respectively.

Calibration may change the performance in terms of precision and recall; thus, it is usually good practice to update the threshold for the methods after calibration. Since our goal is not classification, we consider just the Brier score and see if calibration is worth being used. We report the Brier score in Figure 3, where each boxplot represents the percentage Gap Brier score computed according to a formula analogous to Eq. (2.2). Each boxplot is obtained using 10 repetitions for each relationship, i.e. a total of 50 data. Again, we have a boxplot for each portfolio dimension d.

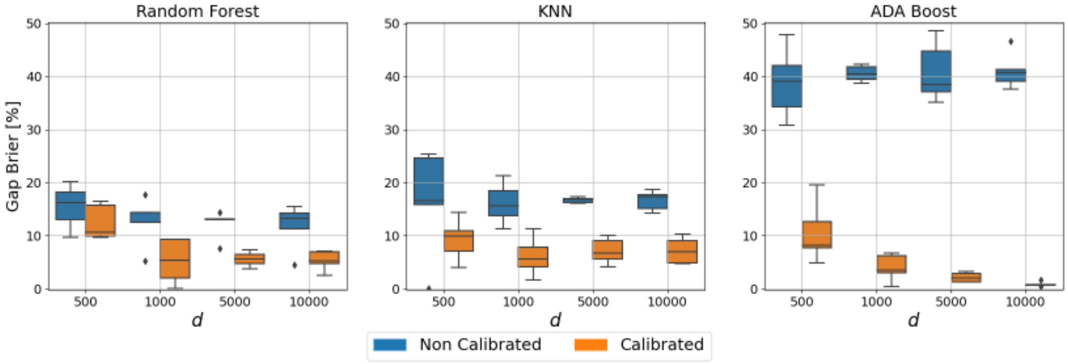

Figure 3: Gap of the Brier Score with and without calibration (in blue and orange respectively) for RF, AB, and KNN computed for different d.

As the reader can observe, calibration is beneficial for all the methods considered, especially for the AB, which thanks to calibration for $d = 1000$ reaches an average gap of 0.83% (with a standard deviation of 0.41%). We therefore work on calibrated data.

Once isotonic calibration and no smoothing techniques have been adopted, we compute the true default probabilities and we evaluate the average absolute error between the true and forecasted probability in the different ML methods, i.e., we compute

$$\frac{1}{d}\sum_{i=1}^{d} |h(\mathbf{x}_i) - \hat{h}(\mathbf{x}_i)|, \tag{2.3}$$

over all the portfolio sizes (4) and repetitions (10). The results for various settings are reported in Table 1.

	UI	2S	NC	TC	5NL
AB	4.55(2.33)	4.56(2.40)	4.61(2.15)	4.84(2.45)	4.53(1.55)
KNN	9.18(0.91)	9.19(0.81)	8.02(0.59)	8.65(0.95)	6.87(0.44)
LR	1.99(1.62)	2.23(1.44)	1.81(1.67)	2.01(1.68)	1.57(0.89)
MLP	1.66(1.17)	2.44(1.42)	1.84(1.57)	2.11(1.62)	1.64(0.98)
RF	8.38(1.14)	8.37(1.07)	7.72(0.97)	8.35(1.17)	5.89(1.02)

Table 1: Average error between true and simulated sample marginal default probability for different ML techniques (per row) and over different relationships (per column). Standard deviation in brackets.

The smallest errors in terms of sample marginal probabilities are obtained by LR and MLP. This is because we generate the data using a logit model so that these two methods do not have model errors. The third best method is AB which, on average, has an error of around 4%.

Table 2 reports the default probability $p_{\hat{h}}$ and equicorrelation $\rho_{\hat{h}}$ estimated with the sample default probabilities obtained with each ML method and the true sample probabilities of the synthetic datasets (called *real* in the table). While the former are similar, the latter are not. The best estimates are again obtained with LR and MLP, since they do not have model errors. However, the other ML methods provide better results when the covariates are generated with a non-linear model (5NL). The best estimates are obtained with AB and the worst with KNN, because they are the methods with the lowest and highest absolute error, respectively (see Table 1). The table contains the results for the portfolio of 5000 obligors (for the others see Appendix B)

	UI		2S		NC		TC		5NL	
	p	ρ	p	ρ	p	ρ	p	ρ	p	ρ
AB	0.20	0.25	0.21	0.16	0.20	0.07	0.23	0.11	0.18	0.07
KNN	0.20	0.15	0.21	0.08	0.20	0.02	0.23	0.04	0.19	0.02
LR	0.20	0.23	0.20	0.14	0.20	0.07	0.23	0.10	0.18	0.07
MLP	0.20	0.25	0.21	0.15	0.20	0.07	0.22	0.10	0.18	0.07
RF	0.20	0.18	0.21	0.12	0.20	0.04	0.23	0.05	0.19	0.05
real	0.20	0.24	0.22	0.16	0.20	0.09	0.21	0.11	0.19	0.06

