CREDIT SELECTION IN COLLATERALIZED LOAN OBLIGATION: EFFICIENT APPROXIMATION THROUGH LINEARIZATION AND CLUSTERING

Arnaud Germain, Frédéric Vrins

LIDAM Discussion Paper LFIN
2024 / 06

LFIN

Voie du Roman Pays 34, L1.03.01 B-1348 Louvain-la-Neuve

Tel (32 10) 47 43 04

Email: lidam-library@uclouvain.be

https://uclouvain.be/en/research-institutes/lidam/lfin/publications.html

Credit selection in Collateralized Loan Obligation: efficient approximation through linearization and clustering*

Arnaud Germain[1] and Frédéric Vrins[1,2]

[1] *Louvain Institute of Data Analysis and Modeling in economics and statics (LIDAM/LFIN), UCLouvain*
[2] *Département des sciences de la décision, HEC Montréal*

Abstract

Despite its role in the global financial crisis, collateralized loan obligation (CLO) remains a powerful tool to direct funds towards the real economy. In particular, it enables development banks to increase credit supply to SMEs. Public financial institutions thus face the challenge of identifying a subset of credits to be pooled in a CLO for the sake of reaching a specific financial target. This is a mixed-integer nonlinear program, known to be NP-hard. In this paper, we provide an efficient method to tackle this problem by relying on the large pool approximation combined with clustering and linearization of ancillary variables. As illustration, we consider two objective functions. We rely on the celebrated one-factor Gaussian copula in the main examples, but make clear that this assumption is not a restriction and can be relaxed. Our results contribute to reduce the funding cost of SMEs and are of direct interest for securitization stakeholders such as public financial institutions, commercial banks and pension funds.

1 Introduction

Small and Medium Enterprises (SMEs) play a key role in the economy by contributing to job creation and global economic development. In the European Union alone, there are 24.7 million SMEs, representing 99.8% of all non-financial businesses, accounting for 64.4% of all employment, and 51.8% of the value added in the non-financial business economy (European Commission, 2023). SMEs are typically much riskier than larger firms because of their limited size and financial capacity, which makes them less resilient to economic shocks (OECD, 2023). According to a recent report of the European Commission, "firms across the EU have experienced an increase in bankruptcies since the second half of 2022, which is largely due to regulatory changes (...), the delayed effect of the pandemic, a greater prevalence of late payments and increased difficulties in accessing finance (...)." The latter is pointed as one of the main issues: "In 2022, the share of SMEs not applying for bank loans due to fear of possible rejection rose to 6.6%, up from 5.4% in 2021 (with small firms being most impacted) – whereas the corresponding share among large firms decreased" (European Commission, 2023). Although the portion of SME loans to total business loans has increased to

*This paper builds upon the master thesis (Jonard, 2021) supervised by F. Vrins, which received the second prize of the 2021 CFA Quant award held in Amsterdam.

reach 44% in the recent years (OECD, 2023), about half of the SMEs have unmet credit needs (Owens and Wilhelm, 2017). This mismatch between credit supply and demand can be explained by a reduced risk tolerance of private banks but, to some extent, also results from decisions taken by central authorities themselves such as changes in bankruptcy regulation, more stringent capital requirements, or rise of interest rates to fight inflation, to name but a few.

The question of SMEs' access and cost of funds is thus a primary concern for policymakers. The European Commission has launched several financing programmes aiming to reduce both interest rates and collateral requirements, and to increase financing volumes. Many of these programmes are handled by the European Investment Bank via the European Investment Fund thanks to securitization, and Collateralized Loan Obligation (CLO) in particular. In a nutshell, the idea is that a public financial institution (PFI) such as the European Investment Fund purchases loans to commercial banks and turns them into tradeable securities which can then be sold on financial markets. Although securitization is sometimes perceived as a *weapon of wealth destruction* since the global financial crisis, it is also considered by many stakeholders as *the greatest financial innovation of the past century*. In our application, securitization of SME loans is appealing, with four positive outcomes. First, it allows one to transfer risk to external investors. Commercial banks can use the released capital to issue new loans, thereby increasing the amount of private money injected in the economy. Second, these securities form a new class of financial products, allowing investors such as pension funds to better diversify their portfolios. Third, the PFI can focus on specific types of loans and boost the development of high-priority sectors (e.g., cybersecurity, sustainalbe activities, renewable energy, etc). Finally, focusing on benefits for SMEs, the securitized assets benefit from a credit enhancement mechanism (via CLO tranches) that will reduce the risk premium required by external investors. This, in turn, leads to a substantial decrease of SMEs' funding costs while putting only a limited amount of PFI's public money at risk. Therefore, securitization provides a unique opportunity for governments willing to support SMEs.

In this context, the problem of optimal loans selection consists in assigning, to each candidate loan among a couple of thousands, a selection indicator specifying whether the loan needs to be securitized in a CLO designed to achieve a specific financial goal. Mathematically, this amounts to determine the entries of a binary vector for the sake of optimizing an objective function, leading to a high-dimensional nonlinear mixed-integer program known to be NP-hard. In addition, evaluating the objective function can be very costly, as it often features the joint distribution of the losses, hence, requires time consuming numerical methods such as Monte Carlo sampling. This calls for efficient alternatives that would provide a good solution in a reasonable computational time. The main contribution of this paper is to propose a fast procedure to find the solution to a simplified form of the initial loan selection problem. The considered approximations enable us to circumvent the most challenging aspects of the problem by decreasing its dimension, relaxing the binary constraints and tackling the nonlinearity issues. If necessary, this solution can be plugged in as an initial guess to further enhance the solution when considering a more sophisticated credit loss model.

The first attempt to address the problem of a credit portfolio construction from a quantitative perspective is due to Bennett (1984). The author argues that the main concepts inherent to Modern portfolio theory also apply to loans selection when constructing credit portfolios. Dependence between losses, in particular, is of primary importance. Since then, most of the developments dealt with the modeling and computational aspects of the loss on a pool of loans. For instance, extensive research has been conducted to understand the limit behavior of portfolio losses, leading to the well-known large pool approximations; see e.g. Vasicek (1991) for homogeneous pools and Gordy (2003) for the extension to heterogeneous loans, which form the corner stone of Basel's IRB regulation (BCBS, 2023). Corrections for finite-size, granularity or multifactor adjustments have been proposed (Pykhtin, 2004; Laurent and Sestier, 2016; Düllmann and Masschelein, 2007; Gordy and Lutkebohmert, 2013). Similarly, efficient computational methods for estimating loss distributions or quantiles of finite pools have been studied, including Laplace transforms, saddle point or advanced Monte Carlo approximations; see Gordy (2002); Glasserman and Ruiz-Mata (2006); Egloff and Leippold (2010), among others.

Some authors focus on the optimization of the waterfall design in the securitization context (Huang et al., 2007). Surprisingly, the literature on asset selection for securitization is relatively scarce. Some papers study the problem of designing optimal synthetic Collateralized Debt Obligations (CDOs), whose underlying assets are not loans but credit default swaps (CDSs). In this problem, the decision variables represent the relative notional amounts of each CDS – called *investment weights* hereafter – and are modeled as continuous variables in $[0, 1]$ defined on the corresponding simplex. For instance, Jewan et al. (2009) consider a multiobjective optimisation problem which consists in determining the investment weights associated with each CDS listed in the iTraxx Europe S5 series. Their objective is to minimize the tail conditional expectation of the CDO while maximizing the spread of a given tranche. This problem is solved under various constraints using a genetic algorithm and the objective function is evaluated via Monte Carlo. Similarly, Veremyev et al. (2012) consider three optimization problems for designing synthetic CDOs, where the decision variables are the attachment and detachment points of tranches, sometimes combined with the investment weights. The authors solve the problems at hand using the S&P CDO Evaluator and the Portfolio Safeguard solver of AORDA. Bo and Capponi (2014) consider a dynamic programming approach to optimally invest in a money-market account and in a portfolio of CDSs under contagion risk. These papers depart from the problem at hand in the sense that they do not feature integer constraints, short-selling might be permitted, the numerical applications focus on much smaller pools (of size 50, 53 and 2, respectively) and seem hardly scalable to face typical CLO sizes (tens of thousands of loans).

Westerfeld and Weber (2009) tackle the problem of optimal loan selection for CLO's. Their objective function takes the form of a weighted sum of the loans' risk contribution where the weights are the membership indicators and the risk measure is the unexpected loss contribution of the loan, that is, the variance of the loan's loss scaled by its exposure. They propose a heuristic exploiting the property that the objective function only relies on the marginal characteristics of the loans and not on their joint distribution, thereby avoiding the need to assume a portfolio loss model. Closest to our work is Saunders et al. (2007) who determine the optimal investment weights associated with a given number of buckets made of homogeneous loans for the sake of minimizing a portfolio's risk measure (VaR or CVaR). In this setup, the objective function takes the form of a sum of buckets' contributions scaled by the investment weights, leading to a linear program whose dimension is the number of buckets. The individual contribution of each bucket is evaluated thanks to the loss distribution provided by the large homogeneous pool assumption associated with a single risk factor model. This approach relies on two key features. First, there must be a single systematic factor and the objective function must display the portfolio invariance property, i.e., can be decomposed as the sum of the buckets' contribution. This property is however not met by the objective functions considered in this paper. Second, it is assumed that there are infinitely many homogeneous loans in each bucket (rating class), such that the decision variables are investment weights rather than membership indicators. Sirignano et al. (2016) consider a pool of finite size and face the challenging aspects related to the number of loans and the integer constraint by relying on an approximation of the initial problem at hand. They consider a broad class of objective functions and a rather general discrete factor model whereby each loan is characterized by a set of static features, some time-varying credit state variables, and a common process driving the correlation between the loans' states. They propose an approximate version of the initial optimization program which relies on the asymptotic (large pool) law of the portfolios for this class of models. They discretize the feature space using a sparse grid, such that multiple loans become equivalent to others. Their approach can accommodate objective functions taking the form of a function of the expectation of transformed returns. Their optimization program searches for a distribution over the space of probability measures whose dimension is the number of loans to be selected, assumed to be given. We are in a different setup since, in our case, this number is an output of the optimization program. In addition, we want to address objective functions possibly capturing other portfolio characteristics, such as the weighted average life.

We contribute to the field by proposing a technique that delivers a fast solution to the loan selection problem featuring a nonlinear objective function and given a large but finite pool of loans to select from, without imposing the number of loans to be fixed. First, we perform a linearization of the ancillary variables underlying the initial program. As in Saunders et al. (2007), we model the loss on the pool using the large pool approximation under a factor model. However, we do not rely on the asymptotic loss distribution of homogeneous pools, and consider a broader class of objective functions which need not exhibit portfolio invariance. This allows us to tackle more sophisticated objectives (e.g., featuring the expected loss on the tranche or its capital requirements) and factor models (e.g., the t-copula, which combines two systematic factors). A clustering algorithm allows us to reduce the dimension. We first group the loans into clusters and treat them as "prototypical loans". Then, we determine the optimal amount to invest while relaxing the binary constraint. This can be related to the rating buckets and sparse grid approaches of Saunders et al. (2007) and Sirignano et al. (2016). Next, we compute explicitly the selection indicators by projecting the solution on the set of conformable vectors with binary entries. Although these approximations can be considered separately, combining linearization and clustering delivers promising results. We apply our approach on two objective functions which aim to be potential targets that a structurer may want to consider, namely, (i) a rating defined as a function of the expected loss and the weighted average life of a CLO tranche and (ii) the amount of capital that is released. Our simulation studies show that the proposed approach provides faster and better solutions than those found by applying standard algorithms able to tackle nonlinear mixed-integer programs.

The paper is organized as follows. The principles of securitization are recalled in Section 2. Section 3 gives a summary of the main financial variables that may be of interest for the structurer and proposed "standard" modeling choices for each of them as illustration. We describe a general form for the optimization problem in Section 4 and discuss several approximations to efficiently solve the problem. Section 5 presents two financial applications that fit the proposed general problem in the One-Factor Gaussian Copula model. Section 6 presents an empirical analysis and discuss the results. Finally, Section 7 explains how this approach can be extended to other loss models, based on multiple factors and/or other copulae.

2 Securitization

We consider the problem of an originator willing to design a CLO starting from an initial pool $\mathcal{S} = \{1, 2, \ldots, I\}$ of fixed-rate defaultable loans taken from credit portfolios collected from issuers such as banks or other credit institutions. The securitization process consists in two main steps: (i) analyzing the loans in $\mathcal{S}$ and (ii) designing the waterfall (that is, the algorithm that will allocate the cashflows across tranches) on a portfolio $\mathcal{P}$ of loans selected from $\mathcal{S}$ that will be embedded in the structure. This second step is tackled in order for the CLO to display specific features in terms of risk-return trade off, rating or capital requirements. In this paper, we focus on a standard sequential architecture featuring J tranches given exogenously by a structurer. The tranches bear losses in reverse seniority order; a tranche $j \in \{1, 2, \ldots J\}$ will incur principal losses if and only if the notional of the more junior tranches has been fully wiped out. Each tranche j is characterized by a pair of attachment and detachment points (A_j, D_j) satisfying $0 \leq A_j < D_j \leq 1$. The J tranches form a partition of the $[0, 1]$ interval in the sense that $A_1 = 0$, $D_J = 1$ and $A_{j+1} = D_j$. Assuming the waterfall to be given, it remains to identify the portfolio $\mathcal{P}$ of loans so as to reach the structurer's objectives. Mathematically, $\mathcal{P}$ can be represented as the vector of IDs of the loans being selected, $\mathcal{P} \subseteq \mathcal{S}$, or via the *membership vector* $\boldsymbol{w} = (w_1, \ldots, w_I)$ where $w_i := \mathbb{1}_{\{i \in \mathcal{P}\}}$ is a binary weight indicating whether loan i is embedded in the securitization or not. The membership vector is found by solving an optimization problem. We assume that the loans are purchased from the issuers such that the lifetime of $\mathcal{P}$ corresponds to the maturity of the longest loan.

2.1 Portfolio of defaultable loans

The loss associated with the default of loan $i \in \mathcal{S}$ with maturity M_i can be decomposed as

$$L_i := N_i \times \lambda_i \times B_i \ .$$

In this expression, $N_i > 0$ stands for the outstanding notional, $\lambda_i \in [0, 1]$ is the loss given default (LGD) defined as the ratio of N_i that is lost upon default prior to the maturity M_i, and $B_i := \mathbb{1}_{\{\tau_i \leq M_i\}}$ is the associated default indicator with τ_i the default time of the loan i.[1]

The credit loss (relative to its notional) on a portfolio $\mathcal{P}$ with membership vector $\boldsymbol{w}$ reads as

$$L(\boldsymbol{w}) := \frac{\sum_{i=1}^{I} w_i L_i}{N(\boldsymbol{w})} \quad \text{where} \quad N(\boldsymbol{w}) := \sum_{i=1}^{I} w_i N_i$$

is the pool's outstanding notional at inception. Because the membership vector $\boldsymbol{w}$ belongs to the set $\mathbb{B}^I$ of I-dimensional bit arrays, the size of the portfolio $\mathcal{P}$ is just its Hamming weight: $|\,\boldsymbol{w}\,| := \sum_{i=1}^{I} w_i$. The relative loss is thus 0 if no loan in $\mathcal{P}$ default prior to the pool maturity

[1] Another common decomposition is to replace N_i by EAD_i, the exposure at default of loan i. The latter is typically smaller than the former in presence of amortization, hence, considering N_i instead of EAD_i is a conservative approach. However, note that this difference does not really matter in our setup because it simply amounts to redefine the LGD factor.

$\max_{i\in\mathcal{P}}(M_i)$. On the other hand, the loss is $\frac{1}{N(\boldsymbol{w})}\sum_{i=1}^{I} w_i\lambda_i N_i \leq 1$ if all the loans in $\mathcal{P}$ default before T, where the equality holds only if all the LGDs are equal to one.

The expected loss (EL) on $\mathcal{P}$ quantifies the average loss that an investor will face when holding the pool up to T:

$$\mathrm{EL}(\boldsymbol{w}) := \mathbb{E}(L(\boldsymbol{w})) = \frac{\sum_{i=1}^{I} w_i \mathbb{E}\left(L_i\right)}{N(\boldsymbol{w})} .$$

In the special case where the LGDs are considered as constant, the EL on loan i reads as[2]

$$\mathbb{E}\left(L_i\right) = N_i\lambda_i p_i \quad \text{with} \quad p_i := \mathbb{E}(B_i) = \mathbb{P}(\tau_i \leq M_i) .$$

2.2 Tranches of a portfolio

The size of tranche j relative to $N(\boldsymbol{w})$ is $S_j := D_j - A_j > 0$, satisfying $\sum_{j=1}^{J} S_j = 1$. The loss on the tranche relative to its outstanding notional $N_j(\boldsymbol{w}) = S_j \times N(\boldsymbol{w})$ becomes:

$$L_j(\boldsymbol{w}) := \left[\frac{L(\boldsymbol{w}) - A_j}{S_j}\right] \quad \text{where} \quad [x] := \min(1; \max(0; x)) .$$

Observe that $L_j(\boldsymbol{w}) = 0$ if and only if $L(\boldsymbol{w}) \leq A_j$ and $L_j(\boldsymbol{w}) = 1$ if and only if $L(\boldsymbol{w}) \geq D_j$. Note the distinction between L_j which represents the loss on loan j in the pool $\mathcal{S}$, and $L_j(\boldsymbol{w})$, the loss on the j-th tranche of $\mathcal{P}$ with membership vector $\boldsymbol{w}$, and similarly for N_j *vs.* $N_j(\boldsymbol{w})$. The sum of the losses on the J tranches is equal to the total loss experienced by the portfolio: $\sum_{j=1}^{J} L_j(\boldsymbol{w})N_j(\boldsymbol{w}) = L(\boldsymbol{w})N(\boldsymbol{w})$.

