

Efficient simulations for a Bernoulli mixture model of portfolio credit risk

İsmail Başoğlu¹ · Wolfgang Hörmann² · Halis Sak^{3,4}

Published online: 3 June 2016
© Springer Science+Business Media New York 2016

Abstract We consider the problem of calculating tail loss probability and conditional excess for the Bernoulli mixture model of credit risk. This is an important problem as all credit risk models proposed in literature can be represented as Bernoulli mixture models. Thus, we deviate from the efficient simulation of credit risk literature in that we propose an efficient simulation algorithm for this general Bernoulli mixture model in contrast to previous works that focus on specific credit risk models like CreditRisk⁺ or Credit Metrics. The algorithm we propose is a combination of stratification, importance sampling based on cross-entropy, and inner replications using the geometric shortcut method. We evaluate the efficiency of our general method considering three different examples: CreditRisk⁺ and two of the latent variable models, the Gaussian and the *t*-copula model. Numerical results suggest that the proposed general algorithm is more efficient than the benchmark methods for these specific models.

Keywords Credit risk · Bernoulli mixture model · Copula models · Geometric shortcut · Cross-entropy method · Stratification

Mathematics Subject Classification 91G40 · 91G60

✉ İsmail Başoğlu
ismailbsgl@gmail.com

Wolfgang Hörmann
hormannw@boun.edu.tr

Halis Sak
halis.sak@gmail.com

¹ School of Economics and Administrative Sciences, İstanbul Kemerburgaz University, Bağcılar, İstanbul, Turkey

² Department of Industrial Engineering, Boğaziçi University, Bebek, İstanbul, Turkey

³ Department of Mathematical Sciences, Xi'an Jiaotong-Liverpool University, Suzhou, China

⁴ Department of Industrial and Systems Engineering, Yeditepe University, Kayışdağı, İstanbul, Turkey

1 Introduction

The essence of control in the financial world relies on how well the used models mimic reality and on the precision of the computational methods used. Monte Carlo simulation is often one of the best alternatives as it leads to confidence intervals on the pin-point values. The subject of this paper is to develop a simulation methodology to quantify credit risk, risks related with the obligors' default. For quantification of credit risk, two of the credit risk models that are used intensively in the industry are Credit Risk Metrics of JP Morgan and Credit Risk⁺ of [Credit Suisse Financial Products \(1997\)](#).

Although credit risk simulations seem to be quite easy, as in each repetition we sum the exposures of the defaulted obligors, rare-event settings are problematic as the number of repetitions should increase enormously to get meaningful estimates of the risk. This is not desired as it degrades the efficiency of the simulation. Efficient Monte Carlo methods for the Gaussian copula model of [Gupton et al. \(1997\)](#) (see, e.g., [Glasserman and Li 2005](#); [Sak and Hörmann 2012](#)), for the t -copula model (see [Bassamboo et al. 2008](#); [Chan and Kroese 2010](#); [Sak 2010](#)) and for the mixed Poisson model (see [Glasserman and Li 2003](#)) of credit risk were proposed in the literature. In this paper, we propose a new method to estimate tail loss probability and conditional excess for the Bernoulli mixture model of credit risk.

Our main contribution is to present a new algorithm to measure credit risk under the general Bernoulli mixture models. These models are credited as more convenient for statistical fitting purposes in [McNeil et al. \(2005\)](#) and all credit risk models proposed in literature can be represented as Bernoulli mixture models. We could not find any simulation method for the general Bernoulli mixture models in the literature, and it is not obvious how to find an equivalent latent variable model for a given Bernoulli mixture model.

In our general algorithm, we implement a combination of stratification and importance sampling for the simulation of the random variables that introduce the dependence across obligors. We use an importance sampling strategy based on the cross-entropy method and increase the probability of rare defaults. The subsequent stratification enhances the variance reduction by optimal sample allocation throughout strata. The remaining source of the variance is the simulation of obligors' defaults, for which we employ inner replications using the geometric shortcut method (see [Sak and Hörmann 2012](#)). The new algorithm yields reasonable variance reduction over the existing methods designed for specific risk models.