Table 2: Each row represents the average of p and ρ for different ML methods. The average is over 10 repetitions of the 5000 portfolio, for different relationships (across columns)

Passing now to risk bounds for joint defaults, we compute the Beta-Binomial and lower bound VaRs for $\alpha = 0.9$ and portfolio size 5000. Figure 4 represents the boxplot of the percentage $\mathrm{VaR}_{0.9}$ and the lower bound for the percentage $\mathrm{VaR}_{0.9}$ considering 10 repetitions. The bottom box plots represent the lower bounds for the VaR_{α} in $\mathcal{S}(p_{\hat{h}})$

computed according to Proposition 5.2 in Fontana et al. (2021) using

$$p_{\hat{h}} = \frac{1}{d}\sum_{i=1}^{d} \hat{h}(\mathbf{x}_i).$$

Instead, the upper part shows the beta-binomial $\mathrm{VaR}_{0.9}$ computed in the following way:

1. for each obligor we compute $p_i^{\hat{h}} := \hat{h}(\mathbf{x}_i)$
2. using the observation of $p_i^{\hat{h}}$, we calibrate a beta-binomial using the methods of moments
3. we compute analytically the $\mathrm{VaR}_{0.9}$ of the estimated beta-binomial distribution.

The upper bounds are omitted because they are 100%. While the lower bounds are similar across the models and the different synthetic data, the VaR_α values obtained using the beta-binomial model are more sensitive to the ML model used and to the structure of the synthetic data.

Moreover, it is possible to see that:

- in all the datasets KNN and RF underestimate the VaR_α (looking at Table 2 you can see how KNN often underestimates ρ; a similar, but less extreme, behavior characterizes RF).
- all the other models (LR, MLP, and AB) have values similar to the true one (therefore they are able to describe the VaR_α well).
- all the lower bounds are practically the same, which means that the estimate of the lower bounds is robust with respect to the choice of the model.
- the VaRs in the relationship `5NL` present very low variability across the different datasets.

Summing up, the in-vitro analysis tells us that if the true model is the LR, the estimates of p are robust with respect to the ML technique chosen, as well as the corresponding VaR_α bounds in $\mathcal{E}(p)$. On the contrary, the estimates of ρ are different, meaning that correlation is more sensitive to the ML method adopted. This is also reflected in the estimate of the joint beta-binomial model, whose parameters are estimated using the first and second-order sample moments of Q - that correspond to the mean and correlation among defaults. In these two cases the best performance is obtained with LR (that does not have model risk), MLP (that also does not have model risk), and AB.

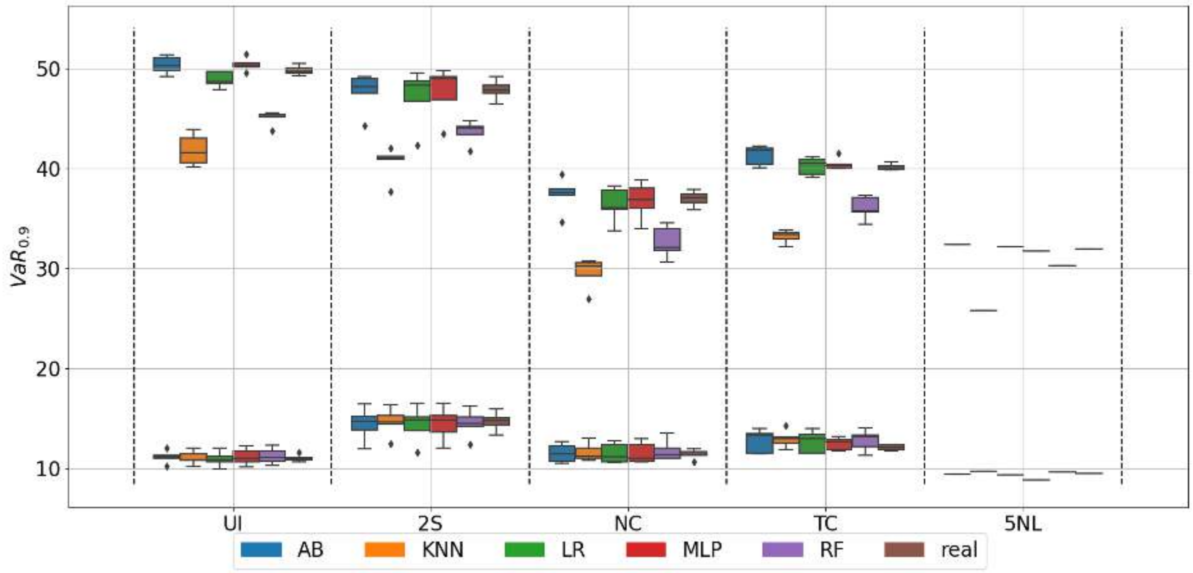

Figure 4: Lower bound for the VaR_α and VaR_α computed using the binomial, level $\alpha = 0.9$. The values are expressed in percentages.