Similar to the EL on a portfolio $\mathcal{P}$, one can compute the EL on a tranche j of $\mathcal{P}$, defined as the expected value of the loss on the tranche relative to its notional:

$$\mathrm{EL}_j(\boldsymbol{w}) := \mathbb{E}(L_j(\boldsymbol{w})) = \mathbb{E}\Big(\left[\frac{L(\boldsymbol{w}) - A_j}{S_j}\right] \Big) .$$

Because of the nonlinear min-max operator $x \mapsto [x]$, computing EL_j requires the distribution of $L(\boldsymbol{w})$. There are two classes of models. Top-down models aim to directly model the total losses on the credit portfolio $L(\boldsymbol{w})$ as a single variable (Giesecke et al., 2011). Credit risk plus, for instance, postulates a model for the number of defaults in the portfolio by assuming for the latter a binomial distribution with random (gamma-distributed) mean. Bottom-up models, instead, start from the marginal distributions of the individual losses and postulate some copula to obtain a joint distribution, from which the distribution of $L(\boldsymbol{w})$ can be inferred. This is the setup adopted in the Basel regulation, which assumes a One-Factor Gaussian Copula framework to couple the default times (see Section 3.1). For a comparison between top-down and bottom up approaches, we refer to Giesecke (2008). We focus on bottom-up models in the sequel. Besides modeling the individual characteristics of the loans (N_i, λ_i and p_i) one thus needs to assume a specific dependence structure between the L_i's, hence, between the B_i's. Factor models will be discussed in sections 3 and 7.

[2] Considering the LGDs as known constants may seem unrealistic but our point in this section is to sketch the salient features of the securitization process and analyze the resulting optimization problem. In practice, one could consider variants of this setup featuring stochastic LGDs; see Section 7.

3 Main financial variables

The features of the securitized products depend both on the waterfall and on the loans embedded in the CLO, that is, on the characteristics of the portfolio $\mathcal{P}$. In modern portfolio theory, the objective of the portfolio manager is to identify the variables that matter to the investors (e.g., the expected return and variance in Markowitz's theory), build an objective function out of them (quadratic expected utility) that would depict the investor's preference and solve an optimization problem to determine how to invest in the assets of a given universe. Similarly, the purpose of the structurer is to identify the variables that matter in the design of the product, build an objective function trading off between those and solve for the membership vector $\boldsymbol{w}$. It is not the purpose of the paper to identify the most relevant objective function a structurer may want to consider, as many different choices can be relevant. Instead, we discuss some standard indicators extensively used in asset-backed securities (ABS) such as the expected loss on the tranche discussed above, the weighted average coupon, the weighted average maturity, the weighted average life and the capital requirements. Two possible objective functions built upon these variables will be considered in Section 5 for illustration purposes.

3.1 Expected loss on the tranche

The Expected loss on a tranche (ELoT) is perhaps the most important risk factor of a tranche. It is driven by the joint behavior of the default indicators B_i's. In line with the Basel model used to compute capital requirements discussed in Section 3.2, specific attention is given to the One-Factor Gaussian Copula (1FGC) approach, which is used for illustration purposes throughout the text. Alternative choices are possible, as discussed in Section 7. This formulation aims to provide the unexpected loss by relying on asymptotic results related to Large Pool (LP) assumptions.

In the one-factor model setup, the default event of loan i is modeled indirectly using latent variables Z_i's taking values below a given threshold z_i:

$$\{\tau_i \leq M_i\} \quad \Leftrightarrow \quad \{B_i = 1\} \quad \Leftrightarrow \quad \{Z_i \leq z_i\}\ .$$

Letting F_X denote the (invertible) cumulative distribution function of a continuous random variable X, setting $z_i := F_{Z_i}^{-1}(p_i)$ guarantees that $\mathbb{P}(Z_i \leq z_i) = p_i$, hence, allows one to preserve the marginal default probability (PD) of the loans. The dependence structure between the default indicators is handled by decomposing the Z_i's as a weighted sum of two standard i.i.d. variables: a common (systematic) random variable Z and a firm-specific (idiosyncratic) random variable ϵ_i,

$$Z_i = a_i Z + \sqrt{1-a_i^2}\,\epsilon_i \quad \text{where} \quad Z, \epsilon_1, \ldots, \epsilon_I \ \text{ are independent and standardized}\ . \tag{1}$$

The loading factors $a_i \in (-1, 1)$ control the correlation between the credit worthiness variables Z_i's, hence, between the default indicators B_i's. In such a model, the default events are independent given the systematic factor Z and the conditional probability of default of firm i is noted

$$p_i(Z) := \mathbb{P}(Z_i \leq z_i | Z) = F_{\epsilon_i}\left(\frac{z_i - a_i Z}{\sqrt{1-a_i^2}}\right)\ . \tag{2}$$

The ELoT can be found by relying on the LP approximation (Gordy, 2003). If the portfolio is infinitely granular in the sense that it has a nearly infinite number of loans with infinitesimal exposures, the impact of the idiosyncratic variables can be completely diversified away and the portfolio loss converges in law to its conditional expectation:

$$L(\boldsymbol{w}) \quad \overset{I\to\infty}{\longrightarrow} \quad L(\boldsymbol{w};Z) := \mathbb{E}(L(\boldsymbol{w})|Z). \tag{3}$$

In other words, the randomness embedded in $L(\boldsymbol{w}) = L(\boldsymbol{w}; Z, \boldsymbol{\epsilon})$ results solely from Z, asymptotically: in the limit where $I \to \infty$, the idiosyncratic risks $\boldsymbol{\epsilon}$ are fully diversified away. The asymptotic ELoT on tranche j is obtained replacing $L(\boldsymbol{w})$ by its asymptotic version $L(\boldsymbol{w}; Z)$:

$$L_j(\boldsymbol{w}) = \left[\frac{L(\boldsymbol{w}) - A_j}{S_j}\right] \quad \overset{I\to\infty}{\longrightarrow} \quad L_j(\boldsymbol{w}; Z) := \left[\frac{L(\boldsymbol{w}; Z) - A_j}{S_j}\right]. \tag{4}$$

We conclude from the tower law that, for I large enough, the ELoT on tranche j can be estimated from its LP approximation as an integral of the asymptotic EL with respect to the density of Z:

$$\mathrm{EL}_j(\boldsymbol{w}) = \mathbb{E}(L_j(\boldsymbol{w})) = \mathbb{E}(\mathbb{E}(L_j(\boldsymbol{w})|Z)) \approx \mathbb{E}(L_j(\boldsymbol{w}; Z)) = \int_{-\infty}^{\infty} \left[\frac{L(\boldsymbol{w}; z) - A_j}{S_j}\right] p_Z(z)dz \ , \tag{5}$$

where, in this expression, $p_Z = F_Z'$ and the conditional asymptotic expected loss on the portfolio is

$$L(\boldsymbol{w}; z) = \frac{1}{N(\boldsymbol{w})} \sum_i w_i N_i \lambda_i p_i(z) \ . \tag{6}$$

Remark 1. *The 1FGC model corresponds to the special case where* $Z, \epsilon_1, \dots, \epsilon_I \sim \mathcal{N}(0,1)$*. This is convenient because then* $Z_i \sim \mathcal{N}(0,1)$ *for all* i *and any choice of the loadings. Letting* Φ *denote the cumulative distribution of the standard normal distribution,* $\phi = \Phi'$ *its density,* $z_i = \Phi^{-1}(p_i)$ *its* p_i*-quantile and setting the loadings* $a_i = \sqrt{\rho_i}$ *for* $\rho_i \in [0,1)$ *yields,*

$$p_i(Z) = \Phi\left(\frac{\Phi^{-1}(p_i) - \sqrt{\rho_i}Z}{\sqrt{1-\rho_i}}\right). \tag{7}$$

The Gaussian coupling is very popular because of its use in Basel regulation. Yet, it suffers from well-known drawbacks such as the lack of asymptotic tail dependency. Nevertheless, this assumption can be easily relaxed; what matters for the LP trick to work is the independence of the underlying variables $Z, \epsilon_1, \dots, \epsilon_I$*. Changing the distribution of the latter only changes the expression of the conditional probabilities* $p_i(z)$ *involved in (6), taken a different form as (7). It is important to note that the class of one-factor models such as the 1FGC is just an example of loss models our approach can deal with. In essence, our method encompasses any model to which the LP approximation applies such as the t-copula or multi-factors models, including some stochastic LGD extensions. Examples of families of models that can be accommodated are provided in Section 7.*

3.2 Capital requirements

In Basel regulation, capital requirements correspond to the unexpected losses computed under the 1FGC model. It is the difference between the value-at-risk (at 99.9% confidence level) and the expected loss (5) when relying on (6) and (7). The corresponding provisions add up,[3] that is, the capital $K(\boldsymbol{w})$ associated with holding $\mathcal{P}$ is the sum of the capital requirements associated with the underlying loans:

$$K(\boldsymbol{w}) = \frac{\sum_{i=1}^{I} w_i K_i N_i}{N(\boldsymbol{w})} , \tag{8}$$

where K_i is the capital requirement of loan i per unit of outstanding exposure. Capital requirements at the loan level are governed by the Basel guidelines and salient points associated with their computation are recalled in Appendix A.

The capital requirement $K_j(\boldsymbol{w})$ for the tranche j (relative to its notional) is a regulatory variable that is defined by Basel guidelines. Leveraging on the operator [.], its expression in the official documents (BCBS, 2019) simplifies to:

$$K_j(\boldsymbol{w}) := \delta_j(\boldsymbol{w}) + (1 - \delta_j(\boldsymbol{w}))K_j^{\text{CLO}}(\boldsymbol{w}) \quad \text{where} \quad \delta_j(\boldsymbol{w}) := \left[\frac{K(\boldsymbol{w}) - A_j}{S_j}\right] \tag{9}$$

and

$$K_j^{\text{CLO}}(\boldsymbol{w}) := \frac{e^{a(\boldsymbol{w})u_j(\boldsymbol{w})} - e^{a(\boldsymbol{w})l_j(\boldsymbol{w})}}{a(u_j(\boldsymbol{w}) - l_j(\boldsymbol{w}))} , \quad \text{with}$$

$$a(\boldsymbol{w}) := \frac{-1}{K(\boldsymbol{w})} , \quad u_j(\boldsymbol{w}) := D_j - K(\boldsymbol{w}) \quad \text{and} \quad l_j(\boldsymbol{w}) := \max(A_j - K(\boldsymbol{w}); 0) .$$

Again, note the difference between K_j, the capital associated with loan j in the pool $\mathcal{S}$, and $K_j(\boldsymbol{w})$, which represents the capital requirement of tranche j on the portfolio $\mathcal{P} \subseteq \mathcal{S}$. Because $K_j(\boldsymbol{w})$ depends on $\boldsymbol{w}$ via $K(\boldsymbol{w})$ only, we introduce for further reference the notation $k_j(\cdot)$:

$$K_j(\boldsymbol{w}) = k_j(K(\boldsymbol{w})) . \tag{10}$$

The capital released by selling tranche j is given by the difference of the capital on the portfolio $\mathcal{P}$ and the capital that one must hold when keeping all the tranches of $\mathcal{P}$ but the j-th one:

$$\Delta K_j(\boldsymbol{w}) = K(\boldsymbol{w}) - \sum_{k=1}^{J} \mathbb{1}_{\{k \neq j\}} S_k K_k(\boldsymbol{w}) > 0 . \tag{11}$$

Remark 2. *In general, $\Delta K_j(\boldsymbol{w}) \neq S_j K_j(\boldsymbol{w})$ because the portfolio invariance property holds for loans but not for tranches: the capital requirements of a tranched structure exceeds the capital on the portfolio in the sense that $\sum_{j=1}^{J} S_j K_j(\boldsymbol{w}) > K(\boldsymbol{w})$ when $J > 1$.*[4] *However, the total capital on the structure $\sum_{j=1}^{J} S_j K_j(\boldsymbol{w})$ does not depend on the number of tranches nor on the choice of the attachment/detachment points.*

3.3 Weighted average maturity, coupon and life

Besides credit risk, investors typically also care about the maturity and the coupon of the product they invest in. However, a portfolio is often made or thousands of loans, each having its own

[3] This property is called *portfolio invariance*.

[4] In other words, keeping the portfolio $\mathcal{P}$ is less costly in terms of capital that keeping all the J tranches of that portfolio when $J > 1$. This seems inconsistent, but is nonetheless inherent to the Basel framework. We presume that this is because it is practically irrelevant: there is no incentive in tranching a portfolio if all the tranches are kept.

maturity M_i and rate r_i. To circumvent this problem, investors in the ABS market rely on meta-variables that describe the aggregated behavior of the portfolio as a whole. The weighted average maturity (WAM) and weighted average coupon (WAC) provide the "portfolio equivalents" of the loans' maturity and coupon rate. The corresponding expressions (defined relative to the pool notional for consistency with the earlier sections) are given by the average of the maturities and coupons of the loans embedded in $\mathcal{P}$, weighted by the relative notional $N_i/N(\boldsymbol{w})$, see e.g. Veronesi (2010) or Petitt et al. (2015):

$$\mathrm{WAC}(\boldsymbol{w}) := \frac{1}{N(\boldsymbol{w})}\sum_{i=1}^{I} w_i N_i r_i \quad \text{and} \quad \mathrm{WAM}(\boldsymbol{w}) := \frac{1}{N(\boldsymbol{w})}\sum_{i=1}^{I} w_i N_i M_i \,. \tag{12}$$

A crucial risk variable in fixed income analysis is *duration*, aiming to capture the sensitivity of the value of a security to interest risk. A similar (although slightly different) concept in ABS is the weighted average life (WAL), aiming to represent the expected amount of time needed to repay one unit of principal. Similarly to the ELoT, a detailed calculation of the weighted average life of a tranche (WALoT) would require a stochastic model and a description of the waterfall driving the amortization profile. Nevertheless, the goal of the WAL is merely to give an indication of the "average lifetime" of the portfolio rather than to be an accurate estimation of interest-rate risk, as illustrated by the market conventions and the lack of academic literature on the topic.

In the ABS industry, the WAL of $\mathcal{P}$ is practically defined as the sum of the payment times t_i weighted by the part of the cashflow paid at t_i that relates to capital. Assuming we are at time $t = 0$ and $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$ is the set of payment dates of the cashflows coming from the loans:

$$\mathrm{WAL}(\boldsymbol{w}) = \frac{\sum_{t_i\in\mathcal{T}} t_i \times p(t_i;\boldsymbol{w})}{\sum_{t_i\in\mathcal{T}} p(t_i;\boldsymbol{w})} \,,$$

where $p(t;\boldsymbol{w})$ is the amount of capital repayment paid at time t in the portfolio $\mathcal{P}$. The function $p(t;\boldsymbol{w})$ depends on the loan characteristics such as the coupon rates, the maturities and the possible prepaid amounts.[5]

In practice, the WAL of a tranche is often modeled using a top-down approach where the portfolio is taken as a single loan whose "maturity" and "coupon rate" can be regarded as $T = \mathrm{WAM}(\boldsymbol{w})$ and $r = \mathrm{WAC}(\boldsymbol{w})$. In this setup, the WAL on a portfolio $\mathcal{P}$ depends on $\boldsymbol{w}, r, T$ and a parameter (possibly functional) γ called *prepayment rate* which controls the rate at which the underlying principal is repaid ahead of schedule. Prepayment is a phenomenon that depends on macro-economic environment (such as interest rates) and private events impacting the customer's life (divorce, moves, fire sales, death, etc), and is therefore difficult to model. Standard market practice is to describe prepayment using the PSA model which provides a specific shape for the monthly prepayment rate (CPR) profile that can be adjusted to the portfolio dynamics by tuning a parameter α called "PSA level". Specifically, the CPR profile grows linearly with the month index i until the 30th month and

[5]Note that the WAL is computed by assuming that the capital payment cashflows are deterministic which is contradictory with the existence of credit risk. This can be seen as a convention. The WAL aims to convey information about other features of $\mathcal{P}$. In practice, the lifetime of a portfolio is impacted by the early termination related to the defaults, which can be considered as a form of prepayment. Therefore, the impact of credit risk on the actual lifetime of a pool could be captured by playing with the prepayment rate.

is constant afterwards:[6]

$$\text{CPR}(i) = \alpha \times 6\% \times \min\left(1; \frac{i}{30}\right) .$$

A 100% PSA pool corresponds to $\alpha = 1$. For more information about PSA we refer to (Veronesi, 2010) which provides a very comprehensive treatment of ABS notions.

Recall that the focus of this paper is on the optimization problem related to securitization designs, not to propose a precise model for a particular waterfall. Therefore, we model the WAL using a schematic continuous-time prepayment framework inspired from the discrete-time PSA model leading to a semi-closed form solution. Specifically, we assume that

$$\text{WAL}(\boldsymbol{w}) = \frac{\int_0^T t \times p(t;\boldsymbol{w})dt}{\int_0^T p(t;\boldsymbol{w})dt}$$

where $T = \text{WAM}(\boldsymbol{w})$ is the WAM of $\mathcal{P}$. The dynamics of p depend on the principal repayment, on the continuously compounded version $\tilde{r} = \ln(1 + r)$ of the WAC coupon rate r (which controls the amortization profile, hence, the stream of so-called *scheduled principal payments*) as well as on the profile of the portfolio's prepayment rate $\gamma(\cdot)$ (stream of *unscheduled principal payments*). Even in the case of a homogeneous pool ($r_i = r$ and $M_i = M$), the portfolio coupon (that is, the cashflow paid) at time t depends on t whenever $\gamma(t) > 0$. This is because the cashflow related to principal payment at t increases in case of prepayment such that, afterwards, the principal payments drop.