Another important contribution of this paper is the efficient estimation of conditional excess for the total loss of a credit portfolio. Conditional excess has a ratio estimator, i.e., the ratio of two different estimators. Therefore, when the stratification is implemented, the optimal sample allocation sizes that minimize the variance of the conditional excess must be derived differently than the ones which minimize the variance of a regular estimator (see [Başoğlu and Hörmann 2014](#)). To our knowledge, we are the first to use a stratification algorithm for the estimation of conditional excess for credit portfolios. Most papers in literature just consider implementing importance sampling for this problem and the proposed importance sampling density is usually the same as the one used for tail loss probability estimation (see, e.g., [Glasserman 2005](#)). In our algorithm, stratification allows us to derive formulas for sample allocations that minimize the variance of the conditional excess estimate. These formulas are useful for any simulation applying stratified ratio estimators.

The paper is organized as follows: Sect. 2 gives an overview of the Bernoulli mixture models. In Sect. 3, we explain the implementation details of inner replications of geometric shortcut, importance sampling based on cross-entropy, and stratification, and how these methods can be combined in simulating tail loss probabilities. Section 4 extends the use of

the same methodology to conditional excess simulation and explains the stratification of the ratio estimator. Finally, in Sect. 5, we present our numerical results for some credit portfolio examples whereas Sect. 6 provides final comments.

Note that, in this paper, vectors and matrices are set in bold to enhance readability.

2 Bernoulli mixture model

Before discussing the model, we give the notation used throughout the paper. There are J obligors in the portfolio and Y_j denotes the Bernoulli random variable for the j th obligor (1 if j th obligor defaults, 0 otherwise). The marginal probability that the j th obligor defaults is p_j and c_j denotes the loss resulting from the default of the j th obligor. The total loss of the portfolio is calculated as $L = \sum_{j=1}^J c_j Y_j$. We only consider a fixed horizon, over which we are interested in the values of tail loss probability $y = P(L > \tau) = E[\mathbb{1}_{\{L > \tau\}}]$ and conditional excess $r = E[L|L > \tau]$ for a fixed threshold value τ .

Following McNeil et al. (2005); given the D -dimensional ($D < J$) random vector $\Psi = (\Psi_1, \dots, \Psi_D)'$, the random vector $\mathbf{Y} = (Y_1, \dots, Y_J)'$ follows a Bernoulli mixture model if there are functions $p_j : \mathbb{R}^D \rightarrow [0, 1]$, $j = 1, \dots, J$, such that, conditional on Ψ , the components of $\mathbf{Y} = (Y_1, \dots, Y_J)'$ are independent Bernoulli random variables. We define $p_j(\Psi)$ as conditional default probabilities for a given Ψ vector: $p_j(\Psi) = P(Y_j = 1|\Psi)$. For $(\varepsilon_1, \dots, \varepsilon_J)'$ in $\{0, 1\}^J$,

$$P(Y_1 = \varepsilon_1, \dots, Y_J = \varepsilon_J|\Psi) = \prod_{j=1}^J p_j(\Psi)^{\varepsilon_j} (1 - p_j(\Psi))^{(1-\varepsilon_j)}. \tag{1}$$

The unconditional distribution of \mathbf{Y} is found by integrating (1) over the distribution of Ψ .

We consider three specific examples of this general model to test our method in Sect. 5. The first one is CreditRisk⁺ model which is represented as a Bernoulli mixture model (see, Frey and McNeil 2002). $\Psi = (\Psi_1, \dots, \Psi_D)'$ are independent Gamma random variables with shape parameters $\alpha_d = \sigma_d^{-2}$ and scale parameters $\beta_d = \sigma_d^2$ for $d = 1, \dots, D$.

Furthermore, the conditional default probabilities for a given Ψ vector are

$$p_j(\Psi) = 1 - \exp(-a_{j0} - a_{j1}\Psi_1 - \dots - a_{jD}\Psi_D), \quad j = 1, \dots, J, \tag{2}$$

where a_{j0}, \dots, a_{jD} are positive coefficients.