2.2 Real data application

We now perform the full analysis on a real database. The real, Kaggle dataset collects data from 30000 clients of a bank issuing credit cards in Taiwan, from April to September 2005. The dataset contains information on 24 covariates, including demographic factors, credit data, history of payment, and bill statements of credit card customers from April 2005 to September 2005. The dataset obviously contains the binary response of default for each obligor. We only consider continuous covariates, i.e. covariates that assume values on the real line, as opposed to discrete ones, to reduce the difference with respect to the in-vitro datasets. In fact, this study aims to compare the effect of adopting different ML methods and not to analyze a specific group of obligors. Although the results themselves do not incorporate all the available information, the magnitudes of the risk measures are coherent with the corresponding measures obtained in the preliminary analysis performed on this dataset in Doria et al. (2022), where all the covariates were included in the analysis. Relying on the in-vitro analysis we do not apply imbalance techniques, but we calibrate the probabilities.

We construct a high dimensional portfolio of $d = 6000$ obligors, find the risk measures bound in $\mathcal{E}(p)$, and estimate the parametrical model. Due to the computational limits when finding by enumeration the VaR_α and ES_α bounds in $\mathcal{E}(p, \rho)$ discussed in Section 1.3, the bounds in this class are found on a portfolio of $d' = 200$ obligors and only for the ES_α, by solving the optimization in Eqs. (1.11)-(1.13). To make the results obtained with the two portfolios of different sizes comparable, we present them in terms of percentages instead of the number of defaults.

	Beta parameters			
	a	b	p	ρ
AB	0.81 [0.77, 0.85]	2.88 [2.67, 3.01]	0.22 [0.22, 0.23]	0.21 [0.21, 0.23]
KNN	0.50 [0.39, 0.59]	1.78 [1.39, 2.11]	0.22 [0.21, 0.23]	0.31 [0.27, 0.36]
LR	2.44 [2.34, 2.53]	6.80 [6.63, 6.97]	0.26 [0.26, 0.27]	0.10 [0.10, 0.10]
MLP	0.63 [0.55, 0.75]	2.27 [1.91, 2.54]	0.22 [0.20, 0.23]	0.26 [0.23, 0.29]
RF	0.94 [0.89, 1.00]	3.31 [3.08, 3.55]	0.22 [0.22, 0.23]	0.19 [0.18, 0.20]

Table 3: Parameters of the beta distribution and corresponding p and ρ of the different ML techniques.

To give an initial sense of how different the ML techniques can be, we focus first, as we did for the in vitro case, on the marginal probabilities: here we cannot compare the estimated beta distribution with the theoretical one, but we can appreciate how far apart they are. Table 3 reports the estimated beta parameters for the distribution of the default probability. Since one estimation can be noisy, we report the problem by sampling 50 different training sets. The LR and the ML techniques lead to different results in the shape and tail of the estimated distribution. This generates different estimates of the default probability and equicorrelation, also reported in Table 3.

Here, we report the coefficients a and b of the beta distribution estimated using the method of moments from the probabilities $\hat{h}(\mathbf{x}_i)$ as in the previous Section.

Considering the results in Table 3, we can notice that:

- LR has p greater and ρ smaller than the other methods. The in-vitro analysis shows that if the true model is LR the other ML methods provide good estimates of p. These results support the idea that LR does not fit well the real data.
- all the ML methods are close to each other.

These parameters are the main factors affecting the portfolio risk, which we measure using both VaR_α and ES_α.

2.2.1 Bounds using only the marginal probability

In this section, we compute both VaR_α and ES_α analytically computing them from a beta-binomial distribution calibrated, using the methods of moments, on the probabilities $\hat{h}(\mathbf{x}_i)$. Moreover, using the average of $\hat{h}(\mathbf{x}_i)$, we also compute the lower bounds using Proposition 5.2 in Fontana et al. (2021). Since one estimation can be noisy we repeat the problem by sampling 50 different training sets. The average results are reported in Table 4 and in Table 5, for the VaR_α and the ES_α, respectively (together with an interval obtained from the minimum and maximum value observed). As the reader can notice the effects of a higher estimate of p and of a higher estimate of ρ are opposite.