We show in Appendix C that, for a tranche j and assuming that $p(t;\boldsymbol{w})$ represents the principal cahsflows relative to the notional of the pool, the WAL on a tranche j can be modeled as:

$$\text{WAL}_j(\boldsymbol{w}) := \frac{\int_{T_{j-1}}^{T_j} t \times (c(t) + (\gamma(t) - \tilde{r})n(t))\, dt}{S_j} , \tag{13}$$

where

$$c(t) := \frac{\tilde{r}\, \mathrm{e}^{-\int_0^t \gamma(u)du}}{1 - e^{-\tilde{r}T}} \qquad \text{and} \qquad n(t) := 1 + c(t)e^{\tilde{r}t} \int_0^t \frac{\tilde{r} - \gamma(u) - c(u)}{c(u)e^{\tilde{r}u}} du \tag{14}$$

are the coupon per unit of time and outstanding notional of $\mathcal{P}$ relative to $N(\boldsymbol{w})$ at time t, respectively, T_j is the time at which tranche j is fully repaid and the denominator comes from the fact that the sum of the principal payment on tranche j relative to the notional of $\mathcal{P}$ is equal to $N_j/N(\boldsymbol{w}) = S_j$. We consider specific profiles for the prepayment rate function $\gamma(t)$ inspired from the PSA model and show that, in such cases, the times T_j are easily found by computing the root of monotonic functions. Note that since the WAL of a tranche is a function of r, T and γ,

$$\text{WAL}_j(\boldsymbol{w}) = \text{wal}_j(\text{WAC}(\boldsymbol{w}), \text{WAM}(\boldsymbol{w})) \tag{15}$$

for some function $\text{wal}_j(\cdot,\cdot)$ because $\gamma(\cdot)$ is a function of $\text{WAC}(\boldsymbol{w})$ and $\text{WAM}(\boldsymbol{w})$, too.

[6] A potential justification for this profile is to notice that people take time before thinking of refinancing their loans as the market conditions or their private situations need to evolve in order for prepayment to make sense.

4 Optimization problems

Having given an overview of the main financial variables driving the risks associated with credit portfolio tranches and some corresponding models to evaluate them, we proceed with the analysis of the related optimization program and derive some alternatives that help circumventing the computational issue resulting from its NP-hard nature. Two parametric forms of objective functions that a structurer may want to minimize and that will be considered in our numerical experiments are introduced in Section 5.

4.1 Original problem

The goal of the structurer is to find the vector $\boldsymbol{w}$ solving:

Program 1

$$\min_{\boldsymbol{w}\in\mathbb{B}^I} F(\boldsymbol{w}) \quad \text{s.t.} \quad \begin{cases} \boldsymbol{w} & \in \mathcal{W} \\ N(\boldsymbol{w}) & \geq \underline{N} \end{cases}$$

where $\underline{N} > 0$ is a strictly positive finite lower bound on the pool's notional and $\mathcal{W}$ represents a set of optional linear constraints involved in the securitization, e.g., aiming to foster diversification across regions or economic sectors. The function $F(\boldsymbol{w})$ represents the preferences of the originator and is typically nonlinear in the main financial variables. This is a knapsack problem (Wolsey, 2020): Program 1 is a nonlinear mixed integer program (NLMIP) which, moreover, is typically high-dimensional (the number of loans is often between 1,000 and 100,000). This problem is NP-hard and using a nonlinear mixed integer solver (such as a genetic algorithm) straight away for Program 1 leads to a sub-optimal solution in a large computation time. It is interesting to reshape the problem because finding efficiently the optimal solution to an approximate problem might be better than tackling the exact problem using brute force. This is the purpose of the next section.

4.2 Approximate optimization problems

The main challenges associated with Program 1 are of two types. First, Program 1 is a high-dimensional problem whose optimization domain is discrete ($\boldsymbol{w} \in \mathbb{B}^I$ where I is large). Second, F is typically nonlinear (see Section 5 for examples of objective functions). In this section, we consider three approximations of Program 1 that mitigate these issues.

4.2.1 Clustering and projection

To tackle the first issue, we propose an approximation inspired by (Sirignano et al., 2016). The main idea is to notice that if two loans are very similar, choosing one or the other has a limited impact on the objective function F. One could thus group the I loans into Q clusters and, instead of assigning a binary weight w_i to each loan, we assign a nonnegative real number $\tilde{w}_q$ to each cluster $q \in \{1, 2, \ldots, Q\}$ representing the notional that we invest in each cluster relative to the maximum outstanding notional $\overline{N}_q$ available in the latter. Noting $q(i)$ the cluster of loan i, one simply gets $\overline{N}_q = \sum_{i=1}^{I} \mathbb{1}_{\{q(i)=q\}} N_i$. By doing so, the optimization domain of the sub-program changes from $\mathbb{B}^I$ to $\mathbb{R}_+^Q$ defined as the set of Q-dimensional vectors with nonnegative real entries. The benefits of doing this are twofold. First, the dimension is reduced from I to Q where $Q \ll I$, typically. Second, the integer constraint is relaxed. Of course, the solution vector $\tilde{\boldsymbol{w}} \in \mathbb{R}_+^Q$ of amounts (relative to the maximum outstanding notional) to be invested in each cluster will then need to be projected on the set of attainable portfolios to obtain $\boldsymbol{w} \in \mathbb{B}^I$. This is tackled in a second step.

Specifically, we first run a clustering algorithm on the pool $\mathcal{S}$. Next, we consider the pool $\tilde{\mathcal{S}}$ where each "prototypical loan" q has the characteristics of its centroid, $c(q)$, except that its "notional" is set to $\overline{N}_q$ and that fractional shares of such loans can now be purchased. This leads to the following program:

Program 2: (Clustered version of Program 1)

$$\min_{\tilde{\boldsymbol{w}} \in \mathbb{R}_+^Q} \quad \text{s.t.} \quad \begin{cases} \tilde{\boldsymbol{w}} & \leq \mathbf{1} \\ \tilde{\boldsymbol{w}} & \in \tilde{\mathcal{W}} \\ N(\boldsymbol{w}) & \geq \underline{N} \end{cases}$$

where $\mathbf{1} \in \mathbb{R}^Q$ is a vector of ones and $N(\tilde{\boldsymbol{w}}) := \sum_{q=1}^{Q} \tilde{w}_q \overline{N}_q$ is the total notional invested. Finally, we need to map the "optimal cluster weight vector" $\tilde{\boldsymbol{w}}^* \in \mathbb{R}_+^Q$ to a portfolio $\boldsymbol{w}^* \in \mathbb{B}^I$. To this end, we select the loans closest to their centroid in each cluster until the optimal proportion is reached.

Projection 2:

$$\boldsymbol{w}^* \in \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{B}^I} \sum_{i=1}^{I} w_i d_i \quad \text{s.t.} \quad \sum_{i=1}^{I} w_i N_i \, \mathbb{1}_{\{q(i)=q\}} \geq \tilde{w}_q^* \overline{N}_q \quad \forall q \in \{1, 2, \ldots, Q\} ,$$

where d_i is the distance of loan i from its centroid $c(q(i))$. By doing so, one obtains a portfolio $\boldsymbol{w}^* \in \mathbb{B}^I$ which allocates the wealth across clusters in a similar way as $\tilde{\boldsymbol{w}}^* \in \mathbb{R}_+^Q$, while being close to the prototypical loans forming $\tilde{\boldsymbol{w}}^*$. Compared to Program 1, Program 2 is much more tractable. It can be solved using a derivative-free algorithm and a Mixed-Integer Linear Program (MILP) for Projection 2. Moreover, the projection step is trivial because it simply consists in selecting, for each cluster q, the loans being closest from their centroid while a notional threshold is not reached.

4.2.2 Linearization

Another common procedure to address the complexity of optimization problems is to consider approximations exploiting the shape of the functions at hand such as linearization schemes. To this end, let us write the objective F as a function of n ancillary variables $X_0(\boldsymbol{w}), X_1(\boldsymbol{w}), \ldots, X_n(\boldsymbol{w})$ and propose, for each of them, a linear approximation $X_i(\boldsymbol{w}) \approx \boldsymbol{w}^\top \mathbf{c}_i =: \hat{X}_i(\boldsymbol{w})$ for $i \in \{0, 1, \ldots, n\}$ where $\mathbf{c}_i \in \mathbb{R}^I$. This yields an alternative version of F featuring the linearized ancillary variables:

$$F(\boldsymbol{w}) := f\big(X_0(\boldsymbol{w}), ..., X_n(\boldsymbol{w})\big) \approx f\left(\hat{X}_0(\boldsymbol{w}), ..., \hat{X}_n(\boldsymbol{w})\right) =: \hat{F}(\boldsymbol{w}) .$$

Note that we do not linearize F, only the ancillary variables X_i. In particular, f can be nonlinear. In our application, the ancillary variables could be the expected loss, the capital requirement or the weighted average life of the tranche. We assume that F is monotonic in at least one of them, and choose the convention that F is increasing in X_0. One can thus optimize F indirectly as follows:

Program 3 (Linear approximation of Program 1)

$$\min_{\hat{\boldsymbol{x}} \in \mathbb{R}^n} \quad g(\hat{\boldsymbol{x}}) := F(\boldsymbol{w}^*(\hat{\boldsymbol{x}}))$$

where

Sub-program 3

$$\boldsymbol{w}^*(\hat{\boldsymbol{x}}) \in \underset{\boldsymbol{w}\in\mathbb{B}^I}{\operatorname{argmin}} \ \hat{X}_0(\boldsymbol{w}) \quad \text{s.t.} \quad \begin{cases} \boldsymbol{w} & \in \mathcal{W} \\ N(\boldsymbol{w}) & \geq \underline{N} \\ \hat{X}_i(\boldsymbol{w}) & = \hat{x}_i \ , \quad i \in \{1, 2, \ldots, n\} \ . \end{cases}$$

Concretely, we seek to find the membership vector $\boldsymbol{w}^*$ minimizing the objective function $F(\boldsymbol{w})$ indirectly, via a function g of a vector $\hat{\boldsymbol{x}} = (\hat{x}_1, \ldots, \hat{x}_n)$ of scalar variables. The value $g(\hat{\boldsymbol{x}})$ coincides with the objective function F evaluated at the portfolio $\boldsymbol{w}^*(\hat{\boldsymbol{x}})$ minimizing the linear approximation $\hat{X}_0$ of the monotonotic variable X_0 over a restriction of $\mathbb{B}^I$ satisfying $\hat{X}_i(\boldsymbol{w}) = \hat{x}_i$, $i \in \{1, \ldots, n\}$. Instead of minimizing F by solving a NLMIP in the space $\mathbb{B}^I$ (Program 1) we now have a nonlinear program (NLP) in dimension $n \ll I$ (Program 3) which features the solution of a MILP (Sub-program 3) at each evaluation of the objective function g. Note that this sub-program could be *infeasible* for some vectors $\hat{\boldsymbol{x}}$ imposed for the linearized ancillary variables as the set of portfolios meeting these constraints might be empty. In practice, replacing the equality constraints of Sub-program 3 by inequalities help decreasing the number of cases for which Sub-program 3 is infeasible. The infeasible cases are ruled out by forcing then g to return a very large number.

4.2.3 Combining linearization, clustering and projection

Finally, we consider a third variant where the clustering and the linearization approaches are combined. This leads to a program similar to Program 3 but where Sub-program 3 is replaced by Sub-program 4 (optimizing over $\mathbb{R}_+^Q$ instead of $\mathbb{B}^I$) and project the solution of Sub-program 4 $\tilde{\boldsymbol{w}}^* \in \mathbb{R}_+^Q$ onto $\mathbb{B}^I$ to get our "optimal" portfolio $\boldsymbol{w}^*$ using Projection 2. Note that relaxing the binary constraint also reduces the number of cases where Sub-program 3 is infeasible. This leads to the following program:

Program 4 (Clustered version of Program 3)

$$\min_{\hat{\boldsymbol{x}}\in\mathbb{R}^n} \quad g(\hat{\boldsymbol{x}}) := F(\boldsymbol{w}^*(\hat{\boldsymbol{x}}))$$

where

1. **Sub-program 4** (Clustered version of Sub-program 3):

$$\tilde{\boldsymbol{w}}^* \in \underset{\tilde{\boldsymbol{w}}\in\mathbb{R}_+^Q}{\operatorname{argmin}} \ \hat{X}_0(\tilde{\boldsymbol{w}}) \quad \text{s.t.} \quad \begin{cases} \tilde{\boldsymbol{w}} & \leq \mathbf{1} \\ \tilde{\boldsymbol{w}} & \in \tilde{\mathcal{W}} \\ N(\tilde{\boldsymbol{w}}) & \geq \underline{N} \\ \hat{X}_i(\tilde{\boldsymbol{w}}) & = \hat{x}_i \ , \quad i \in \{1, 2, \ldots, n\} \end{cases}$$

2. **Projection 2** $\tilde{\boldsymbol{w}}^* \mapsto \boldsymbol{w}^*$ (as in Program 2).

Program 4 combines the best of the two worlds. It can be solved using a derivative-free algorithm that will compute at each evaluation of the objective function g the solution of Sub-program 4, that is, a Linear program (LP) and the solution of Projection 2, that is, a MILP. As stated before, finding a solution to the MILP is trivial. This program is fast and provides high-quality solutions in our applications, as shown in Section 6.

5 Financial applications

We now introduce two different applications of loan selection for securitization. In line with the existing literature (Sirignano et al., 2016; Jewan et al., 2009), we first consider the case of a structurer working for a development bank whose mission is to stimulate local economy by reducing the SMEs' cost of funds. A common procedure to achieve this goal is by securitizing loans bought from commercial banks, and issuing the senior tranche. The risk associated with the first tranche is kept by the investment bank. The senior tranche, however, benefits from a credit enhancement mechanism and can now be sold to risk-averse institutional investors such as pension funds. The main objective of the structurer is thus to design a "AAA" tranche. To mimic the rating score procedure, we consider two key drivers for the rating of a tranche: the ELoT and the WALoT. This is consistent with both previous studies (Ayotte and Gaon, 2011; He et al., 2012; Chen et al., 2024) and industry practice. Clearly, the attractiveness of a tranche decreases with the expected loss but increases with the average life.[7]

The second application takes the point of view of a PFI aiming to maximize the economic impact of the money invested, as motivated in the introduction. For this purpose, a PFI would theoretically finance as many loans as possible. Unfortunately, they are constrained by the capital requirements needed for each loan. The idea behind securitization is to release the capital of the PFI by selling those loans to investors as securitized assets, the PFI keeping only the most junior tranche of the CLO (and therefore capital requirement on this tranche). However, releasing this capital has a cost. We consider an objective function where we minimize the cost of releasing one unit of capital. This objective function is particularly relevant because it describes the trade-off faced by the structurer on the riskiness of the loans to include. By releasing a riskier tranche, the PFI releases more capital but the investor is expecting a higher return and this increases the cost of release. Our objective function therefore encapsulates existing formulations in the literature that optimize risk metrics of tranches. Note that the objective of optimizing the rating of the senior tranche and minimizing the cost of capital release could also be the goal of a private structurer.

5.1 Target rating

As first application, we consider a structurer willing to maximize the attractiveness of the most senior tranche, J. The attractiveness of a tranche is represented via a risk score function (the lower the score, the more attractive the tranche *ceteris paribus*). Here, we consider a risk score bounded between 0 and a 20 as per the following form for the objective function:

$$F(\boldsymbol{w}) = \min\left(20; \max\left(0; a \times \sqrt{\mathrm{EL}_J(\boldsymbol{w})} - b \times \log(\mathrm{WAL}_J(\boldsymbol{w}))\right)\right) \quad \text{with} \quad a, b > 0\,. \tag{16}$$

[7] In this setup, we do not include the return of the tranche because we adopt the viewpoint of an investment bank willing to target a specific rating. The corresponding spread will be determined accordingly.

Since $\mathrm{WAL}_J(\boldsymbol{w}) = \mathrm{wal}_J(\mathrm{WAC}(\boldsymbol{w}), \mathrm{WAM}(\boldsymbol{w}))$, $F(\boldsymbol{w})$ is a deterministic function of $\boldsymbol{w}$ via the ancillary variables $\mathrm{EL}_J(\boldsymbol{w}), \mathrm{WAC}(\boldsymbol{w})$ and $\mathrm{WAM}(\boldsymbol{w})$,

$$F(\boldsymbol{w}) := f(\mathrm{EL}_J(\boldsymbol{w}), \mathrm{WAM}(\boldsymbol{w}), \mathrm{WAC}(\boldsymbol{w})) \approx f\left(\hat{\mathrm{EL}}_J(\boldsymbol{w}), \hat{\mathrm{WAM}}(\boldsymbol{w}), \hat{\mathrm{WAC}}(\boldsymbol{w})\right) =: \hat{F}(\boldsymbol{w}) \ , \tag{17}$$

where $\hat{\mathrm{EL}}_J(\boldsymbol{w}), \hat{\mathrm{WAM}}(\boldsymbol{w}), \hat{\mathrm{WAC}}(\boldsymbol{w})$ are the linearized version of the corresponding financial variable (see Section 5.3). It is obvious from (16) that $F(\boldsymbol{w})$ is increasing in EL_J, such that the latter can be taken as X_0. One can rely on the approximate optimization problems of Section 4.2 to optimize $F(\boldsymbol{w})$. Note that finding the best functional form to approximate the rating as a function of ELoT and WALoT is out of the scope of this paper. Any rating function that only depends of $\boldsymbol{w}$ via the ELoT and the WALoT (and any other key driver of the rating that could be reasonably linearized) could be optimized using the approximate optimization problems, if the rating function is monotonic in at least one of the drivers.