The second example is the Gaussian copula model which is a multi-dimensional probit-normal mixing distribution (see McNeil et al. 2005, p. 354). The Gaussian copula model introduces a multivariate normal vector $\mathbf{Z} = (Z_1, \dots, Z_J)'$ of latent variables to obtain dependence across obligors. The relationship between the default indicators and the latent variables is described by $Y_j = \mathbb{1}_{\{Z_j > z_j\}}$, $j = 1, \dots, J$, where Z_j follows standard normal distribution, $z_j = \Phi^{-1}(1 - p_j)$ and Φ^{-1} denotes the inverse of the cumulative distribution function (CDF) of the standard normal distribution. Obviously, the choice of the threshold value, z_j , implies that $P(Y_j = 1) = p_j$.

The correlations among the Z_j values are modeled by defining

$$Z_j = b_j \epsilon_j + a_{j1}\Psi_1 + \dots + a_{jD}\Psi_D, \quad j = 1, \dots, J, \tag{3}$$

where ϵ_j and Ψ_1, \dots, Ψ_D are independent standard normal random variables with $b_j^2 + a_{j1}^2 + \dots + a_{jD}^2 = 1$. While, Ψ_1, \dots, Ψ_D are systematic risk factors affecting all obligors, ϵ_j is the idiosyncratic risk factor affecting only obligor j . Furthermore, a_{j1}, \dots, a_{jD} are

constant and non-negative factor loadings, assumed to be known. Thus, given the vector $\Psi = (\Psi_1, \dots, \Psi_D)'$, we have the conditional default probabilities

$$p_j(\Psi) = P(Y_j = 1|\Psi) = \Phi\left(\mathbf{a}_j\Psi + \Phi^{-1}(p_j)b_j^{-1}\right), \quad j = 1, \dots, J, \quad (4)$$

where $\mathbf{a}_j = (a_{j1}, \dots, a_{jD})$.

The last example is the t -copula model in which latent variables follow multivariate t -distribution instead of multivariate normal distribution. The model that has been widely used (see, e.g., Bassamboo et al. 2008; Kang and Shahabuddin 2005) is

$$T_j = \left(b_j\epsilon_j + \sum_{d=1}^{D-1} a_{jd}\Psi_d\right) (\Psi_D/\eta)^{-1/2}, \quad j = 1, \dots, J,$$

where the definitions of $\Psi_1, \dots, \Psi_{D-1}, \epsilon_j, a_{j1}, \dots, a_{j(D-1)}$ and b_j are the same as in (3), and Ψ_D denotes a chi-square random variable with η degrees of freedom that is independent of $\Psi_1, \dots, \Psi_{D-1}$ and ϵ_j . The relationship between the default indicators and the latent variables is described by $Y_j = \mathbb{1}_{\{T_j > t_j\}}$. Since T_j is a t -distributed random variable, to preserve the marginal default probabilities, we select $t_j = F_\eta^{-1}(1 - p_j)$ where F_η denotes the CDF of the t -distribution with η degrees of freedom. Finally, given the vector $\Psi = (\Psi_1, \dots, \Psi_D)'$, we have the conditional default probabilities

$$p_j(\Psi) = P(Y_i = 1|\Psi) = \Phi\left(\frac{\tilde{\mathbf{a}}_j\tilde{\Psi} - \sqrt{\Psi_D/\eta}F_\eta^{-1}(1 - p_j)}{b_j}\right), \quad j = 1, \dots, J, \quad (5)$$

where $\tilde{\mathbf{a}}_j = (a_{j1}, \dots, a_{j(D-1)})$ and $\tilde{\Psi} = (\Psi_1, \dots, \Psi_{D-1})'$.

3 Efficient simulation of tail loss probabilities

In order to simulate tail loss probabilities under the Bernoulli mixture model, in each replication of the simulation algorithm, we need to generate the input vector Ψ with density $f(\Psi; \mathbf{u})$ where \mathbf{u} denotes the set of parameters of the specific model. Then, we calculate the conditional default probabilities $p_j(\Psi)$, $j = 1, \dots, J$ again for the specific model. Once, $p_j(\Psi)$ is calculated, it can be used to generate the default indicator Y_j of the j th obligor. Finally, we compute the total loss L of the portfolio and the response function $\mathbb{1}_{\{L > \tau\}}$ as an estimator of the tail loss probability. Before giving details on the variance reduction methods, we describe the naive simulation (NV) method in Algorithm 1 as it is a simpler guide on how Monte Carlo can be applied for quantifying credit risk measures. Note that, this algorithm and all other algorithms given in this paper are designed to estimate both tail loss probability and conditional excess.