		α		
		0.9	0.95	0.99
VaR_α β bin	AB	49.50 [49.45, 52.23]	60.94 [59.63, 62.80]	77.46 [76.18, 79.38]
	KNN	58.06 [54.93, 62.45]	70.69 [66.80, 76.11]	87.65 [84.10, 92.15]
	LR	45.56 [45.21, 45.85]	51.86 [51.46, 52.23]	63.26 [62.80, 63.75]
	MLP	53.83 [51.45, 57.30]	65.36 [62.66, 69.48]	82.56 [80.05, 86.43]
	RF	48.77 [47.53, 50.01]	58.36 [56.90, 59.91]	74.33 [72.61, 76.10]
min VaR_α	AB	13.42 [12.82, 13.97]	17.97 [17.40, 18.50]	21.29 [20.75, 21.80]
	KNN	13.36 [12.37, 14.05]	17.93 [16.97, 18.58]	21.24 [20.33, 21.87]
	LR	18.21 [17.75, 18.62]	22.52 [22.08, 22.90]	25.65 [25.23, 26.02]
	MLP	13.12 [11.47, 14.40]	17.69 [16.12, 18.90]	21.02 [19.50, 22.18]
	RF	13.42 [12.87, 13.98]	17.97 [17.45, 18.50]	21.29 [20.78, 21.80]

Table 4: VaR_α for different α, $d = 6000$. The values are expressed in percentage.

The range for the VaR_α and ES_α that depend only on p are significantly wider for the LR estimate, while the beta-binomial VaR_α and ES_α, that depend on two moments, are significantly higher for the ML estimates. The latter provide results close to the other.

Here we can notice that:

- For all the α we have that the LR entry is the lowest. AB and RF follow. The highest entries are given by MLP and KNN. So all MLs are above LR. We cannot say which of the MLs is more correct but we can say that LR underestimates ρ (both from Table 3 and because otherwise MLP should have results similar to it, since it is a more flexible model).

- For the ML estimates the lower VaR_α bounds are close to each other, with LR standing out from all the others; it is reasonable to think that LR is not able to describe the dataset considered well.

- For the ML estimates the beta-binomial VaR_α has higher values than LR; in this case, we see more variability also within different ML methods.

- MLP VaR_α has values that intersect AB, RF, and KNN, so it can be seen as a tradeoff.

- Considering the results of Table 3 we see that the greater VaR_α measured by ML goes together with a higher ρ.

- Similar conclusions hold for ES_α.

		α		
		0.9	0.95	0.99
ES β bin	AB	63.18 [61.92, 65.00]	70.92 [69.67, 72.82]	83.32 [82.12, 85.10]
	KNN	72.48 [68.83, 77.42]	80.95 [77.27, 85.90]	92.14 [89.18, 95.52]
	LR	53.66 [53.25, 54.07]	58.84 [58.38, 59.27]	68.45 [68.02, 68.98]
	MLP	67.47 [64.90, 71.35]	75.75 [73.13, 79.70]	87.93 [85.75, 91.10]
	RF	60.62 [59.18, 62.18]	68.03 [66.47, 69.73]	80.36 [78.73, 82.00]
min ES	AB	22.09 [21.55, 22.60]	22.09 [21.55, 22.60]	22.10 [21.55, 22.60]
	KNN	22.04 [21.13, 22.67]	22.04 [21.13, 22.67]	22.04 [21.13, 22.67]
	LR	26.40 [25.97, 26.77]	26.40 [25.97, 26.77]	26.41 [25.98, 26.77]
	MLP	21.82 [20.33, 22.98]	21.82 [20.33, 22.98]	21.82 [20.33, 22.98]
	RF	22.09 [21.60, 22.60]	22.09 [21.60, 22.60]	22.09 [21.60, 22.60]

Table 5: ES_α for different α, $d = 6000$. The values are expressed in percentage.

2.2.2 Bounds using the marginal probability and the correlation

Using the results in Fontana et al. (2021) we could in principle find the bounds for the VaR_α in $\mathcal{S}_d(p, \rho)$, as the authors do for a portfolio of dimension $d = 100$. However, the number of extreme points increases very fast and it is computationally infeasible to compute the VaR_α for enumeration for larger portfolios. For example with $d = 1000$ and $p = 0.2$ the number of extremal points in $\mathcal{E}(p)$ is 160.000. Building on the convexity of ES_α we can find its bounds on $\mathcal{S}(p, \rho)$, using a different approach and solving the optimization method described in Eqs (1.11)-(1.13), which can be solved by the commercial solver Gurobi. We set $d = 200$ which leads to instances that can be solved in 1 minute. The results are shown in Table 6.