5.2 Cost of capital release

As second application, we consider a structurer willing to minimize the cost of releasing one unit of capital. For the sake of simplicity, we consider for this section a setup with two tranches: a junior one (referred to as tranche 1) kept by the issuer and a senior one (tranche 2) sold to investors. In such a case, the objective function takes the form

$$F(\boldsymbol{w}) = \frac{c_2(\boldsymbol{w})}{\Delta K_2(\boldsymbol{w})} \ . \tag{18}$$

The denominator corresponds to the release of the regulatory provisions associated with tranche 2 given in (11). The numerator $c_2(\boldsymbol{w})$ corresponds to the annualized cashflows that will be lost as a result of transferring the senior tranche to investors.[8] Relying on expressions relative to the pool notional $N(\boldsymbol{w})$, the cost of releasing tranche 2 is the product of the tranche spread (an annualized rate, $s_2(\boldsymbol{w})$) with the tranche size:

$$c_2(\boldsymbol{w}) = s_2(\boldsymbol{w}) \times S_2 \ . \tag{19}$$

There is no clear expression for the spread of a tranche as a function of the pool's characteristics. This can be explained by the fact that such markets are deeply incomplete: all these products are unique in the sense that the portfolios are different and the related cashflows cannot be replicated because trading the underlying loans is not possible (in contrast, for instance, with what happens for synthetic CDOs where investors can trade the underlying CDSs). Since the spread strongly depends on the risk score, we use the same drivers as in the first application: the ELoT and the WALoT. A reasonable parametric form is obtained by assuming that the cashflows collected during the lifetime of the tranche should be an increasing function of its WAL (time-value of money) and

[8] Note that the cashflow on the numerator is the annual premium for bearing the credit risk of tranche j while the capital requirement in the denominator aims to cover the unexpected losses only, that is, the losses exceeding $\mathrm{EL}_j(\boldsymbol{w})$. One could reduce the numerator by the annualized ELoT in order to focus on the unexpected loss and related cashflows, but we stick to (18) as it does not change the expression of F.

EL (credit losses):

$$s_2(\boldsymbol{w}) \times \text{WAL}_2(\boldsymbol{w}) = \alpha \times \text{WAL}_2(\boldsymbol{w}) + \beta \times \text{EL}_2(\boldsymbol{w}) \quad \text{with} \quad \alpha, \beta \geq 0 \, .$$

The scaling coefficients α, β control the weight of each risk in the spread. This yields the expression for the tranche spread as a linear function of the annualized expected loss:

$$s_2(\boldsymbol{w}) = \alpha + \beta \times \frac{\text{EL}_2(\boldsymbol{w})}{\text{WAL}_2(\boldsymbol{w})} \, . \tag{20}$$

Note that the spread increases with the ELoT but decreases with the WALoT. Therefore, the latter acts as a penalization ruling out the portfolios consisting of loans bearing low credit risk because of expiring shortly. Clearly, $F(\boldsymbol{w}) = f(\text{EL}_2(\boldsymbol{w}), K(\boldsymbol{w}), \text{WAC}(\boldsymbol{w}), \text{WAM}(\boldsymbol{w}))$ because $\Delta K_2(\boldsymbol{w}) = K(\boldsymbol{w}) - S_1 k_1(K(\boldsymbol{w}))$ and $\text{WAL}_2(\boldsymbol{w})$ is a function of $(\text{WAM}(\boldsymbol{w}), \text{WAC}(\boldsymbol{w}))$. It is obvious from (18), (19) and (20) that $F(\boldsymbol{w})$ is a monotonic function of $\text{EL}_2(\boldsymbol{w})$. One can rely on the approximate optimization problems of Section 4.2 to optimize $F(\boldsymbol{w})$ if one can approximate $\text{EL}_2(\boldsymbol{w})$, $K(\boldsymbol{w})$, $\text{WAC}(\boldsymbol{w})$, $\text{WAM}(\boldsymbol{w})$ by a linearized version $\widehat{\text{EL}}_2(\boldsymbol{w})$, $\hat{K}(\boldsymbol{w})$, $\widehat{\text{WAM}}(\boldsymbol{w})$, $\widehat{\text{WAC}}(\boldsymbol{w})$. This is the purpose of the next section.

5.3 Linearization of the main financial variables

In those two applications, the objective functions feature as ancillary variables $\text{EL}_2(\boldsymbol{w})$, $K(\boldsymbol{w})$, $\text{WAC}(\boldsymbol{w})$, $\text{WAM}(\boldsymbol{w})$. We now propose linear approximations for these four ancillary variables. The EL on tranche j departs from a finite weighted sum of the membership vector $\boldsymbol{w}$ for three reasons: it features an integral, it relies on the min-max operator $[.]$ and the pool notional $N(\boldsymbol{w})$ shows up in the denominator.

To deal with the first problem, we replace the integral in (5) by a finite sum using a quadrature scheme. Quadratures are efficient procedures to estimate the value of an integral using a finite sum of P terms. Some quadrature schemes are tailored for expectations. For instance,

$$\mathbb{E}(h(Z)) = \int_{-\infty}^{\infty} h(x) p_Z(x) dx \approx \sum_{k=1}^{P} \pi_k h(x_k) p_Z(x_k) \, ,$$

where (x_k, π_k) are the P-points quadrature nodes and weights, respectively. The approximation is exact for polynomials of degree $2P-1$ or less. Moreover, the approximation can be made arbitrary good in the sense that the RHS tends to the LHS as $P \to \infty$. Applying this to (5) leads to the approximation EL_j^P of the exact EL_j:

$$\text{EL}_j(\boldsymbol{w}) \approx \text{EL}_j^P(\boldsymbol{w}) \quad = \quad \sum_{k=1}^{P} \pi_k \left[\frac{\sum_{i=1}^{I} w_i N_i \lambda_i p_i(x_k) - A_j N(\boldsymbol{w})}{S_j N(\boldsymbol{w})} \right] p_Z(x_k) \, , \tag{21}$$

where (x_k, π_k) are associated with a P-point quadrature. When $Z \sim \mathcal{N}(0,1)$, for instance, $p_Z = \phi$. The fact that EL_j^P is still nonlinear due to the operator $[.]$ is not an issue for optimization purposes because it can be easily circumvented using slack variables.[9] Finally, we replace $N(\boldsymbol{w})$ in the

[9]Alternative schemes can be considered, dropping the $[.]$ operator and/or relying on quadrature schemes for

denominator of (6) by the lower bound $\underline{N}$ used in the constraints, which amounts to replace $N(\boldsymbol{w})$ by $\underline{N}$ in (21). The resulting expression will serve as ancillary variable $\hat{X}_0(\boldsymbol{w})$.

It is obvious from (8) and (12) that applying the same trick to the denominator of $K(\boldsymbol{w})$, $\mathrm{WAC}(\boldsymbol{w})$ and $\mathrm{WAM}(\boldsymbol{w})$ makes them linear in $\boldsymbol{w}$, hence, could serve as ancillary variables, too. We note $\hat{\mathrm{EL}}_j^P(\boldsymbol{w})$, $\hat{\mathrm{WAC}}(\boldsymbol{w})$, $\hat{\mathrm{WAM}}(\boldsymbol{w})$ and $\hat{K}(\boldsymbol{w})$ the financial variables linearized as explained above.

6 Empirical analysis

Let us now assess the performance of the approximate optimization problems proposed in Section 4.2 on either financial applications proposed in Section 5.

6.1 Data

Testing empirically the performance of the approximate optimization problems in our context requires to make some assumptions about the credit pool's characteristics and model parameters. The proposed setup is motivated by previous academic research or, for more business-related items such as loans' interest rate or rating functions, based on extensive discussions with major players active in the field of SME loans securitization.

As for the pool characteristics, two setups can be adopted: a deterministic or a stochastic approach. Regarding the default probabilities, for example, Glasserman and Ruiz-Mata (2006) sets the marginal PD's in $[0, 0.02]$ using a deterministic data generating process $p_i = 0.01(1+\sin(16\pi i/I))$ while Gordy (2002) sample from the exponential law, $p_i \sim \mathrm{Exp}(\xi)$ i.i.d. for some scale parameter $\xi > 0$. In this empirical study, we adopt the stochastic approach, which allows us to assess the impact of various parameters on different pools which are statistically equivalent.

We build our heterogeneous pools according to the stochastic model summarized in Table 1. The notional N_i and the maturity M_i of each loan are positive variables that can be generated using i.i.d. samples from a Gamma distribution. Based on discussions with practitioners, we calibrate the parameters of the notional (in units of 5k) distribution to satisfy $\mathbb{E}[N_i] = 5$ and $\mathbb{V}[N_i] = 3.5$. For the maturity (in years), we choose the Gamma parameters to match $\mathbb{E}[M_i] = 5$ and $\mathbb{V}[M_i] = 1$. As for the LGD λ_i, we choose to draw from a Beta distribution to comply with the $[0, 1]$ interval. We set the parameters to match the first and second moments reported in previous studies (Caselli et al., 2008; Kaposty et al., 2022). Regarding the marginal default probabilities, we are interested here in two horizons: the probability that the loan defaults before maturity, $p_i = \mathbb{P}(\tau_i \leq M_i)$, and the 1-year PD that matters for regulatory capital requirements, $\mathrm{PD}_i = \mathbb{P}(\tau_i \leq 1)$. In order to have consistent probabilities bounded in $[0, 1]$, we adapt the procedure of Gordy (2002) and draw exponential marginal cumulative probability *curves*, $F_i(t) = \mathbb{P}(\tau_i \leq t)$. To this end, we first draw a default rate $\xi_i > 0$ for each loan using i.i.d. Gamma samples, $\xi_i \sim \Gamma(a, b)$, where a and b are chosen to match the first and second moments of previous studies (Andreeva et al., 2016; Kim and Sohn, 2010; Stevenson et al., 2021). Then, we postulate an exponential law with parameter ξ_i for the marginal default distributions,

$$F_i(t) = 1 - e^{-\xi_i t} \,. \tag{22}$$

truncated distributions, but this result in a nonlinear expression in $\boldsymbol{w}$, hence, are not appropriate for our purpose.

Then, we simply set $p_i = F_i(M_i)$ and $\mathrm{PD}_i = F_i(1)$. The expected default rate in the pool by a fixed time horizon t is

$$\mathbb{E}\left(\frac{1}{I}\sum_{i=1}^{I} F_i(t)\right) = 1 - \frac{1}{I}\sum_{i=1}^{I}\mathbb{E}(1-\mathrm{e}^{-\xi_i t}) = 1-(1+bt)^{-a} ,$$

such that the expected 1-year PD is 3.43%.[10] The annual rate r_i of loan i is taken as a deterministic function of the maturity M_i and the 1-year relative expected loss $\mathrm{PD}_i\lambda_i$ of the loan:

$$r_i = \frac{50 + 0.8\ M_i + 8000\ \mathrm{PD}_i\lambda_i}{10000} . \tag{23}$$

In Appendix B, we present histograms and descriptive statistics for the characteristics of the loans in a pool of size $I = 10,000$, and show that they comply with the moments in Table 1.[11]

Symbol	Variable	Distribution	Mean	Std Dev.
N_i	notional	$\Gamma(2, 2.5)$	5	3.535
λ_i	LGD	$\beta(2, 2)$	0.5	0.224
M_i	maturity	$\Gamma(20, 0.25)$	5	1.118
ξ_i	default rate	$\Gamma(7, 0.005)$	0.035	0.013
p_i	PD at maturity		0.158	0.063
PD_i	PD at 1Y horizon		0.034	0.013
r_i	interest rate		0.019	0.008

Table 1: Stochastic model used to generate the parameters of the loans. The notional is given in units of 5k euros and maturity in years. The marginal cumulative default probability curves follow an exponential law with parameter ξ_i. The expectations and standard deviations are computed analytically from the exact distributions.

Having fixed the pool parameters, we can now proceed with the model driving the portfolio losses. Although other choices can be made, we consider for illustration purpose the 1FGC setup, and assume that the expression of the loading factors controlling the dependence between the defaults coincides with the Basel formula of capital requirements (see Appendix A).

With regards to the CLO design, we assume that the structurer wants to select $\underline{N} = 0.75$ of the total outstanding of the loans in the portfolio. For the sake of simplicity, we do not consider any constraint in $\mathcal{W}$ (see Appendix F for a case with sector diversification constraints). For the tranching structure, in line with Antoniades and Tarashev (2014), we assume a CLO with 3 tranches: a junior one with attachment and detachment points equal respectively to 0% and 10%, a mezzanine tranche with attachment and detachment points of respectively 10% and 20% and a senior one with attachment and detachment points of respectively 20% and 100%. Since the second application features only two tranches, we merge the mezzanine and the senior tranches to get $A_1 = 0$, $D_1 = A_2 = 0.1$, $D_2 = 1$. In the first application, the parameters a and b are set respectively to 300 and 0.5. In the second application, the parameters α and β are respectively set to 0.0004 and 0.5. With

[10] Note that this setup is merely used to set the parameters of the pool, not to actually sample default times.

[11] Of course, many different setups could be considered. Still, it is comparable to credit pools investigated in other studies. Düllmann and Masschelein (2007), for instance, consider a homogeneous pool of $I = 6,000$ SME exposure with 1-year default probability PD=2%, an LGD of $\lambda = 45\%$ and a notional of $N = 1,000$. In this work, we consider larger PD's and notional amounts in order to better stick to actual figures. In addition, we consider a heterogeneous pool as the loan selection problem is trivial otherwise, all loans being mutually exchangeable.

regards to the integral in the ELoT, we rely on a change of variable $x = \tan(\pi/2u)$ combined with the Gauss-Legendre quadrature:

$$\int_{-\infty}^{\infty} h(x)dx = \frac{\pi}{2}\int_{-1}^{1} \frac{h(\tan(\pi/2u))}{\cos^2(\pi/2u)}du \approx \frac{\pi}{2}\sum_{k=1}^{P} \pi_k \frac{h(x_k)}{\cos^2(\tilde{u}_k)} , \tag{24}$$

where $\tilde{u}_k = u_k\pi/2$, $x_k = \tan(\tilde{u}_k)$ and the (u_k, π_k) pairs are obtained using `gauss.quad(P,'legendre')` available in the `statmod` library of the statistical software `R`, using $P = 100$ points.[12]

A 100% PSA model is considered to assess the impact of prepayments on the WAL. For the clustering procedure, we use k-means algorithm on the standardized set of all loans characteristics used in the optimization and we report the performance for different number of clusters Q.

6.2 Solutions of the programs and benchmarks

In this section, we compare the efficiency of different methods in terms of the terminal value of the objective function, the similarities in the optimal solution portfolios, and computation time. To build the four different optimization programs, we have used 2 different approximations: (i) linear approximations of the ancillary variables and (ii) clustering loans to reduce the dimensionality and relax the binary constraint. Table 2 summarizes which assumptions is used in each program.

Program	Type	Linearization	Clustering
1	NLMIP	No	No
2	NLP + projection	No	Yes
3	NLP (MILP at each iteration)	Yes	No
4	NLP (LP and projection at each iteration)	Yes	Yes

Table 2: Approximations used in each optimization program.

Program 1 is solved using Genetic Algorithm from `R` package `GA`.[13] Programs 2, 3 and 4 are solved using Nelder-Mead algorithm from `R` package `nloptr` using default convergence criterion: stop when an optimization step changes every parameter by less than $1 \times e^{-04}$ multiplied by the absolute value of the parameter. The notional constraint is added to the objective as a penalty in Programs 1 and 2. All MILPs (Program 3) and LPs (Program 4) are solved with Gurobi using default parameters. We compare the solutions of those four algorithms to those of various benchmarks. First, we consider a benchmark (called "Heuristics (EL)" in the sequel) exploiting the connection between the riskiness of a tranche and that of the underlying loans. It consists in selecting the loans in ascending order of their expected losses until all the constraints are met. Alternatively, the priority rule can be determined by sorting the loans in descending order of their maturity, capital requirement or coupon, leading to Heuristics (M), (K) and (r), respectively.

Tables 3 and 4 display the results of each heuristic and program considering Application 1 and featuring 1,000 and 10,000 loans, respectively. Tables 5 and 6 show the corresponding results for

[12]Note that the `gauss.quad.prob(P,'normal')` command yields the points (x_k, π_k) such that $\mathbb{E}(h(Z)) = \int h(x)\phi(x)dx \approx \sum_{k=1}^{P} \pi_k h(x_k)$ and seems therefore more appropriate to compute integrals with respect to the normal distribution, making the economy of a change of variable and of the evaluation of the $\phi(x_k)$s. However, the Gauss-Legendre scheme proves to be much more efficient when dealing with ELoTs, as noted in Vrins (2009).

[13]Parameters `popSize` and `maxiter` are set to match a desired running time (e.g. 1 hour).

Application 2. The computation time is capped to 1 hour.[14] We fix the number of clusters to $Q = 200$ for the 1,000 loans dataset and $Q = 2,500$ for the 10,000 loans dataset.

	EL	WAL	$\hat{F}(\tilde{\boldsymbol{w}}^*)$	$F(\boldsymbol{w}^*)$	Comp. time
Heuristics (EL)	0.0001	2.8111		2.3103	< 1 sec
Heuristics (M)	0.0011	3.1504		9.1633	< 1 sec
Heuristics (K)	0.0024	2.9634		14.2832	< 1 sec
Heuristics (r)	0.0024	2.9612		14.1423	< 1 sec
Program 1	0.0003	2.8745		4.3503	-
Program 2	0.0002	2.8808	4.2443	4.2104	-
Program 3	0.0001	2.8303	2.0764	**2.0764**	7 sec
Program 4	0.0001	2.8229	**2.0568**	2.1072	14 sec

Table 3: Application 1, pool of $I = 1,000$ loans built according to the setup depicted in Table 1 and $Q = 200$ clusters for programs 2 and 4. Columns 1-2 display the value of intermediary variables. Column 3 displays the value of the objective function before projection and/or using quadrature. Column 4 displays the exact value of the objective function at the final solution. Column 5 gives the computation time (a - indicates that the 1-hour limit was reached before the trigger of the default convergence criterion).

	EL	WAL	$\hat{F}(\tilde{\boldsymbol{w}}^*)$	$F(\boldsymbol{w}^*)$	Comp. time
Heuristics (EL)	0.0038	2.5833		2.1533	< 1 sec
Heuristics (M)	0.0149	2.9029		8.7901	< 1 sec
Heuristics (K)	0.0217	2.7165		14.1319	< 1 sec
Heuristics (r)	0.0221	2.7318		14.0525	< 1 sec
Program 1	0.0005	2.9121		6.4991	-
Program 2	0.0004	2.9014	5.5170	5.3440	-
Program 3	0.0001	2.8474	1.8895	**1.8895**	2 min
Program 4	0.0001	2.8436	**1.8810**	1.8913	1 min

Table 4: Application 1. Same as Table 3 but for a pool of $I = 10,000$ loans and $Q = 2,500$ clusters for programs 2 and 4.