Algorithm 1 Naive (NV) simulation of the tail loss probability and the conditional excess

- 1: **for** replications $k = 1, \dots, N$ **do**
 - 2: generate Ψ_d from $f(\cdot; \mathbf{u})$, $d = 1, \dots, D$ independently
 - 3: compute $p_j(\Psi)$, $j = 1, \dots, J$
 - 4: generate $Y_j \sim \text{Bernoulli}(p_j(\Psi))$, $j = 1, \dots, J$
 - 5: compute $L^{(k)} = \sum_{j=1}^J c_j Y_j$
 - 6: **end for**
 - 7: return $\hat{y}^{NV} = N^{-1} \sum_{k=1}^N \mathbb{1}_{\{L^{(k)} > \tau\}}$ for $P(L > \tau)$ and \hat{r}^{NV} using (16) for $E[L|L > \tau]$
-

The random variables Ψ and Y_j , $j = 1, \dots, J$ are the only sources of randomness of the tail loss probability estimates. We call the generation of Y_j , $j = 1, \dots, J$ values for a given Ψ “inner simulation” and the generation of the input vector Ψ “outer simulation”. In order to increase the efficiency of the tail loss probability estimates, we need to focus on these two parts separately.

To reduce the variance coming from the inner simulation, one can use exponential twisting (see, e.g., [Glasserman and Li 2005](#)). However, in a recent paper, [Sak and Hörmann \(2012\)](#) propose a more efficient (see Sect. 5 for the definition of the efficiency measure) method called geometric shortcut (GS) algorithm, which we also use in this paper to reduce the inner variance. Implementing GS alone is not sufficient to decrease the variance for highly dependent obligors. Therefore, we employ importance sampling based on the cross-entropy (CE) (see chapter 8 of [Rubinstein and Kroese 2008](#)) for decreasing the variance coming from the outer simulation. To enhance the efficiency of CE, we combine it with stratification. In the following subsections, we explain the implementation details of these methods under the general Bernoulli mixture models.

3.1 Inner replications using the geometric shortcut

The geometric shortcut idea was introduced in Sect. 3 of [Sak and Hörmann \(2012\)](#) to simulate tail loss probability $P(L > \tau)$ and conditional excess $E[L|L > \tau]$ for independent obligors. The idea is simply to generate instead of many Bernoulli random variables, a geometric random variate that is used as index of the next default in the Gaussian copula framework. To be able to use this idea, the number of repetitions of the simulation of the defaults is increased from 1 to $N_{in} > 1$ (N_{in} denotes the number of inner replications), thus, the generated Ψ vector is not used once but N_{in} times by replacing the generation of Bernoulli random variables by geometric random variables. They argue that the optimum number of N_{in} depends on the contribution of the inner and the outer repetitions on the variance. Although, they give approximate analytical results on N_{in} , their final suggestion is to use $N_{in} = \min(\lfloor 1/\bar{p}(\Psi) \rfloor, J)$ where $\bar{p}(\Psi)$ denotes the average value of the default probabilities, $p_j(\Psi)$, $j = 1, \dots, J$ for the current Ψ vector. Figure 1 provides a visualization of the geometric shortcut method in the flowchart of the full algorithm in Sect. 3.3. For the details of the GS methodology, we refer to [Sak and Hörmann \(2012\)](#). The same idea was applied successfully to the t -copula model in [Sak \(2010\)](#).