Here, we can notice that:

- the lower bound computed using only p is higher for the LR and the same happens for all the ML methods, as in the previous section;
- the lower bounds computed using only p and ρ are lower for LR and higher for all the ML methods. However, they differ across themselves more than if only p is considered;
- an analogous result holds for the ES_α Beta (KNN leads to the greatest ES_α, MLP AB and RF are almost the same and LR is way lower);

As a final consideration, we observe that ML methods provide more similar risk evaluation than LR. This is evident for bounds produced using only p, where ML estimates also provide similar results. Including correlation ML estimates differ across themselves, supporting the idea that the risk measure is more sensitive to the correlation estimate than to the marginal default probabilities. While for estimating default probability the

		ES_p	ES_β	$ES_{p,\rho}$
$\alpha = 0.9$	AB	22.40 [22.00, 22.50]	64.50 [64.00, 65.50]	39.10 [38.50, 39.50]
	KNN	22.20 [21.50, 23.00]	72.80 [70.00, 75.50]	45.60 [44.00, 47.50]
	LR	26.60 [26.50, 27.00]	55.70 [55.00, 56.50]	34.20 [34.00, 34.50]
	MLP	21.70 [20.50, 22.50]	69.10 [66.50, 72.50]	43.00 [40.50, 45.50]
	RF	22.30 [22.00, 22.50]	62.00 [61.00, 63.00]	37.50 [37.00, 38.00]
$\alpha = 0.95$	AB	22.40 [22.00, 22.50]	72.60 [71.50, 74.00]	39.10 [38.50, 39.50]
	KNN	22.20 [21.50, 23.00]	81.70 [80.00, 83.50]	45.60 [44.00, 47.50]
	LR	26.60 [26.50, 27.00]	60.80 [60.00, 61.50]	34.20 [34.00, 34.50]
	MLP	21.70 [20.50, 22.50]	77.10 [74.50, 80.00]	42.90 [40.50, 45.50]
	RF	22.30 [22.00, 22.50]	69.80 [69.00, 70.50]	37.50 [37.00, 38.00]
$\alpha = 0.99$	AB	22.40 [22.00, 22.50]	87.40 [85.50, 89.00]	39.10 [38.50, 39.50]
	KNN	22.20 [21.50, 23.00]	95.00 [92.50, 98.00]	45.60 [44.00, 47.50]
	LR	26.60 [26.50, 27.00]	72.80 [71.50, 74.00]	34.20 [34.00, 34.50]
	MLP	21.70 [20.50, 22.50]	90.30 [89.00, 92.00]	42.90 [40.50, 45.50]
	RF	22.30 [22.00, 22.50]	83.70 [82.00, 85.50]	37.50 [37.00, 38.00]

Table 6: ES_p: ES_α lower bound in $\mathcal{S}(p)$; $ES_{p,\rho}$: ES_α lower bound in $\mathcal{S}(p,\rho)$; ES_β: ES_α of the beta-binomial model. The values are expressed in percentage.

only relevant choice is to adopt a traditional LR estimate or a ML technique, when incorporating correlation as well as for the estimate of a specific joint distribution for the vector of default indicators the choice of a specific ML method becomes relevant.

3 Conclusions

Both in the in-vitro and in the actual dataset case, risk measures do differ between LR and ML techniques.

In the in-vitro case, which we know to be linear, LR performs very well, by construction. Sophisticated neural network techniques - such as the MLP - which collapse into the LR when needed, are so flexible that they can capture even this extreme case in which traditional techniques are by definition sufficient. However, alternative methods such as AB also perform very well in capturing the risk.

The real point of the analysis is that, although all ML methods give similar estimates of the marginal probability on single defaults (as expected from the existing literature), they actually differ in the estimate of the correlation, and therefore give different bounds for the risk measures and different specifications for single models such as the beta-binomial distribution. This consideration is helpful in reading the real credit case application, where the appropriateness of a specific model can no longer be judged from its distance from the true VaR_α. Risk measures differ between LR and ML techniques, and also among ML techniques, when we use both the marginal default probability and the correlation, or a specific reference model such as the beta-binomial distribution. When little information is used, the estimates from different ML approaches are closer. ML techniques provide different risk bounds in the richer model with correlation or under the binomial assumption.

We conclude that for credit portfolios, if only information on the mean is available, the risk measures significantly differ between LR or ML. In contrast, they are very close across all the different ML techniques. When information on correlation is included, the difference between LR and ML remains, but more variability is revealed across the different ML techniques: this is due to the way in which different models exploit data information.

4 Acknowledgements

This work is partially supported by the Italian Ministry of Education, University and Research (MIUR), PRIN 2022-PNRR project P2022XT8C8.