	EL_2	WAL_2	c_2	ΔK_2	$\hat{F}(\tilde{\boldsymbol{w}}^*)$	$F(\boldsymbol{w}^*)$	Comp. time
Heuristics (EL)	0.0040	2.5794	3.9286	75.2371		0.0522	< 1 sec
Heuristics (M)	0.0155	2.8916	10.3280	190.8660		0.0541	< 1 sec
Heuristics (K)	0.0220	2.7199	14.9265	275.4356		0.0542	< 1 sec
Heuristics (r)	0.0223	2.7180	15.1388	265.2227		0.0571	< 1 sec
Program 1	0.0100	2.6384	7.6761	163.1167		0.0471	-
Program 2	0.0091	2.6474	7.1857	158.5953	0.0457	0.0453	-
Program 3	0.0068	2.5583	5.7981	138.9650	**0.0417**	0.0417	1 min
Program 4	0.0073	2.6185	6.0073	145.1654	**0.0417**	**0.0414**	20 sec

Table 5: Application 2, pool of $I = 1,000$ loans built according to the setup depicted in Table 1 and $Q = 200$ clusters for programs 2 and 4. Columns 1-4 display the value of intermediary variables. Column 5 displays the value of the objective function before projection and/or using quadrature. Column 6 displays the exact value of the objective function at the final solution. Column 7 gives the computation time (a - indicates that the 1-hour limit was reached before the trigger of the default convergence criterion).

[14] We use a 12th Gen Intel(R) Core(TM) i7-12700H processor, 2300 MHz, 14 cores, 20 logical processors with a RAM of 16 Go.

	EL_2	WAL_2	c_2	ΔK_2	$\hat{F}(\tilde{\boldsymbol{w}}^*)$	$F(\boldsymbol{w}^*)$	Comp. time
Heuristics (EL)	0.0038	2.5833	38.1221	703.1298		0.0542	< 1 sec
Heuristics (M)	0.0149	2.9029	99.9399	1870.2261		0.0534	< 1 sec
Heuristics (K)	0.0217	2.7165	148.0086	2753.4397		0.0538	< 1 sec
Heuristics (r)	0.0221	2.7318	149.4919	2671.4309		0.0560	< 1 sec
Program 1	0.0112	2.6832	84.2844	1668.3336		0.0505	-
Program 2	0.0111	2.6757	91.7393	1895.7960	0.0490	0.0484	-
Program 3	0.0068	2.5652	57.9572	1392.9926	0.0416	0.0416	7 min
Program 4	0.0075	2.6555	60.8801	1477.7290	**0.0413**	**0.0412**	2 min

Table 6: Application 2. Same as Table 5 but for a pool of $I = 10,000$ loans and $Q = 2,500$ clusters for programs 2 and 4.

Heuristic portfolios perform badly compared to any of the optimization program in each application and each dataset, except for Heuristics (EL) in Application 1. This underlines the usefulness of solving the optimization problem rather than selecting the “best” loans based on marginal criteria. Yet, solving the initial problem (Program 1) leads to disappointing solutions, as they are significantly worse than those featuring the clustering (Program 2) or the linearization (Program 3). Between the two, the latter leads to a stronger improvement compared to Program 1 as the solution to Program 3 is better than that of Program 2 in all configurations. Program 4, which consists in combining both approximations, allows to improve the results further in Application 2. For Application 1, it seems that adding the clustering to the linearization slightly deteriorates the solution, but the results are extremely close. Column 5 underlines that the value of the objective function before projection and/or using quadrature remains very close to the final value.

6.3 Similarities of the solution portfolios

Section 6.2 discusses the results of the optimization program by analyzing the terminal value of the objective function after convergence or a 1-hour limit, whichever comes first. Program 3 and Program 4 seem to yield similar results in this regard. In addition, Table 7 shows that the programs select a relatively similar proportion of loans compared to the original pool size I, about 75% in each case. Of course, this ratio results from the properties of the pool, combined with the $\underline{N} = 0.75$ constraint.

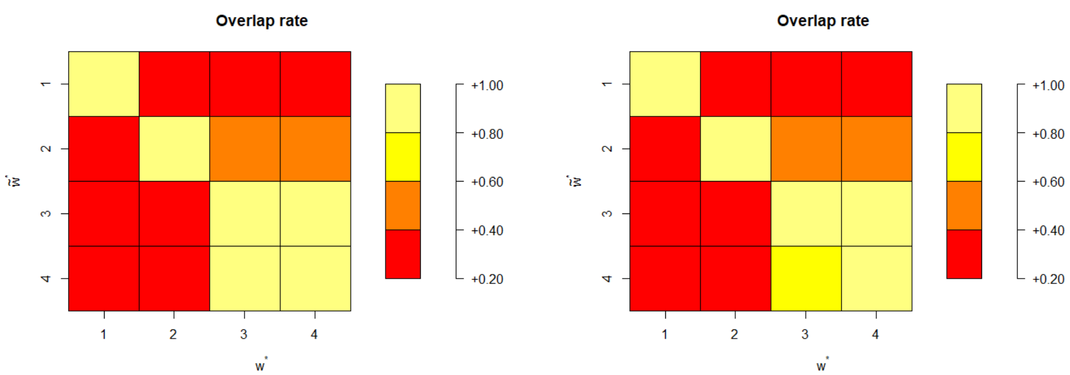
	Application 1		Application 2	
	1,000 loans	10,000 loans	1,000 loans	10,000 loans
Program 1	74.10	74.62	75.80	75.38
Program 2	72.30	78.20	72.20	77.62
Program 3	75.50	75.13	75.30	74.71
Program 4	75.70	75.04	75.20	74.66

Table 7: Proportion $|\,\boldsymbol{w}^*\,|/I$ (in %) of selected loans for the solution portfolios $\boldsymbol{w}^*$ associated with tables 3, 4, 5 and 6.

Yet, two portfolios with similar hamming weights could lead to similar values of the objective function while selecting a very different set of loans. To investigate the similarities between the solution portfolios, we introduce a measure of overlap between membership vectors $\boldsymbol{w}, \tilde{\boldsymbol{w}} \in \mathbb{B}^I$ as[15]

$$\langle \boldsymbol{w} \,|\tilde{\boldsymbol{w}}\rangle = \frac{\boldsymbol{w}\,\tilde{\boldsymbol{w}}^\top - (|\tilde{\boldsymbol{w}}| + |\,\boldsymbol{w}\,| - I)^+}{\max(|\,\boldsymbol{w}\,|, |\tilde{\boldsymbol{w}}|) - (|\tilde{\boldsymbol{w}}| + |\,\boldsymbol{w}\,| - I)^+} \quad \text{if} \quad \min(|\,\boldsymbol{w}\,|, |\tilde{\boldsymbol{w}}|) < I \quad \text{and 1 otherwise .}$$

The first term in the numerator yields the number of loans in common in the portfolios $\boldsymbol{w}$ and $\tilde{\boldsymbol{w}}$, while the second provides a lower bound of the later, i.e., is the minimum number of loans in common between two portfolios of such sizes. The first term in the denominator gives the maximum size, which is larger than $\boldsymbol{w}\,\tilde{\boldsymbol{w}}^\top$ with equality if and only if the two vectors are identical. Thus, our indicator is nonnegative, is zero if the number of loans in common is equal to the achievable minimum given the size of those portfolios, and 1 if they are identical. The overlap rates for all the pairs of portfolios returned by the four programs are given for both pool sizes and applications in Figure 1. One can see that with an overlap rate of about 20-40% with the other solutions, Program 1 leads to a solution that is relatively different compared to the others. Program 2 leads to solutions that are closer to that of Program 3 and Program 4. Programs 3 and 4 yield quite similar solutions, in the sense that these programs lead to portfolios having an overlap rate higher than 90%.



(a) Application 1.

(b) Application 2.

Figure 1: Heatmaps of overlap rates $\langle \boldsymbol{w}^*, \tilde{\boldsymbol{w}}^* \rangle$ for all the pairs of solution portfolios $\boldsymbol{w}^*, \tilde{\boldsymbol{w}}^*$ reported in tables 3, 4, 5 and 6. The upper (resp. lower) triangle gives the overlap rate for $I = 1,000$ (resp. $I = 10,000$). The diagonals compare two solutions coming from a same program, hence, leads to 1 in either case.

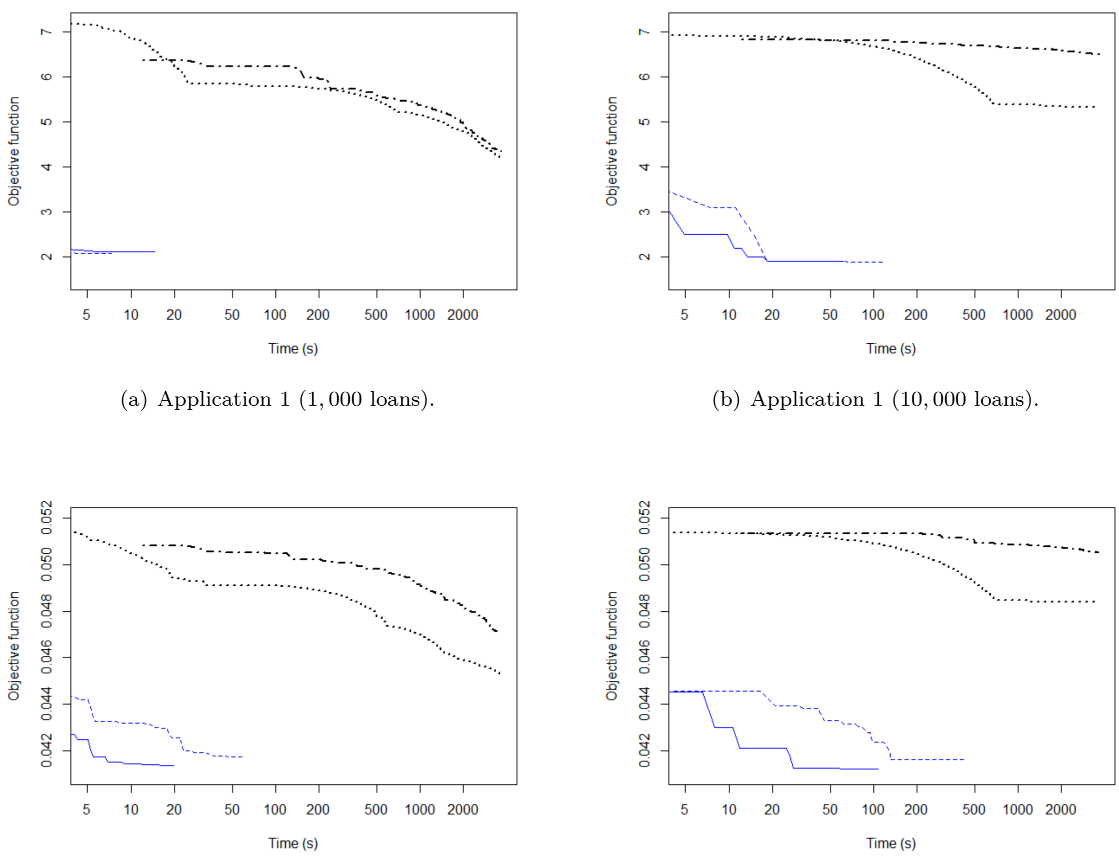
6.4 Convergence

The results reported in Section 6.2 show that our approximations help increase the quality of the solution. Interestingly, they also help reduce the computation time, sometimes drastically. Figure 2 illustrates the convergence speed of the algorithms for Application 1 for a pool featuring 1,000 (a) and 10,000 (b) loans. The solution provided by Program 1 improves at a very low pace, especially when the pool size is large. Note that first iteration of Program 1 – hence, the time taken to return the first value of the objective function – takes more time ($\approx$ 10s sec) than the ones of Programs 2,3

[15] A simpler choice to measure the overlap would be $\boldsymbol{w}\,\tilde{\boldsymbol{w}}^\top / I$. However, this would return a number which is 1 only of both portfolios select all the loans, and that would naturally increase with the number of selected loans, hence, with the constraint $\underline{N}$. For instance, two portfolios satisfying $|\,\boldsymbol{w}\,|/I = |\tilde{\boldsymbol{w}}|/I = 75\%$ will necessarily exhibit $\boldsymbol{w}\,\tilde{\boldsymbol{w}}^\top / I \geq (|\tilde{\boldsymbol{w}}|/I + |\,\boldsymbol{w}\,|/I - 1)^+ = 50\%$. Removing the lower bound from the indicator mitigates this effect.

and 4 due to the nature of the Genetic Algorithm. Program 2 behaves slightly better in this case, but exhibits longstanding plateaus. The differences in terms of solution portfolios highlighted in Section 6.3 suggests that Program 2 tends to get stuck in local optima. Program 3 and Program 4 seems to behave much better in this respect. The solutions are quite similar, and much better than those returned by programs 1 and 2, as reported in Tables 3 and 4. In addition, the algorithms converge – hence, stop – much before the 1-hour stopping criterion (below 20 second for 1,000 loans and below 2 minutes for 10,000 loans), although Program 4 is about twice as fast as Program 3 when facing large pools.

The results for Application 2 are shown in Figure 2 and exhibit a similar behavior. In particular, the quality of the solutions of programs 3 and 4 are comparable, and much better than that of the solutions of programs 1 and 2. In addition, they also converge faster, albeit Program 4 proves to be superior to Program 3 for either pool sizes.



(a) Application 1 (1,000 loans).

(b) Application 1 (10,000 loans).

(c) Application 2 (1,000 loans).

(d) Application 2 (10,000 loans).

Figure 2: Evolution of the objective function $F(\boldsymbol{w})$ with respect to time for each program in Application 1 (top) and 2 (bottom). Logarithmic scale for x-axis. Dash-dotted (black, top) line for Program 1. Dotted (black, top) line for Program 2. Dashed (blue, bottom) line for Program 3. Solid (blue, bottom) line for Program 4.

6.5 Impact of the number of clusters

A central parameter in programs 2 and 4 is Q, the number of clusters. In this section, we discuss its impact on the quality of the solution. To this end, we draw ten independent credit portfolios of 10,000 loans according to the procedure described above (i.e., we have ten different independent copies of a pool of size $I = 10,000$)[16] and report the solution of Program 4 after a running time of 60 seconds for different number of clusters as a form of box-plots. We compare it to the solution of Program 3 after the same running time. The result is shown in Figure 3. The U-shape of the solution with respect to Q was expected because the approximation is too rough at the extremes. The clustering is too rough when Q is small, such that the projected solution $\boldsymbol{w}^*$ can deviate substantially from $\tilde{\boldsymbol{w}}^*$. When Q is too large, many clusters will contain a single loan, in which case the procedure essentially amounts to relax the binary constraint and then project the solution. In Application 1 (panel a), the function is rather flat, which suggests that the solution is not too sensitive to the number Q of clusters chosen. With regard to Application 2 (panel b), the dependence with respect to Q is more striking but the solution does not critically depend on Q in the sense that Program 4 performs better, on average, than Program 3 (last box) for every Q, and is quite insensitive to Q for reasonable choices.

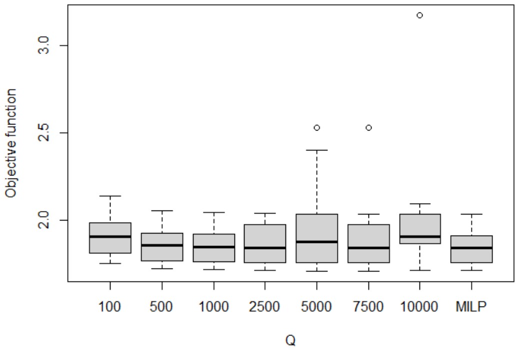
7 Extensions of the credit loss model

Our approach is illustrated using the One-Factor Gaussian Copula loss model combined with the large pool approximation. This choice is motivated by the fact that the 1FGC loss model is not central in our paper and can be exposed in a concise way. Moreover, in spite of its simplicity, it remains widely used in the industry and for regulation purposes. Nevertheless, it is important to realize that this choice is not a restriction of our approach and can be easily relaxed. For instance, Remark 1 highlights that only the shape of $p_i(z)$ in (7) changes when changing the distribution of the latent variables $Z, \epsilon_1, \ldots, \epsilon_I$. In addition, one could consider other kinds of coupling schemes (copula or number of factors). Moreover, the LGD can also be dependent upon the factors. In this section, we briefly explain the relevance of relying on the LP assumption and introduce some variants of the 1FGC setup that can be easily tackled in our framework. Some related results are reported in the Appendix. They illustrate that the conclusions drawn in the previous section are not only valid in the 1FGC setup, but hold in those frameworks, too.

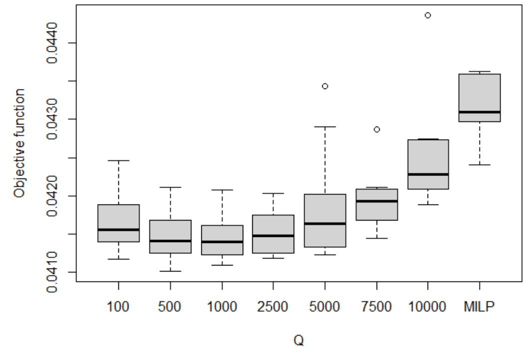
7.1 Large pool approximation

In this paper, we have chosen to rely on the large pool assumption for two reasons. First, because the underlying assumptions precisely coincide with the problem at hand. Indeed, SME securitization feature a very large number of loans (typically, a couple of thousands), each with a small amount compared to the total. This explains why no granularity adjustment correcting for the finite size of the pool is required. Second, the LP assumption combined with factor models has the very nice feature that the loss can be decomposed as a sum of individual contributions, resulting in a function that is linear in the membership weights.