3.2 Importance sampling based on cross-entropy method

We consider changing one of the parameters of Ψ_d , $d = 1, \dots, D$ to increase $p_j(\Psi)$ values for the importance sampling (IS). This increases the observed frequency of defaults in the simulation. Most of the papers in quantitative risk management literature consider changing the scale parameter (such as the scale parameter of the gamma distribution) and/or the shift parameter (mean of normal distribution) of the distributions at hand (see, e.g., [Glasserman and Li 2005](#); [Sak et al. 2010](#)). It is our numerical experience that this is a sensible approach as changing the parameters other than these in importance sampling increases the variance of the likelihood ratios in higher dimensions. For a more theoretical approach to choose IS density, we refer the reader to [Geweke \(1989\)](#).

Suppose that $f(\Psi; \mathbf{u})$ is the density of the original vector Ψ . Here, $\mathbf{u} \in \mathbb{R}^D$ denotes the vector of original parameters which are subject to change in IS. Let $\mathbf{v} \in \mathbb{R}^D$ be the vector of new parameters for the IS distribution. The CE method aims to find the optimal IS parameter,

\mathbf{v}^* , by solving the maximization problem (for details see Rubinstein and Kroese 2008, pp. 136–141):

$$\max_{\mathbf{v}} \int \mathbb{1}_{\{L(\Psi) > \tau\}} \log(f(\Psi; \mathbf{v})) f(\Psi; \mathbf{u}) d\Psi. \tag{6}$$

The problem in (6) is a stochastic optimization problem as the indicator function is uncertain for a given Ψ , i.e. it depends also on the Bernoulli random variables Y_1, \dots, Y_J . Therefore, we replace $\mathbb{1}_{\{L(\Psi) > \tau\}}$ with $E[\mathbb{1}_{\{L(\Psi) > \tau\}} | \Psi] = P(L(\Psi) > \tau | \Psi)$.

The solution of the problem in (6) can be estimated by using Monte-Carlo simulation:

$$\max_{\mathbf{v}} \sum_{k=1}^M P(L(\Psi^{(k)}) > \tau | \Psi^{(k)}) \log(f(\Psi^{(k)}; \mathbf{v})), \tag{7}$$

where M is the number of replications in the CE method and $\Psi^{(1)}, \dots, \Psi^{(M)}$ are generated from $f(\cdot; \mathbf{u})$, independently.

If the distribution of Ψ is of exponential family, then a closed-form solution for (7) is possible (see Appendix A.3 of Rubinstein and Kroese 2008). Not to lose the generality of the method, we do not assume any distribution for Ψ here.

Finding the exact value of $P(L(\Psi^{(k)}) > \tau | \Psi^{(k)})$ is possible but generally difficult, since the number of obligors in the portfolio can be large. However we can use one of the approximations listed in Glasserman and Li (2005). Here we use the simplest approach based on the normal approximation. Since,

$$E[L | \Psi] = \sum_{j=1}^J c_j p_j(\Psi)$$

and

$$Var[L | \Psi] = \sum_{j=1}^J c_j^2 [p_j(\Psi) - p_j(\Psi)^2],$$

we obtain the approximation:

$$P(L(\Psi^{(k)}) > \tau | \Psi^{(k)}) \approx 1 - \Phi \left(\frac{\tau - \sum_{j=1}^J c_j p_j(\Psi^{(k)})}{\sqrt{\sum_{j=1}^J c_j^2 [p_j(\Psi^{(k)}) - p_j(\Psi^{(k)})^2]}} \right). \tag{8}$$

Instead of searching for the optimal IS parameters $\mathbf{v} = (v_1, \dots, v_D)'$ independently, we combine identical elements of Ψ , say $(\Psi_1, \dots, \Psi_{D'})'$, in a group such that

$$v_d = \sum_{j=1}^J a_{jd} c_j \theta, \quad d = 1, \dots, D', \tag{9}$$

and optimize for θ given that we know $v_{D'+1}, \dots, v_D$. This is simply related with finding the direction of loss function in $\Psi_d, d = 1, \dots, D'$, which is:

$$\left(\sum_{j=1}^J a_{j1} c_j, \dots, \sum_{j=1}^J a_{jD'} c_j \right)' / \left\| \left(\sum_{j=1}^J a_{j1} c_j, \dots, \sum_{j=1}^J a_{jD'} c_j \right) \right\|.$$