A Dataset description

The dataset contains 30.000 obligors with 24 covariates each, namely:

- ID: ID of each client.
- LIMIT_BAL: Amount of the given credit (NT dollar), which includes both the individual consumer credit and his/her family (supplementary) credit.
- SEX: Gender (1 = male; 2 = female).
- EDUCATION: (1 = graduate school; 2 = university; 3= high school; 4= others; 5= unknown; 6= unknown).
- MARRIAGE: Marital status (1 = married; 2 = single; 3 = others).
- AGE: Age in years.
- PAY_1 to 6 : Repayment status in September to April 2005.
- BILL_AMT1 to 6: Amount of bill statement in September to April, 2005 (NT dollar).
- PAY_AMT1: Amount of previous payment in September to April, 2005 (NT dollar).

- default.payment.next.month: Default payment (1=yes, 0=no).

We do not include in our analysis sex, education, marriage, and age, because they are not continuous variables. Table 7 provides the descriptive statistics of the data.

-	**def_pay**	**LIMIT_BAL**	**SEX**	**EDUCATION**	**MARRIAGE**	**AGE**
mean	0.2212	167484.3227	-	-	-	35.4855
std	0.4150	129747.6616	-	-	-	9.217904068
-	**BILL_AMT1**	**BILL_AMT2**	**BILL_AMT3**	**BILL_AMT4**	**BILL_AMT5**	**BILL_AMT6**
mean	51223.3309	49179.07517	47013.1548	43262.94897	40311.40097	38871.7604
std	73635.86058	71173.76878	69349.38743	64332.85613	60797.15577	59554.10754
-	**PAY_1**	**PAY_2**	**PAY_3**	**PAY_4**	**PAY_5**	**PAY_6**
mean	-	-	-	-	-	-
std	-	-	-	-	-	-
-	**PAY_AMT1**	**PAY_AMT2**	**PAY_AMT3**	**PAY_AMT4**	**PAY_AMT5**	**PAY_AMT6**
mean	5663.5805	5921.1635	5225.6815	4826.076867	4799.387633	5215.502567
std	16563.28035	23040.8704	17606.96147	15666.15974	15278.30568	17777.46578

Table 7: Descriptive statistics.

B Other Results

In this section, we report the extended results for the computational experiments shown in Section 2.1. In particular, Table 8 reports the results of p and ρ of the different ML methods for the different relationships and the different numbers of obligors. The last rows of the table report p and ρ computed using the true model. As the reader can notice, the results are all very close to the true one already when 500 obligors are considered. This further testifies the expressive power of ML methodologies.

Instead, in Figure 5 we represent the percentage VaR_α with $\alpha = 0.9$ for different numbers of obligors (d). For completeness, we report $d = 1000$ (i.e., Figure 4). As the reader can notice, the general trend is that the more data, the smaller the boxplot variation, with some small exceptions due to the stochasticity in the data.

In all the plots it is possible to observe that:

- the variation for the relationship `5NL` dataset is far smaller than the other and it shrinks very fast as d increases;
- the relationship `TC` has the second greatest variation (after `5NL`) but the reduction of variance is far smaller:
- the bound are very similar independently by the methodology used;
- logistic regression performances are deeply influenced by d and it start to have good performance from $d = 5000$.

		UI		2S		NC		TC		5NL	
	d	p	ρ	p	ρ	p	ρ	p	ρ	p	ρ
AB	500	0.21	0.22	0.25	0.17	0.21	0.08	0.21	0.06	0.18	0.03
	1000	0.19	0.24	0.27	0.19	0.20	0.09	0.17	0.14	0.17	0.08
	5000	0.20	0.25	0.21	0.16	0.20	0.07	0.23	0.11	0.18	0.07
	10000	0.20	0.23	0.24	0.16	0.20	0.08	0.20	0.11	0.19	0.07
KNN	500	0.20	0.11	0.24	0.11	0.23	0.04	0.23	0.02	0.18	0.03
	1000	0.19	0.14	0.26	0.10	0.20	0.02	0.17	0.06	0.17	0.04
	5000	0.20	0.15	0.21	0.08	0.20	0.02	0.23	0.04	0.19	0.02
	10000	0.20	0.14	0.24	0.08	0.20	0.02	0.21	0.04	0.19	0.02
LR	500	0.20	0.14	0.23	0.11	0.22	0.06	0.22	0.04	0.18	0.04
	1000	0.20	0.19	0.26	0.14	0.20	0.06	0.17	0.11	0.17	0.07
	5000	0.20	0.23	0.20	0.14	0.20	0.07	0.23	0.10	0.18	0.07
	10000	0.20	0.23	0.24	0.15	0.20	0.08	0.20	0.10	0.19	0.07
MLP	500	0.20	0.18	0.23	0.12	0.21	0.07	0.22	0.04	0.18	0.03
	1000	0.20	0.23	0.26	0.14	0.20	0.07	0.17	0.12	0.17	0.07
	5000	0.20	0.25	0.21	0.15	0.20	0.07	0.22	0.10	0.18	0.07
	10000	0.20	0.23	0.23	0.15	0.20	0.08	0.21	0.11	0.18	0.07
RF	500	0.20	0.22	0.24	0.15	0.21	0.09	0.22	0.05	0.17	0.05
	1000	0.20	0.21	0.26	0.15	0.21	0.05	0.15	0.09	0.17	0.09
	5000	0.20	0.18	0.21	0.12	0.20	0.04	0.23	0.05	0.19	0.05
	10000	0.20	0.16	0.24	0.10	0.20	0.04	0.21	0.06	0.19	0.05
true	500	**0.20**	**0.23**	**0.23**	**0.16**	**0.19**	**0.09**	**0.23**	**0.10**	**0.19**	**0.06**
	1000	**0.20**	**0.25**	**0.25**	**0.16**	**0.21**	**0.09**	**0.19**	**0.11**	**0.18**	**0.07**
	5000	**0.20**	**0.24**	**0.22**	**0.16**	**0.20**	**0.09**	**0.21**	**0.11**	**0.19**	**0.06**
	10000	**0.20**	**0.24**	**0.24**	**0.16**	**0.21**	**0.09**	**0.21**	**0.11**	**0.19**	**0.06**