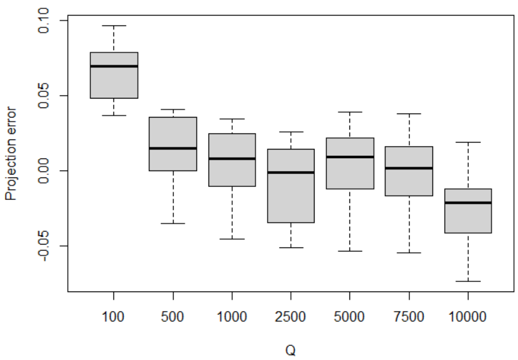
[16] Results are similar for $I = 1,000$ but are skipped for conciseness.



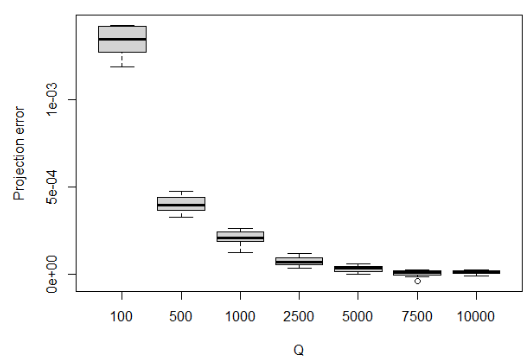
(a) Application 1.



(b) Application 2.



(c) Application 1.



(d) Application 2.

Figure 3: Distribution of the objective function $F(\boldsymbol{w}^*)$ (top) and of the projection error $(\hat{F}(\tilde{\boldsymbol{w}}^*) - F(\boldsymbol{w}^*))$ in Program 4 as a function of the number Q of clusters estimated by running the algorithm on 10 independent copies of a portfolio of size $I = 10,000$ loans.. The last boxplot (MILP) in the top panels refers to Program 3 (no clustering).

The case of finite pool is more problematic because the computational cost is much higher (e.g., Parzen recursion provides the exact loss distribution, but already extremely time-consuming for a couple of hundreds of loans with various LGDs) and the appropriate numerical procedures lead to a nonlinear expression in the weights. For instance, the Laplace transform is commonly used when relaxing the LP assumption. The latter relies on the moment generating function of the portfolio loss $N(\boldsymbol{w})L(\boldsymbol{w})$, which derives from the conditional independence:

$$\mathbb{E}\left(\mathrm{e}^{uN(\boldsymbol{w})L(\boldsymbol{w})}\right) = \mathbb{E}\left(\mathrm{e}^{u\sum_{i=1}^{I} w_i L_i}\right) = \mathbb{E}\left(\prod_{i=1}^{I} \mathbb{E}(\mathrm{e}^{uw_i N_i \lambda_i B_i} \,|Z)\right) = \mathbb{E}\left(\prod_{i=1}^{I} \psi_{B(p_i(Z))}(uw_i N_i \lambda_i)\right),$$

where $\psi_{B(\pi)}(u) = (1-\pi) + \mathrm{e}^u \pi$ is the moment generating function of a Bernoulli random variable with parameter π. Next, an estimation of $F_{L(\boldsymbol{w})}(l)$ is obtained by inverse Laplace transform, and the ELoT is found using an extra integral similar to (5), except that the integral is with respect to

L instead of Z:

$$\begin{aligned}\mathrm{EL}_j(\boldsymbol{w}) &= \int_0^\infty \left[\frac{l-A_j}{S_j}\right] dF_{L(\boldsymbol{w})}(l) \\ &= \frac{1}{S_j}\int_{A_j}^{D_j} l dF_{L(\boldsymbol{w})}(l) - A_j \frac{F_{L(\boldsymbol{w})}(D_j) - F_{L(\boldsymbol{w})}(A_j)}{S_j} + \left(1 - F_{L(\boldsymbol{w})}(D_j)\right) . \qquad (25)\end{aligned}$$

Appendix D highlights that the LP approximation does not harm the performance of our approach, on the contrary. In particular, the programs' ranking built upon the exact objective function $F^L(\boldsymbol{w}^*)$ computed using Laplace transform is the same as that obtained when ranking the performance using the objective $F(\boldsymbol{w}^*)$, relying on the LP approximation. In addition, the solutions returned by programs 1 and 2 exploiting the exact objective $F^L(\boldsymbol{w})$ are much worse than those returned by programs 3 and 4, featuring the objective $F(\boldsymbol{w})$ relying on the LP assumption. This shows the benefit of relying on the LP approximation in conjunction with programs 3 and 4, despite the fact that $F(\boldsymbol{w})$ is an approximation of $F^L(\boldsymbol{w})$.[17]

7.2 Non-Gaussian copula

Let us now move to the possible variants that our framework can accommodate. A first extension of the 1FGC approach deals with the type of copula controlling the dependence structure between the default events. For instance, dealing with the LP approximation in a t-copula is not much different. This model is obtained by rescaling the Z_i's in (1) with another systematic component $\sqrt{\nu/V}$ where $V \sim \chi^2(\nu)$ is independent from the standard normal variables $Z, \epsilon_1, \ldots, \epsilon_I$. Hence, we have independence between individual losses when conditioning on the pair (Z, V), leading to a 2-factor model. One gets that for $I \to \infty$,

$$L(\boldsymbol{w}) \to \mathbb{E}(L(\boldsymbol{w})|Z,V) = \frac{1}{N(\boldsymbol{w})}\sum_{i=1}^{I} w_i N_i \lambda_i p_i(Z,V) \quad \text{where} \quad p_i(Z,V) = \Phi\left(\frac{\sqrt{\frac{V}{\nu}}\Phi_\nu^{-1}(p_i) - a_i Z}{\sqrt{1-a_i^2}}\right)$$

and Φ_ν^{-1} is the quantile function of the standard Student-t distribution with ν degrees of freedom. The integral in (5) is now bi-dimensional (one needs to integrate V away on the top of Z), which remains tractable. Because the Chi-squared belongs to the Gamma family, the integration with respect to V can be achieved using appropriate quadrature schemes. Appendix E displays the solution of each program when a t-copula is considered. The results in terms of programs' performance are similar to the ones obtained previously with the 1FGC.

7.3 Multi-factor models

The loss model can also feature several systematic factors, for example, capturing regional or industry effects. The multifactor setup corresponds to the case where the default events are all mutually independent conditional upon a set of factors $\mathbf{Z} = (Z_1, \ldots, Z_k)^\top$. In this model, eq. (1) changes to

$$Z_i = a_i Y_i + \sqrt{1-a_i^2}\epsilon_i \quad \text{where} \quad Y_i = \boldsymbol{b}_i^\top \mathbf{Z} \quad \text{and} \quad \boldsymbol{b}_i \in \mathbb{R}^k \quad \text{satisfies} \quad \|\boldsymbol{b}_i\| = 1 . \qquad (26)$$

[17] Recall that one cannot use the objective F^L in programs 3 and 4 as it does not lead to an expression for EL_j being linear in the weights.

In this case, the large pool approximation yields

$$L(\boldsymbol{w}) \to \mathbb{E}(L(\boldsymbol{w})|\mathbf{Z}) = \frac{1}{N(\boldsymbol{w})}\sum_{i=1}^{I} w_i N_i \lambda_i p_i(\mathbf{Z}) \quad \text{as} \quad I \to \infty \,,$$

where $p_i(\mathbf{Z})$ is the conditional PD of i given $\mathbf{Z}$. This shows that extending our setup to multiple and arbitrary factors is straightforward; the only price to pay is that the dimension of the integral in (5) becomes multidimensional. This is not an obstacle to using our quadrature approximation as long as the number k of factors and the number P_j of quadrature points are not too large. Note that this setup can be combined with the t-copula, it suffices to scale the Z_i's in (26) by $\sqrt{\nu/V}$. One could also proxy this multi-factor model using a *comparable single-factor* (CSF) model following (Pykhtin, 2004). Appendix F illustrates the performance of our approach in such a case, and includes diversification constraints for each sector. The conclusions regarding both the ranking and computational times of the models are similar to those obtained with the 1FGC. In particular, optimizing the objective function associated with the CSF model using programs 3 and 4 leads to a substantial improvement compared to running programs 1 and 2. Note that, in contrast with Section 7.1, it is hardly possible to apply programs 1 and 2 to the exact EL_j computed using the multi-factor mode, as each evaluation of the latter is extremely time-consuming.

7.4 PD-LGD dependency

Finally, our setup can also accommodate stochastic LGDs (Λ_i) provided that the default events and the losses all remain mutually independent conditionally upon the systematic factor, Z. For any such coupling scheme, it suffices to replace λ_i in (6) by $\lambda_i(Z)$ where $\lambda_i(Z) := \mathbb{E}(\Lambda_i|B_i = 1, Z)$ and the linearity in $\boldsymbol{w}$ is preserved. See Barbagli and Vrins (2023) for more details and some tractable examples. This also applies when considering a multifactor setup and/or the t-copula (conditioning with respect to $\mathbf{Z}$, (Z, V) or $(\mathbf{Z}, V)$, depending on the approach chosen).

8 Conclusion

SMEs account for an important part of wealth creation in modern economies. Yet, they are typically riskier than large firms and, consequently, face more difficulties to get financed. In an era of costly energy, high inflation, and scarce public investment capacities, reducing the funding costs of SMEs is therefore more critical than ever.

Among the various strategies put in force by public financial institutions (PFIs), securitization proves to be particularly efficient for this purpose. Thanks to the credit enhancement mechanism inherent to collateralized loan oblogations (CLO), transferring SME loans from commercial credit institutions to private investors while the junior tranche is held by PFIs is a way to reduce the requirements regarding both interest rates and collateral. On the other hand, these loans are removed from the balance sheet of commercial banks, leading to capital release and, consequently, increasing the available financing volumes. Finally, these tranches form a new asset class with various risk-return profiles, enabling fund managers to build more efficient portfolios.

A key step in the securitization process is the identification of which loan to embed in a CLO. Indeed, the loan selection problem is a high-dimensional nonlinear mixed-integer optimization program (NP-hard) featuring complex objective functions whose evaluation relies on costly numerical methods such as heavy Monte Carlo simulations. In this paper, we circumvent this challenge by relying on clustering and linearization. Specifically, we derive approximations of the original problem which are lower dimensional, relax the binary constraints and tackle nonlinearity issues. Our approach can accommodate pools made of several thousands of loans, various forms of objective functions and different loss models. Our simulation studies feature realistic pools and show that the proposed approach provides better and much faster solutions than those found by standard nonlinear mixed-integer program solver in every tested configuration.

Our contribution is useful for any stakeholder involved in the selection step of the loans securitization process, including public financial institutions such as development and central banks, as well as the structuring teams of commercial banks. However, by enhancing the securitization procedure, our research also benefits to public authorities, private investors such as pension funds, as well as to the SME sector as a whole.

Acknowledgment

The work of Arnaud Germain is funded by the ING-LFIN chair. Frédéric Vrins benefits from the financial support of the Fonds de la Recherche Scientifique F.S.R.-FNRS (grant J.0225.24) as well as of the Belgian Federal Science Policy Office (grant ARC 18-23/089). The authors warmly thank Aurélien Mouly, Lilan Lecomte and Jacques Makutonin (European Investment Fund) as well as Guillermo Navas and Dylan Cissou (Morningstar DBRS) for interesting discussions about this project and their guidance regarding the design of the objective functions and empirical setup. The authors are grateful to Erick Delage (HEC Montréal), Laurence Wolsey (CORE, UCLouvain), the participants of AFMath24 (Belgium) and ICCF24 (Amsterdam) whose constructive comments and feedback contributed to improve preliminary versions.

References

Andreeva, G., Calabrese, R., and Osmetti, S. A. (2016). A comparative analysis of the uk and italian small businesses using generalised extreme value models. *European Journal of Operational Research*, 249(2):506–516.

Antoniades, A. and Tarashev, N. A. (2014). Securitisations: Tranching concentrates uncertainty. *BIS Quarterly Review December*.

Ayotte, K. and Gaon, S. (2011). Asset-backed securities: Costs and benefits of "bankruptcy remoteness". *The Review of Financial Studies*, 24(4):1299–1335.

Barbagli, M. and Vrins, F. (2023). Accounting for PD-LGD dependency: A tractable extension to the basel asrf framework. *Economic Modelling*, 125:106321.

BCBS (2019). Calculation of RWA for credit risk - securitisation: Standardised approach. Technical report, Bank for International Settlements.

BCBS (2023). Calculation of RWA for credit risk - IRB approach: Risk weight functions. Technical report, Bank for International Settlements.

Bennett, P. (1984). Applying portfolio theory to global bank lending. *Journal of Banking & Finance*, 8(2):153–169.

Bo, L. and Capponi, A. (2014). Optimal investment in credit derivatives portfolio under contagion risk. *Mathematical Finance*, 26(4).

Caselli, S., Gatti, S., and Querci, F. (2008). The sensitivity of the loss given default rate to systematic risk: New empirical evidence on bank loans. *Journal of Financial Services Research*, 34:1–34.

Chen, X., Cheng, Q., and Luo, T. (2024). The economic value of blockchain applications: Early evidence from asset-backed securities. *Management Science*, 70(1):439–463.

Düllmann, K. and Masschelein, N. (2007). A tractable model to measure sector concentration risk in credit portfolios. *Journal of Financial Services Research*, 32(1-2):55–79.

Egloff, D. and Leippold, M. (2010). Quantile estimation with adaptive importance sampling. *The Annals of Statistics*, 38(2):1244 – 1278.

European Commission (2023). SME and high inflation (summary report). Technical report, Directorate-General for Internal market, Industry, Entrepreneurship and SMEs.

Giesecke, K. (2008). Portfolio credit risk: Top-down versus bottom-up approaches. In *Frontiers in Quantitative Finance*, chapter 10, pages 251–267. John Wiley & Sons, Ltd.

Giesecke, K., Goldberg, L., and Ding, X. (2011). A top-down approach to multi-name credit. *Operations Research*, 59(2).

Glasserman, P. and Ruiz-Mata, J. (2006). Computing the credit loss distribution in the gaussian copula model: A comparison of methods. *Journal of Credit Risk*, 2(4).

Gordy, M. (2002). Saddlepoint approximation of creditrisk+. *Journal of Banking & Finance*, 26:1335–1353.

Gordy, M. B. (2003). A risk-factor model foundation for ratings-based bank capital rules. *Journal of Financial Intermediation*, 12(3):199–232.

Gordy, M. B. and Lutkebohmert, E. (2013). Granularity adjustment for regulatory capital assessment. *International Journal of Central Banking*, 9(3):33–71.

Hahnenstein, L. (2004). Calibrating the creditmetrics correlation concept - empirical evidence from germany. *Financial Markets and Portfolio Management*, 4(18):358–381.

He, J., Qian, J., and Strahan, P. E. (2012). Are all ratings created equal? the impact of issuer size on the pricing of mortgage-backed securities. *The Journal of Finance*, 67(6):2097–2137.

Huang, D., Kai, Y., Fabozzi, F. J., and Fukushima, M. (2007). An optimal design of collateralized mortgage obligation with pac-companion structure using dynamic cash reserve. *European Journal of Operational Research*, 177(2):1134–1152.

Jewan, D., Guo, R., and Witten, G. (2009). Optimal bespoke CDO design via NSGA-II. *Journal of Applied Mathematics and Decision Sciences*, Article ID 925169.

Jonard, M. (2021). Selection of securities for collateralized debt obligations contracts: An optimization approach applied to european SME loan portfolios. CEMS Master in International Management and business engineering, UCLouvain. (supervisor: F. Vrins).

Kaposty, F., Klein, P., Löderbusch, M., and Pfingsten, A. (2022). Loss given default in SME leasing. *Review of Managerial Science*, 16(5):1561–1597.

Kim, H. S. and Sohn, S. Y. (2010). Support vector machines for default prediction of smes based on technology credit. *European Journal of Operational Research*, 201(3):838–846.

Laurent, J.-P. and Sestier, M. (2016). Trading book and credit risk: How fundamental is the Basel review ? *Journal of Banking & Finance*, 73:211–223.

OECD (2023). OECD financing SMEs and entrepreneurs scoreboard: 2023 highlights. (36).

Owens, J. and Wilhelm, L. (2017). Alternative data transforming SME finance. Technical report, World Bank.

Petitt, B., Pinto, J., and Pirie, W. (2015). *Fixed Income Analysis*. CFA Institute Investment Series. Wiley, third edition edition.

Pykhtin, M. (2004). Multi-factor adjustment. *Risk*, 17:85–90.

Saunders, D., Xiouros, C., and Zenios, S. (2007). Credit risk optimization using factor models. *Annals of Operations Research*, 152:49–77.

Sirignano, J. A., Tsoukalas, G., and Giesecke, K. (2016). Large-scale loan portfolio selection. *Operations Research*, 64(6):1239–1255.

Stevenson, M., Mues, C., and Bravo, C. (2021). The value of text for small business default prediction: A deep learning approach. *European Journal of Operational Research*, 295(2):758–771.

Vasicek, O. (1991). Limiting loan loss probability distribution. Technical report.

Veremyev, A., Tsyurmasto, P., and Uryasev, S. (2012). Optimal structuring of CDO contracts: Optimization approach. *Journal of Credit Risk*, 8(4).

Veronesi, P. (2010). *Fixed Income Securities: Valuation, risk and risk management*. Wiley.

Vrins, F. (2009). Double t copula pricing of structured credit products: Practical aspects of a trustworthy implementation. *Journal of Credit Risk*.

Westerfeld, S. and Weber, F. (2009). Selecting credit portfolios for collateralized loan obligation transactions: A heuristic algorithm. *Journal of Credit Risk*, 5(4):65.

Wolsey, L. (2020). *Integer Programming*. John Wiley & Sons, Ltd.