The optimal value for θ determines the IS parameters for $(\Psi_1, \dots, \Psi_{D'})'$ along this direction. This reflects the importance of Ψ on the final probabilities $p_j(\Psi), j = 1, \dots, J$ and decreases the dimension of the optimization problem in (7) to $D - D' + 1$. We can easily decide on how to construct the group by looking at the conditionally independent default probability formula for the model at hand. For example, for the Credit Risk⁺ and Gaussian

copula models, the structure of (2) and (4) suggests that we can combine all of the elements of Ψ and thus choose $D' = D$ (dimension of the problem is one in (7)). However, for (5) it is appropriate to combine $\Psi_1, \dots, \Psi_{D-1}$ thus $D' = D - 1$ (dimension of the problem is two in (7)). Note that, Ψ_D , a chi-squared random variable with η degrees of freedom, is a Gamma random variable with shape and scale parameters equal to $\eta/2$ and 2 respectively.

Using the information that $\Psi_d, d = 1, \dots, D$ are independent in our models, (7) can be written as

$$\max_{\theta, v_{D'+1}, \dots, v_D} \sum_{k=1}^M P(L(\Psi^{(k)}) > \tau | \Psi^{(k)}) \left[\sum_{d=1}^{D'} \log \left(f \left(\Psi_d^{(k)}; \sum_{j=1}^J a_j d c_j \theta \right) \right) + \sum_{d=D'+1}^D \log \left(f \left(\Psi_d^{(k)}; v_d \right) \right) \right], \tag{10}$$

where $P(L(\Psi^{(k)}) > \tau | \Psi^{(k)})$ is given in (8). The full algorithm to compute \mathbf{v} is given in Algorithm 2.

Algorithm 2 Computation of v using CE for a grouping of first D' elements of Ψ

- 1: generate $\Psi_d^{(k)}, d = 1, \dots, D$ independently, $k = 1, \dots, M$ (We use $M = 10,000$)
 - 2: solve for $\theta, v_{D'+1}, \dots, v_D$ using (10)
 - 3: compute $v_d, d = 1, \dots, D'$ using (9)
 - 4: return v_1, \dots, v_D
-

The likelihood ratio under the computed IS parameter \mathbf{v} is calculated using

$$\rho(\Psi^{(k)}; \mathbf{u}, \mathbf{v}) = \prod_{d=1}^D f(\Psi_d^{(k)}; u_d) \left(f(\Psi_d^{(k)}; v_d) \right)^{-1}. \tag{11}$$

The combination of IS based on cross-entropy and the inner replications using the geometric shortcut (CEGS) is presented as Algorithm 3.

Algorithm 3 Tail loss probability and conditional excess estimation using CEGS

- 1: determine D' , then solve for \mathbf{v} using Algorithm 2
 - 2: **for** replications $k = 1, \dots, N$ **do**
 - 3: generate Ψ_d from $f(\cdot; \mathbf{v}), d = 1, \dots, D$ independently
 - 4: calculate $\rho^{(k)}$ as in (11)
 - 5: compute $p_j(\Psi), j = 1, \dots, J$
 - 6: construct a loss vector $L_{(\text{in})}$ of size $N_{\text{in}} = \min(\lfloor 1/\bar{p}_j(\Psi) \rfloor, J)$
 - 7: **for** obligors $j = 1, \dots, J$ **do**
 - 8: initialize λ to zero and *cont* to false
 - 9: **repeat**
 - 10: generate the $U(0, 1)$ random variate U
 - 11: set $\lambda = \lambda + \text{ceiling}(\log(1 - U) / \log(1 - p_j(\cdot)))$
 - 12: **if** $(\lambda > N_{\text{in}})$ **then** set *cont* to true
 - 13: **else** set $L_{(\text{in})}^{(\lambda)} = L_{(\text{in})}^{(\lambda)} + c_j$
 - 14: **until** *cont* = true
 - 15: **end for**
 - 16: compute $\bar{p}_{\text{in}}^{(k)} = N_{\text{in}}^{-1} \sum_{l=1}^{N_{\text{in}}} \mathbb{1}_{\{L_{(\text{in})}^{(l)} > \tau\}}$
 - 17: compute $\bar{L}_{\text{in}}^{(k)} = N_{\text{in}}^{-1} \sum_{l=1}^{N_{\text{in}}} L_{(\text{in})}^{(l)} \mathbb{1}_{\{L_{(\text{in})}^{(l)} > \tau\}}$
 - 18: **end for**
 - 19: return $\hat{y}^{\text{CEGS}} = N^{-1} \sum_{k=1}^N \rho^{(k)} \bar{p}_{\text{in}}^{(k)}$ for $P(L > \tau)$ and \hat{r}^{CEGS} using (17) for $E[L | L > \tau]$
-