Table 8: Each cell represents average p and ρ for the different ML methods. The bold numbers are the ones achieved by using the real probability computed by $\hat{h}(X)$.

Acknowledgements

Elisa Luciano gratefully acknowledges financial support from the Italian Ministry of Education, University and Research, MIUR, ”Dipartimenti di Eccellenza” and from the PNRR project Growing Resilient, INclusive and Sustainable - PE00000018 - CUP D13C22002160001. Elisa Luciano and Patrizia Semeraro are partially supported by the Italian Ministry of Education, University and Research (MIUR), PRIN 2022-PNRR project P2022XT8C8. Edoardo Fadda thanks Evelina Di Corso for helpful discussions.

References

Barbaglia, L., Manzan, S., and Tosetti, E. (2021). Forecasting loan default in europe with machine learning. *Journal of Financial Econometrics*, 21(2):569–596.

Barrieu, P. and Scandolo, G. (2015). Assessing financial model risk. *European Journal of Operational Research*, 242(2):546–556.

Bernard, C., De Vecchi, C., and Vanduffel, S. (2023). The impact of correlation on (range) value-at-risk. *Scandinavian Actuarial Journal*, 2023(6):531–564.

Bernard, C., Rüschendorf, L., Vanduffel, S., and Yao, J. (2017). How robust is the value-at-risk of credit risk portfolios? *The European Journal of Finance*, 23(6):507–534.

Brownlee, J. (2016). *Machine Learning Mastery with Python: Understand Your Data, Create Accurate Models and Work Projects End-to-end*. Jason Brownlee.

Desai, V. S., Crook, J. N., and Overstreet Jr, G. A. (1996). A comparison of neural networks and linear scoring models in the credit union environment. *European journal of operational research*, 95(1):24–37.

Dhaene, J. and Denuit, M. (1999). The safest dependence structure among risks. *Insurance: Mathematics and Economics*, 25(1):11–21.

Doria, M., Luciano, E., and Semeraro, P. (2022). Machine learning techniques in joint default assessment. *arXiv preprint arXiv:2205.01524*.

Embrechts, P., Puccetti, G., and Rüschendorf, L. (2013). Model uncertainty and var aggregation. *Journal of Banking & Finance*, 37(8):2750–2764.

Fitzpatrick, T. and Mues, C. (2016). An empirical comparison of classification algorithms for mortgage default prediction: evidence from a distressed mortgage market. *European Journal of Operational Research*, 249(2):427–439.

Fontana, R., Luciano, E., and Semeraro, P. (2021). Model risk in credit risk. *Mathematical Finance*, 31(1):176–202.

Fontana, R. and Semeraro, P. (2024). High dimensional bernoulli distributions: Algebraic representation and applications. *Bernoulli*, 30(1):825–850.

Geron, A. (2022). *Hands-on machine learning with scikit-learn, keras, and TensorFlow*. O'Reilly Media, Sebastopol, CA, 3 edition.

Gurobi Optimization, LLC (2023). Gurobi Optimizer Reference Manual.

Kaas, R., Dhaene, J., and Goovaerts, M. J. (2000). Upper and lower bounds for sums of random variables. *Insurance: Mathematics and Economics*, 27(2):151–168.

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

Lessmann, S., Baesens, B., Seow, H.-V., and Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, 247(1):124–136.