On-line appendix

A Basel guidelines for loan capital requirements

Assume a loan i has maturity M_i, LGD λ_i, and 1-year probability of default PD_i (to distinguish from the probability of default before maturity noted p_i). We use Basel formula (BCBS, 2023) to compute the correlation ρ_i of firm i to the systematic risk factor:

$$\rho_i = 0.12\frac{1-e^{-50\text{PD}_i}}{1-e^{-50}} + 0.24\Big(1 - \frac{1-e^{-50\text{PD}_i}}{1-e^{-50}}\Big) . \tag{27}$$

The capital requirement K_i of a loan i per unit of notional N_i is:

$$K_i = \left(\lambda_i \Phi\left(\frac{\Phi^{-1}(\text{PD}_i)}{\sqrt{1-\rho_i}} + \sqrt{\frac{\rho_i}{1-\rho_i}}\Phi^{-1}(0.999)\right) - \text{PD}_i\lambda_i\right)\frac{1+b_i(M_i-2.5)}{1-1.5b_i} , \tag{28}$$

where the maturity adjustment b_i of a firm i is:

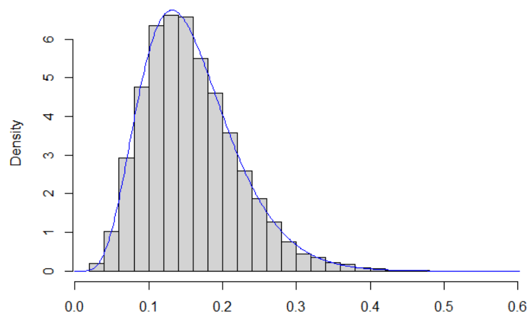
$$b_i = \left(0.11852 - 0.05478\ln(\text{PD}_i)\right)^2 . \tag{29}$$

B Histograms and descriptive statistics

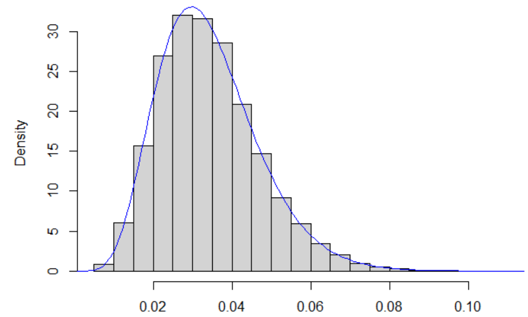
We draw 10,000 loans and report the histograms and descriptive statistics for the probability of default before maturity (p_i), the 1-year probability of default (PD_i), the LGD (λ_i), the interest rate (r_i), the maturity (M_i) and the notional (N_i).

Symbol	Min.	1^{st} Q.	Median	Mean	3^{rd} Q.	Max.	Std Dev.
p_i	0.022	0.112	0.149	0.158	0.195	0.575	0.063
PD_i	0.006	0.025	0.033	0.034	0.042	0.109	0.013
λ_i	0.007	0.325	0.498	0.498	0.672	0.998	0.223
M_i	1.485	4.206	4.940	5.006	5.713	10.601	1.116
r_i	0.005	0.013	0.018	0.019	0.024	0.069	0.008
N_i	0.007	2.443	4.251	5.093	6.895	28.947	3.572

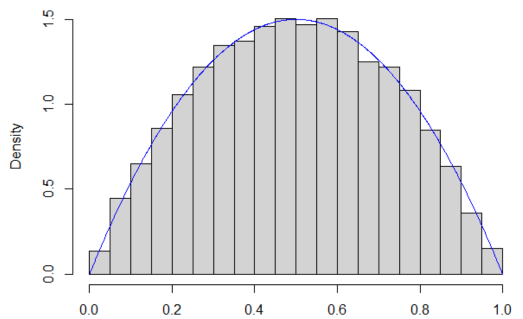
Table 8: Descriptive statistics of the loans' characteristics in a pool of size $I = 10,000$.



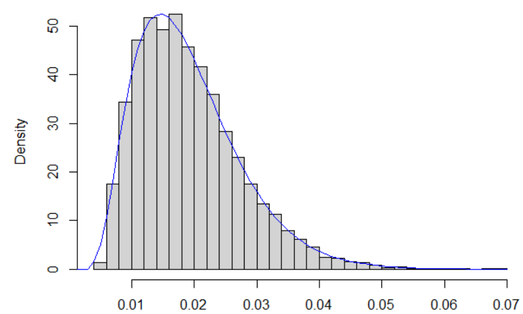
(a) Probability of default, p_i



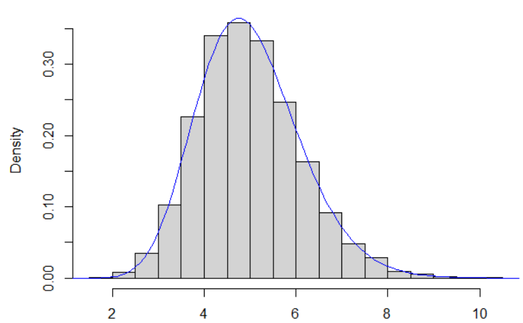
(b) Probability of default at 1 year, PD_i



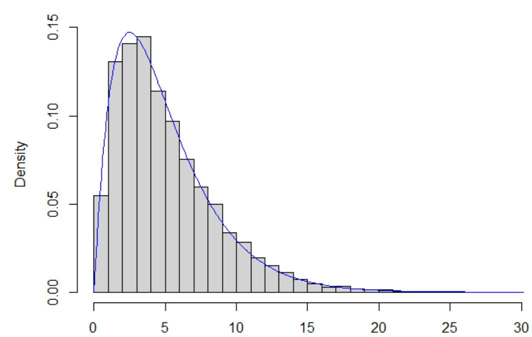
(c) Loss given default, λ_i



(d) Interest rate, r_i



(e) Maturity, M_i (in years)



(f) Notional, N_i (in million)

Figure 4: Histograms of the parameters of the loans in the pool and the exact densities (solid line) from which they are drawn.

C A continuous-time PSA model for the WAL

We consider that all the principal figures are expressed relative to the notional $N(\boldsymbol{w})$ of the portfolio $\mathcal{P}$. Note that the results depend on the chosen portfolio, hence, on $\boldsymbol{w}$. However, we drop the reference to $\boldsymbol{w}$ in this appendix for conciseness.

We note $c(t)$ the instantaneous interest rate at time t, such that $c(t)dt$ stands for the amount paid during the period $[t, t+dt[$ per unit of notional. The dynamics of the payments are governed by the instantaneous prepayment rate γ:

$$dc(t) = -\gamma(t)c(t)dt \quad \text{with solution} \quad c(t) = c(0)e^{-\Gamma(t)} \quad \text{and} \quad \Gamma(t) := \int_0^t \gamma(u)du \ ,$$

where $\gamma(.) \geq 0$. As standard in continuous time, we work with the continuous version of the coupon. The initial coupon value $c(0)$ is determined by imposing that the present value of the principal payments is equal to 100% of the portfolio notional when adding up all the coupons up to $T = \text{WAM}$, which represents the "maturity" of $\mathcal{P}$:

$$1 = \int_0^T c(0)e^{-\tilde{r}t}dt \quad \Rightarrow \quad c(0) = \frac{\tilde{r}}{1-e^{-\tilde{r}T}} \ .$$

The instantaneous interest amount $i(t)$ is given by the product of the prevailing interest rate $\tilde{r}$ with the outstanding principal $n(t)$:

$$i(t) = \tilde{r} \times n(t) \ .$$

The instantaneous scheduled principal (that is, the part of the payment that is not related to prepayment) at time t, noted $s(t)$, is the difference between the cashflow and the interests:

$$s(t) = c(t) - i(t).$$

The principal prepaid at time t, $p_\gamma(t)$, is given by the prepayment rate times the outstanding principal:

$$p_\gamma(t) = n(t) \times \gamma(t) \ .$$

Therefore, the part in the cashflow $c(t)$ related to principal repayments (scheduled and prepaid) at time t is:

$$p(t) = s(t) + p_\gamma(t) = c(t) + n(t) \times (\gamma(t) - \tilde{r}) \ .$$

This leads to the differential equation governing the dynamics of the instantaneous cashflow $n(t)$:

$$\begin{aligned} dn(t) &= -p(t)dt \\ n'(t) &= \underbrace{-c(t)}_{:=\alpha(t)-n(0)\beta(t)} + \underbrace{(\tilde{r} - \gamma(t))}_{:=\beta(t)} \times n(t) \ . \end{aligned}$$

This is a linear ODE whose solution is, imposing $n(0) = 1$,

$$\begin{aligned} n(t) &= 1 + e^{\int_0^t \beta(v)dv} \int_0^t \alpha(u) e^{-\int_0^u \beta(v)dv} du = 1 + \int_0^t \alpha(u) e^{\int_u^t \beta(v)dv} du \\ &= 1 + \int_0^t \big(\tilde{r} - \gamma(u) - c(u)\big) e^{\tilde{r}(t-u)} \frac{c(t)}{c(u)} du \\ &= 1 + c(t) e^{\tilde{r}t} \int_0^t \frac{\tilde{r} - \gamma(u) - c(u)}{c(u) e^{\tilde{r}u}} du \, . \end{aligned}$$

Using these expressions, the WAL of the portfolio is:

$$\mathrm{WAL} = -\int_0^T t \times n'(t) dt = \int_0^T t \times \left(c(t) + (\gamma(t) - \tilde{r}) n(t)\right) dt \, .$$

It can be checked that whatever the value chosen for the parameters $n(T) = 0$, i.e., that the outstanding notional is zero at maturity, as it should. Also, the sum of the principal payments agree is 100% of the portfolio's notional:

$$\int_0^T p(t)dt = -\int_0^T dn(t) = n(0) - n(T) = 1 - 0 = 1 \, .$$

Similarly, the Weighted Average Life of tranche j is:

$$\mathrm{WAL}_j := \frac{\int_{T'_{j-1}}^{T'_j} t \times \left(c(t) + (\gamma(t) - \tilde{r}) n(t)\right) dt}{S_j} \, .$$

The time T_j at which tranche $j \in \{1, 2, \ldots, J\}$ is fully repaid can be defined as:[18]

$$V_j(T'_j) = 0 \qquad \text{where} \qquad V_j(t) := D_j - (1 - n(t)) = 0 \qquad \text{and} \qquad T_0 := 0 \, .$$

Moreover, $V'_j(t) = n'(t) < 0$ on $[0, T]$ because $\tilde{r} > 0$. In addition, for such j, $0 < D_j \leq 1$ such that V_j is decreasing monotonously from $V_j(0) = D_j > 0$ to $V_j(T) = D_j - 1 \leq 0$, such that V_j admits one and only one root T_j on the interval $[0, T]$. We report below the analytical solutions for the dynamics of $n(t)$, the WAL of the pool and the WAL of the tranches under PSA model which assumes linearly increasing prepayment rates until a point t^* (usually 30 months) and a constant prepayment rate thereafter. Note that the solution for the WAL of a tranche is a nonlinear function of the weights and is semi-analytical (in the sense that it requires the use of a root-finding algorithm).

PSA model assumes linearly increasing prepayment rates until a point t* (usually 30 months). The instantaneous prepayment rate is thus a piecewise linear function:

$$\gamma(t) = \gamma \times \min(t, t^*) \, .$$

[18] Note that $V_j(t)$ should not be interpreted as the outstanding notional of tranche j, which is $S_j(t) = \max(0 \min(S_j, D_j - n(t) + 1))$. However, T_j is the time at which the total notional repaid reaches D_j, and coincides with the first hitting time of $S_j(t)$ to 0: $T'_j = \inf_t S_j(t) = 0$.

We have:

$$c(t) = c(0) \times \begin{cases} e^{-\frac{\gamma}{2}t^2} & \text{if } t \leq t^*, \\ e^{\gamma t^*(\frac{t^*}{2}-t)} & \text{if } t \geq t^*. \end{cases}$$

$$n(t) = \Big(1 + \frac{c(0)}{\tilde{r}}(e^{-\tilde{r}t} - 1)\Big) \times \begin{cases} e^{-\frac{-\gamma}{2}t^2+\tilde{r}t} & \text{if } t \leq t^*, \\ e^{-\frac{\gamma}{2}{t^*}^2+\tilde{r}t-\gamma t^* t} & \text{if } t \geq t^*. \end{cases}$$

The Weighted Average Life is the sum of two integrals,

$$\begin{aligned} \text{WAL} &= \int_0^{t^*} t\Big(c(0)e^{-\frac{\gamma}{2}t^2} + (\gamma t - \tilde{r})e^{-\frac{-\gamma}{2}t^2+\tilde{r}t}\big(1 + \frac{c(0)}{\tilde{r}}(e^{-\tilde{r}t} - 1)\big)\Big)dt \\ &\quad + \int_{t^*}^{T} t\Big(c(0)e^{\gamma t^*(\frac{t^*}{2}-t)} + (\gamma t^* - \tilde{r})e^{-\frac{\gamma}{2}{t^*}^2+\tilde{r}t-\gamma t^* t}\big(1 + \frac{c(0)}{\tilde{r}}(e^{-\tilde{r}t} - 1)\big)\Big)dt \end{aligned}$$

whose solution is:

$$\begin{aligned} \text{WAL} &= \Big(1 - \frac{c(0)}{\tilde{r}}\Big)\Big\{-t^* e^{-\frac{\gamma}{2}{t^*}^2+\tilde{r}t^*} + \sqrt{\frac{2\pi}{\gamma}}e^{\frac{\tilde{r}^2}{2\gamma}}\left(\Phi\left(\sqrt{\gamma}t^* - \frac{\tilde{r}}{\sqrt{\gamma}}\right) - \Phi\left(\frac{-\tilde{r}}{\sqrt{\gamma}}\right)\right)\Big\} \\ &\quad + \frac{c(0)}{\tilde{r}}\left(-t^* e^{-\frac{\gamma}{2}{t^*}^2} + \sqrt{\frac{2\pi}{\gamma}}\left(\Phi(\sqrt{\gamma}t^*) - \frac{1}{2}\right)\right) \\ &\quad + e^{\frac{\gamma}{2}{t^*}^2}\Big(\Big(\frac{c(0)}{\tilde{r}} - 1\Big)\Big(Te^{(\tilde{r}-\gamma t^*)T} - t^* e^{(\tilde{r}-\gamma t^*)t^*} - \frac{e^{(\tilde{r}-\gamma t^*)T} - e^{(\tilde{r}-\gamma t^*)t^*}}{(\tilde{r} - \gamma t^*)}\Big) \\ &\quad + \frac{c(0)}{\gamma t^*}\Big(1 + \frac{\gamma t^* - \tilde{r}}{\tilde{r}}\Big)\Big(t^* e^{-\gamma {t^*}^2} - Te^{-\gamma t^* T} + \frac{e^{-\gamma {t^*}^2} - e^{-\gamma t^* T}}{\gamma t^*}\Big), \end{aligned}$$

where Φ is the standard normal cdf. There is no closed-from expression for T_j but a root-finding algorithm can be used. The expression of the Weighted Average Life of a tranche j depends of T_j. To keep the notations concise, we define $T_j^\gamma := \sqrt{\gamma}T_j$. If $T_j \leq t^*$:

$$\begin{aligned} WAL_j S_j &= \Big(1 - \frac{c(0)}{\tilde{r}}\Big)\Big\{ - T_j e^{-\frac{(T_j^\gamma)^2}{2}+\tilde{r}T_j} + T_{j-1}e^{-\frac{(T_{j-1}^\gamma)^2}{2}+\tilde{r}T_{j-1}} + \sqrt{\frac{2\pi}{\gamma}}e^{\frac{\tilde{r}^2}{2\gamma}}\left(\Phi\left(T_j^\gamma - \frac{\tilde{r}}{\sqrt{\gamma}}\right)\right. \\ &\quad \left.-\Phi\left(T_{j-1}^\gamma - \frac{\tilde{r}}{\sqrt{\gamma}}\right)\right)\Big\} + \frac{c(0)}{\tilde{r}}\Big\{ - T_j e^{-\frac{(T_j^\gamma)^2}{2}} + T_{j-1}e^{-\frac{(T_{j-1}^\gamma)^2}{2}} + \sqrt{\frac{2\pi}{\gamma}}\big(\Phi(T_j^\gamma) - \Phi(T_{j-1}^\gamma)\big)\Big\}. \end{aligned}$$

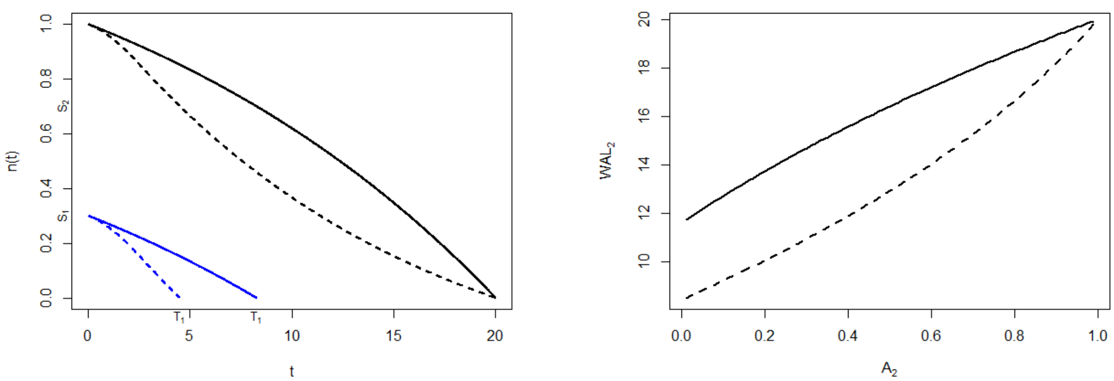
In the case where $T_{j-1} \leq t^* \leq T_j$:

$$\begin{aligned} \text{WAL}_j S_j &= \Big(1 - \frac{c(0)}{\tilde{r}}\Big)\Big\{T_{j-1}e^{-\frac{{T_{j-1}^\gamma}^2}{2}+\tilde{r}T_{j-1}} - t^* e^{-\frac{\gamma}{2}{t^*}^2+\tilde{r}t^*} + \sqrt{\frac{2\pi}{\gamma}}e^{\frac{\tilde{r}^2}{2\gamma}}\left(\Phi\left(\sqrt{\gamma}t^* - \frac{\tilde{r}}{\sqrt{\gamma}}\right) - \Phi\left(T_{j-1}^\gamma - \frac{\tilde{r}}{\sqrt{\gamma}}\right)\right)\Big\} \\ &\quad + \frac{c(0)}{\tilde{r}}\Big(- t^* e^{-\frac{\gamma}{2}{t^*}^2} + T_{j-1}e^{-\frac{{T_{j-1}^\gamma}^2}{2}} + \sqrt{\frac{2\pi}{\gamma}}\big(\Phi(\sqrt{\gamma}t^*) - \Phi(T_{j-1}^\gamma)\big)\Big) \\ &\quad + e^{\frac{\gamma}{2}{t^*}^2}\Big(\Big(\frac{c(0)}{\tilde{r}} - 1\Big)\Big(T_j e^{(\tilde{r}-\gamma t^*)T_j} - t^* e^{(\tilde{r}-\gamma t^*)t^*} - \frac{e^{(\tilde{r}-\gamma t^*)T_j} - e^{(\tilde{r}-\gamma t^*)t^*}}{(\tilde{r} - \gamma t^*)}\Big) \\ &\quad + \frac{c(0)}{\gamma t^*}\Big(1 + \frac{\gamma t^* - \tilde{r}}{\tilde{r}}\Big)\left(t^* e^{-\gamma {t^*}^2} - T_j e^{-\gamma t^* T_j} + \frac{e^{-\gamma {t^*}^2} - e^{-\gamma t^* T_j}}{\gamma t^*}\right). \end{aligned}$$