3.3 Stratifying the random input

Stratified sampling aims to find the Monte Carlo estimate by conditioning on disjoint and covering subsets (or strata) of the random input domain. Suppose we intend to estimate $E[q(\Psi)]$ with the property that $\Psi \in \mathbb{R}^D$, $q: \mathbb{R}^D \rightarrow \mathbb{R}$, and $E[q(\Psi)^2] < \infty$. We define the random vector $\mathbf{S}(\Psi)$, as a surjective mapping from \mathbb{R}^D onto \mathbb{R}^δ where $\delta \leq D$. Let ξ_1, \dots, ξ_I denote equiprobable strata of \mathbb{R}^δ . Then, the expectation can be calculated using conditional expectations,

$$E[q(\Psi)] = I^{-1} \sum_{i=1}^I E[q(\Psi) | \mathbf{S}(\Psi) \in \xi_i] = I^{-1} \sum_{i=1}^I E[q(\Psi_i)],$$

where I^{-1} is the probability of each equiprobable strata and Ψ_i follows the distribution of Ψ conditional on $\mathbf{S}(\Psi) \in \xi_i$. For $E[q(\Psi)]$,

$$\hat{y} = I^{-1} \sum_{i=1}^I \hat{y}_i = I^{-1} \sum_{i=1}^I N_i^{-1} \sum_{k=1}^{N_i} q(\Psi_i^{(k)})$$

is the stratified estimator, where N_i is the size of the sample drawn from stratum i , $\Psi_i^{(k)}$ is the k th drawing of the random input Ψ conditional on $\mathbf{S}(\Psi) \in \xi_i$, and \hat{y}_i is the sample mean in stratum i . The variance of the stratified estimator is:

$$\text{Var}[\hat{y}] = I^{-2} \sum_{i=1}^I N_i^{-1} (s_i^y)^2, \quad (12)$$

where s_i^y is the sample standard deviation of the responses $q(\Psi_i^{(k)})$, $k = 1, \dots, N_i$.

The variance given in (12) can be minimized by allocating the sample sizes, N_i , proportional to conditional standard deviations s_i^y (see, e.g., Glasserman 2004). As N_i values must be integers, we simply use the formula:

$$N_i = \left\lceil s_i^y N / \sum_{i=1}^I s_i^y \right\rceil, \quad i = 1, \dots, I. \quad (13)$$

Unfortunately, we have no prior information on s_i^y , $i = 1, \dots, I$. A practical solution is to use the estimates of conditional standard deviations obtained through a pilot study with N_p replications. We suggest selecting a sufficiently large N_p to assure the normality of the conditional estimates, \hat{y}_i , and to obtain accurate estimates for the optimal allocation sizes. The remaining $N - N_p$ replications, then, can be used in the main run according to the allocation rule in (13). In the end of the simulation, the sample generated in the pilot study is combined with the sample generated in the main run. By this approach, no drawings are wasted.

The variance of the stratified estimator also depends on the random vector $\mathbf{S}(\Psi)$. By using the conditional variance formula, the total variance of $q(\Psi)$ can be decomposed into two parts:

$$\text{Var}[q(\Psi)] = \text{Var}[E[q(\Psi) | \mathbf{S}(\Psi)]] + E[\text{Var}[q(\Psi) | \mathbf{S}(\Psi)]] \quad (14)$$

The variance of the stratified estimator in (12) is only depending on the conditional variances. Thus, it is only influenced by the second component of (14). Therefore, we try to choose $\mathbf{S}(\Psi)$ such that it maximizes, as far as possible, the first component of (14). In other words, we should choose $\mathbf{S}(\Psi)$ such that the variance between the conditional estimates of each stratum is large. On the other hand, $\mathbf{S}(\Psi)$ should be computationally tractable, so that we can generate the conditional vector Ψ_i easily.