Liu, Y., Yang, M., Wang, Y., Li, Y., Xiong, T., and Li, A. (2022). Applying machine learning algorithms to predict default probability in the online credit market: Evidence from china. *International Review of Financial Analysis*, 79:101971.

McNeil, A. J., Frey, R., and Embrechts, P. (2005). *Quantitative risk management*, volume 3. Princeton university press.

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

Rockafellar, R. T. and Uryasev, S. (2000). Optimization of conditional value-at-risk. *The Journal of Risk*, 2(3):21–41.

Youden, W. J. (1950). Index for rating diagnostic tests. *Cancer*, 3(1):32–35.

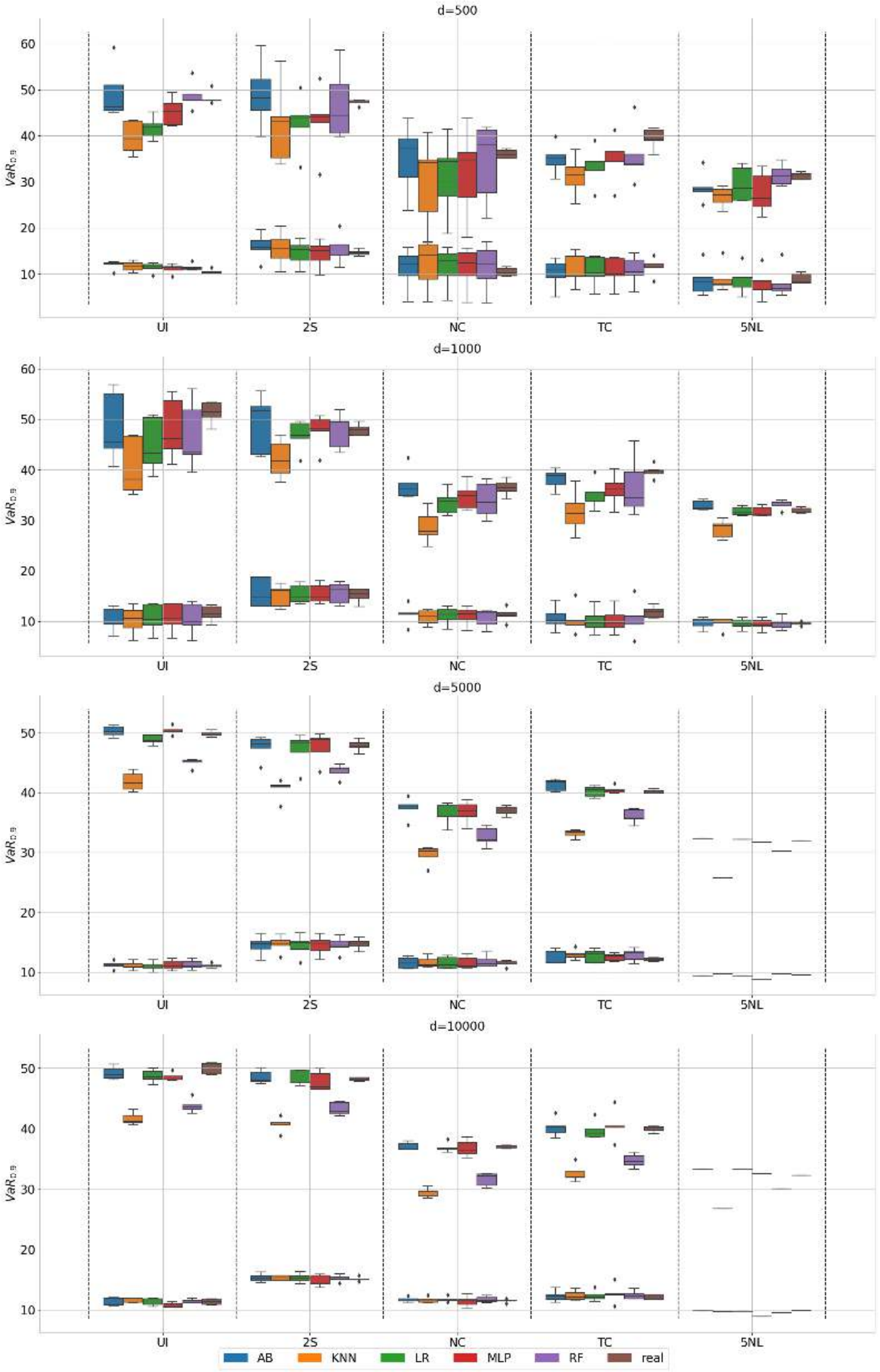

Figure 5: Lower bound for the VaR_α and VaR_α computed using the binomial, level $\alpha = 0.9$.