Lastly, in the case where $t^* \leq T_{j-1}$:

$$\begin{aligned}\mathrm{WAL}_j S_j &= e^{\frac{\gamma}{2}t^{*2}} \left\{ \left(\frac{c(0)}{\tilde{r}} - 1\right)\left(T_j e^{(\tilde{r}-\gamma t^*)T_j} - T_{j-1}e^{(\tilde{r}-\gamma t^*)T_{j-1}} - \frac{e^{(\tilde{r}-\gamma t^*)T_j} - e^{(\tilde{r}-\gamma t^*)t^*}}{(\tilde{r}-\gamma T_{j-1})}\right)\right. \\ &\left. + \frac{c(0)}{\gamma t^*}\left(1 + \frac{\gamma t^* - \tilde{r}}{\tilde{r}}\right)\left(T_{j-1}e^{-\gamma t^* T_{j-1}} - T_j e^{-\gamma t^* T_j} + \frac{e^{-\gamma t^* T_{j-1}} - e^{-\gamma t^* T_j}}{\gamma t^*}\right)\right\} .\end{aligned}$$

Figure 5 illustrates the effect of prepayment on a tranche's amortization.



(a) Evolution of the notional of the pool (black, top) and of the junior tranche (blue, bottom): $D_1 = A_2 = 0.3$.

(b) Weighted Average Life of the senior tranche as a function of the attachment point of the senior tranche.

Figure 5: Comparison between 100% PSA model (dashed line) and no prepayment case (solid line) for a pool made of two tranches ($J = 2$), with $T = 20$, $r = 5\%$ and $n(0) = 100\%$. The labels T_1 indicates the time at which tranche 1 is repaid following the deterministic top-down model.

D Impact of granularity adjustment

The purpose of this section is to analyze the impact of the LP assumption when working with pools of finite size. We consider heterogeneous pools since, otherwise, all the loans are exchangeable and the selection problem becomes trivial. The exact distribution is computed using conditional independence combined with Laplace transform. The exact moment generating function of the relative loss is

$$\psi_{L(\boldsymbol{w})}(u) = \mathbb{E}\left(\prod_{i=1}^{I} \psi_{B(p_i(Z))}(uw_i\lambda_i n_i(\boldsymbol{w}))\right) = \mathbb{E}\left(\prod_{i=1}^{I}\left(q_i(Z) + \mathrm{e}^{uw_i\lambda_i n_i(\boldsymbol{w})}\, p_i(Z)\right)\right),$$

with $n_i(\boldsymbol{w}) = N_i/N(\boldsymbol{w})$, $q_i(Z) = 1 - p_i(Z)$ is the conditional survival probability of I given Z. This expectation can be evaluated using a quadrature scheme, as above. Inverting the characteristic function using inverse FFT yields the distribution of L, which can then be inserted in (25) to get the ELoT.

Tables 9 and 10 respectively display a comparison of the algorithms' performance after a maximum of 1 hour running time (see Section 6.2). The objective function features the ELoT being evaluated using either the LP assumption (F, where the ELoT is evaluated using (5)) or the finite-size pool via the Laplace transform approach discussed above (F^L, where the ELoT is evaluated using (25)). The last two lines of these tables display $F^L(\boldsymbol{w}^*)$ where $\boldsymbol{w}^*$ is found by relying on the exact EL_j (25) in the iterations of programs 1 and 2. Clearly, the ranking of the programs' performance is not changed whether ones evaluates the terminal value of the objective function using F or F^L. Secondly, programs 3 and 4 perform better than programs 1 and 2, even when the exact distribution is used in the optimization procedure. In fact, not using the LP approximation in programs 1 and 2 is detrimental, in the sense that $F^L(\boldsymbol{w}^*)$ is smaller when $\boldsymbol{w}^*$ is found by using programs 1 and programs 2 relying on the LP approximation (5). In Application 1 featuring 1,000 loans, for instance, applying programs 1 and 2 on the exact ELoT (25) yields 6.7578 and 5.4913, respectively, while the solutions found using the same programs but using the LP approximation of the ELoT (5) in the iterations yields 4.5589 and 4.4202. The same conclusion applies to every other configuration (Application 1 with 10,000 loans and Application 2 for both pool sizes). This can be explained by the fact that evaluating F^L is way more costly than evaluating F. There are therefore way less iterations when optimizing F^L in the 1 hour running time.

	$1,000$ loans		$10,000$ loans	
	$F(\boldsymbol{w}^*)$	$F^L(\boldsymbol{w}^*)$	$F(\boldsymbol{w}^*)$	$F^L(\boldsymbol{w}^*)$
Heuristics (EL)	2.3103	2.3586	2.1534	2.0753
Heuristics (M)	9.1633	9.5012	2.1534	2.0753
Heuristics (K)	14.2832	14.6687	14.1319	14.2227
Heuristics (r)	14.1423	14.5344	14.0525	14.1388
Program 1 on F	4.3503	4.5589	6.4991	6.5057
Program 2 on F	4.2104	4.4202	5.3440	5.3968
Program 3 on F	**2.0764**	**2.1611**	**1.8895**	**1.9745**
Program 4 on F	2.1072	2.1866	1.8913	2.0151
Program 1 on F^L		6.7578		7.0079
Program 2 on F^L		5.4913		7.1217

Table 9: Application 1. Terminal values of the objective function found by all programs/heuristics after a computation time capped at 1 hour. Columns labelled $F(\boldsymbol{w}^*)$ refer to the objective function used in the approximate problems taken from tables 3 and 4, while columns $F^L(\boldsymbol{w}^*)$ give the value of the exact objective function (i.e., evaluated using inverse Laplace) evaluated at the final solution. The last two lines show the value of the exact objective function when applying Program 1 and Program 2 directly to the finite-size loss distribution.

E t-copula setup

We consider the t-copula model of Section 7.2. The proposed quadrature scheme for the LP ELoT approximation (21) changes to

$$\text{EL}_j(\boldsymbol{w}) \approx \text{EL}_j^P(\boldsymbol{w}) := \frac{\pi}{2} \sum_{k_1=1}^{P_1} \sum_{k_2=1}^{P_2} \pi_{k_1}^x \pi_{k_2}^v \left[\frac{\sum_{i=1}^{I} w_i \underline{n}_i \lambda_i p_i(x_{k_1}, v_{k_2}) - A_j}{S_j} \right] \frac{\phi(x_{k_1})}{\cos^2(\tilde{u}_{k_1})} ,$$

where $\underline{n}_i = N_i/\underline{N}$, (x_k, π_k^x) and (v_k, π_k^v) are the (nodes,weights) pairs associated with the Gauss-Legendre and the Chi-squared quadrature, respectively, and $(\tilde{u}_k, x_k) = (u_k\pi/2, \tan(\tilde{u}_k))$, as in (24). For the latter, we rely on the `R` command `gauss.quad.prob(`P_2`, dist = "gamma", `ν`/2, 2)`. We

	1,000 loans		10,000 loans	
	$F(\boldsymbol{w}^*)$	$F^L(\boldsymbol{w}^*)$	$F(\boldsymbol{w}^*)$	$F^L(\boldsymbol{w}^*)$
Heuristics (EL)	0.0522	0.0535	0.0542	0.0543
Heuristics (M)	0.0522	0.0535	0.0534	0.0535
Heuristics (K)	0.0542	0.0550	0.0538	0.0539
Heuristics (r)	0.0571	0.0580	0.0560	0.0561
Program 1 on F	0.0471	0.0482	0.0505	0.0506
Program 2 on F	0.0453	0.0464	0.0484	0.0484
Program 3 on F	0.0417	0.0428	0.0416	0.0416
Program 4 on F	**0.0414**	**0.0424**	**0.0412**	**0.0414**
Program 1 on F^L		0.0513		0.0518
Program 2 on F^L		0.0486		0.0517

Table 10: Application 2. Terminal values of the objective function found by all programs/heuristics after a computation time capped at 1 hour. Columns labelled $F(\boldsymbol{w}^*)$ refer to the objective function used in the approximate problems taken from tables 5 and 6, while columns $F^L(\boldsymbol{w}^*)$ give the value of the exact objective function (i.e., evaluated using inverse Laplace) evaluated at the final solution. The last two lines show the value of the exact objective function when applying Program 1 and Program 2 directly to the finite-size loss distribution.

set the number of points for the Gaussian and Gamma quadrature to $P_1 = 100$ and $P_2 = 32$, respectively. The degrees of freedom is set to $\nu = 10$ and the other parameters remain (PD, correlation, etc) are unchanged.

Table 11 displays the results of each heuristic and each program considering two datasets of 1,000 and 10,000 loans, and our two applications above. The computation time of each program is capped to 1 hour. We fix the number of clusters $Q = 200$ for the 1,000 loans dataset and $Q = 2,500$ for the 10,000 loans dataset. One can see that the conclusions of Section 6.2 remain essentially unchanged compared to the Gaussian copula. In particular, programs 3 and 4 lead to a substantial improvement compared to programs 1 and 2.

	Application 1		Application 2	
	1,000 loans	10,000 loans	1,000 loans	10,000 loans
Heuristics (EL)	4.7345	4.7302	0.0700	0.0729
Heuristics (M)	12.7291	12.3145	0.0622	0.0616
Heuristics (K)	18.8449	18.6635	0.0614	0.0610
Heuristics (r)	18.4959	18.4375	0.0643	0.0632
Program 1	9.6900	10.3423	0.0606	0.0615
Program 2	9.2358	10.7126	0.0587	0.0618
Program 3	**4.4662**	**4.1045**	**0.0542**	**0.0551**
Program 4	4.5233	4.1097	0.0544	0.0554

Table 11: Terminal values of the objective function found by all programs/heuristics after a computation time capped at 1 hour for each application and for each dataset in a t-copula setup. We set the number of clusters to $Q = 200$ and $Q = 2,500$ for Application 1 and 2, respectively.

F Multi-industry model

The purpose of this section is to illustrate how to use our framework in a setup featuring various systematic factors, and to discuss the corresponding results. The proposed setup relies on the comparable factor model mentioned in Section 7.3. In this exercise, we consider the multi-sector setup of Düllmann and Masschelein (2007), where each loan in a pool $\mathcal{S}$ belongs to one sector out of $S = 11$. Each sector is characterized by a specific factor, Y_s. The S sector factors can be correlated, to reflect sector correlations. This is achieved by defining the sector factors as linear combinations of S independent standard factors $\boldsymbol{Z}^{\perp} = (Z_1^{\perp}, \ldots, Z_S^{\perp})$:

$$Y_s = \boldsymbol{\alpha}_s^{\top} \boldsymbol{Z}^{\perp} , \quad \|\boldsymbol{\alpha}_s\| = 1 , \quad s \in \{1, 2, \ldots, S\} ,$$

where the norm constraint guarantees that the sector factors are standardized, too. The correlation between two sector factors Y_r, Y_t is $\boldsymbol{\alpha}_r^{\top} \boldsymbol{\alpha}_t$. Assuming a S-by-S matrix $\boldsymbol{R}$ of the sector correlations, the vectors $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_S$ can be computed by Cholesky decomposition of $\boldsymbol{R}$ such that $R_{rt} = \boldsymbol{\alpha}_r^{\top} \boldsymbol{\alpha}_t$. One obtains a S-factor model by defining the standard i.i.d. variables $\epsilon_1^S, \ldots, \epsilon_I^S$, also independent from $\boldsymbol{Z}^{\perp}$, letting $s(i)$ denote the sector of loan i and introducing creditworthiness variables as

$$Z_i^S = \sqrt{\rho_{s(i)}} Y_{s(i)} + \sqrt{1 - \rho_{s(i)}} \epsilon_i^S . \tag{30}$$

In the model of Düllmann and Masschelein (2007), all the variables $Z_s^{\perp}$ and ϵ_i^S are all i.i.d. standard normal. As for the calibration, we consider as $\boldsymbol{R}$ the sector correlation matrix given in Table 3 of Düllmann and Masschelein (2007), and set $\sqrt{\rho_{s(i)}} = 0.39$, as suggested in Hahnenstein (2004) for German SMEs.

We then follow (Pykhtin, 2004) and proxy this S-factor model using a CSF model. We introduce a vector $\epsilon_1, \ldots, \epsilon_I$ of i.i.d. standard normal variables, independent from $\boldsymbol{Z}^{\perp}$, and define a new set of creditworthiness variables as

$$Z_i = \sqrt{\gamma_i} Z^* + \sqrt{1 - \gamma_i} \epsilon_i , \tag{31}$$

where

$$Z^* := \boldsymbol{\beta}^{\top} \boldsymbol{Z}^{\perp} \quad \text{for some} \quad \boldsymbol{\beta} \in \mathbb{R}^S \quad \text{satisfying} \quad \|\boldsymbol{\beta}\| = 1 .$$

Clearly, Z^* is a standard normal variable independent from the ϵ_i's, such that (31) defines a 1FGC model, similar to (1). In order for the latter to proxy the multi-factor model (30) well, one must choose the loading factors γ_i in (31) in an appropriate way given $\boldsymbol{\beta}$. Pykhtin (2004) proposes to impose that the conditional default probability of each loan given Z^* is the same under the S-factor model and its comparable one-factor version, that is,

$$\mathbb{P}(Z_i^S \leq \Phi^{-1}(p_i) | Z^*) = \mathbb{P}(Z_i \leq \Phi^{-1}(p_i) | Z^*) \quad \Rightarrow \quad \gamma_i = \rho_{s(i)} (\boldsymbol{\alpha}_{s(i)}^{\top} \boldsymbol{\beta})^2 .$$

The vector $\boldsymbol{\beta}$ is chosen in order to insure that Z^* is indeed able to optimally resume the multi-factor structure. Here again, Pykhtin (2004) proposes an heuristic to estimate the value-at-risk at the a given confidence. Inspired by the latter, we set

$$\beta_s = \frac{\sum_{i=1}^{I} d_i \alpha_{s(i),s}}{\sqrt{\sum_{s=1}^{S} (\sum_{i=1}^{I} d_i \alpha_{s(i),s})^2}} \quad \text{where} \quad d_i = N_i \lambda_i p_i . \tag{32}$$

This yields a 1FGC model that mimics the S-factor model (30); it suffices to plug $a_i = \sqrt{\gamma_i}$ in (1). We can then simply compute the LP approximation of the ELoT using (5).

We also include in our experiment some diversification constraints, requiring that at least 50% of the notional of each sector is selected. To this purpose, we constrain the clustering step so that each cluster contains loans belonging to a same sector. All parameters remain unchanged.

Table 12 displays the results of each heuristic and each program considering two datasets of 1,000 and 10,000 loans, and our two applications above. As before, we cap the computation time to 1 hour and fix the number of clusters to $Q = 200$ when $I = 1,000$ and $Q = 2,500$ when $I = 10,000$. To have a more reliable estimation of the objective function at the optimum, we estimate the ELoT in the multifactor model model via Monte-Carlo featuring 250,000 simulations. Those simulations takes approximately 8 minutes when $I = 1,000$ and 83 minutes when $I = 10,000$. One can see that the conclusions of Section 6.2 remain essentially unchanged compared to the 1FGC, in the sense that optimizing the objective function using the comparable single-factor indeed helps to enhance the true objective function, featuring the ELoT of the S-factor model. In particular, programs 3 and 4 lead to a substantial improvement of the solution found with programs 1 and 2.

	Application 1				Application 2			
	1,000 loans		10,000 loans		1,000 loans		10,000 loans	
	CSF	MF	CSF	MF	CSF	MF	CSF	MF
Program 1	9.1574	9.9119	13.7628	13.8820	0.0674	0.0705	0.0718	0.0733
Program 2	8.1563	8.8514	12.1763	12.4669	0.0659	0.0689	0.0685	0.0696
Program 3	**5.1050**	**6.1082**	**5.4857**	**6.6824**	**0.0605**	0.0663	0.0590	0.0644
Program 4	5.4866	6.2162	5.5532	6.7473	0.0613	**0.0652**	**0.0589**	**0.0627**

Table 12: Objective function at the optimum found by all programs after max. 1 hour for each application and for each dataset in a multi-factor setup. 'CSF' shows the terminal objective function for which the final the ELoT is computed using the comparable single-factor model (as in the optimization programs). 'MF' refers to solutions for which the final ELoT is evaluated with the true multi-factor model via Monte Carlo with 250,000 simulations.