The general idea is to stratify normally distributed elements of Ψ along the IS shift and we suggest stratifying the maximum of the remaining elements. For example, for the CreditRisk⁺ model, we choose $\mathbf{S}(\Psi) = \max \{\Psi_1, \dots, \Psi_D\}$. For the Gaussian copula model, we stratify the Ψ vector along the direction of the IS parameters, \mathbf{v} . So, $\mathbf{S}(\Psi) = \mathbf{v}'\Psi$. Finally, for the t -copula model, we stratify the $\tilde{\Psi}$ vector along the direction of the IS parameters, $\tilde{\mathbf{v}} = (v_1, \dots, v_{D-1})'$, and we also stratify Ψ_D . So, we choose $\mathbf{S}(\Psi) = (\tilde{\mathbf{v}}'\tilde{\Psi}, \Psi_D)'$.

The stratified version of the CEGS method (called STCEGS) is presented as Algorithm 4. We also provide a flowchart diagram to illustrate the steps of the STCEGS method in Fig. 1.

Algorithm 4 Tail loss probability and conditional excess estimation using STCEGS

- 1: determine D' , then solve for \mathbf{v} using Algorithm 2
 - 2: choose an appropriate random valued function $\mathbf{S} : \mathbb{R}^d \rightarrow \mathbb{R}^\delta$ and set strata size vector $(I_1, \dots, I_\delta)'$ and total strata size $I = \prod_{l=1}^\delta I_l$
 - 3: define equiprobable strata ξ_1, \dots, ξ_I on \mathbb{R}^δ
 - 4: define empty lists Λ_i^x and Λ_i^y for $i = 1, \dots, I$
 - 5: set *pilot* = true
 - 6: **if** (*pilot* = true) **then** set $N_i = \lceil I^{-1} N_p \rceil$ for $i = 1, \dots, I$
 - 7: **else** set $N = N - N_p$ and set $N_i, i = 1, \dots, I$ using (13) for $P(L > \tau)$ or using (21) for $E[L|L > \tau]$
 - 8: **for** stratum $i = 1, \dots, I$ **do**
 - 9: **for** replications $k = 1, \dots, N_i$ **do**
 - 10: generate Ψ from $f(\cdot; \mathbf{v})$ such that $\mathbf{S}(\Psi) \in \xi_i$
 - 11: calculate $\rho^{(k)}$ using (11)
 - 12: compute $p_j(\Psi), j = 1, \dots, J$
 - 13: construct a loss vector $L_{(\text{in})}$ of size $N_{\text{in}} = \min(\lfloor 1/\bar{p}_j(\Psi) \rfloor, J)$
 - 14: **for** obligors $j = 1, \dots, J$ **do**
 - 15: initialize λ to zero and *cont* to false
 - 16: **repeat**
 - 17: generate the $U(0, 1)$ random variate U
 - 18: set $\lambda = \lambda + \text{ceiling}(\log(1 - U) / \log(1 - p_j(\Psi)))$
 - 19: **if** ($\lambda > N_{\text{in}}$) **then** set *cont* to true
 - 20: **else** set $L_{(\text{in})}^{(\lambda)} = L_{(\text{in})}^{(\lambda)} + c_j$
 - 21: **until** *cont* = true
 - 22: **end for**
 - 23: compute $\bar{p}_{\text{in}}^{(k)} = N_{\text{in}}^{-1} \sum_{l=1}^{N_{\text{in}}} \mathbb{1}_{\{L_{(\text{in})}^{(l)} > \tau\}}$ and add $\rho^{(k)} \bar{p}_{\text{in}}^{(k)}$ in list Λ_i^y
 - 24: compute $\bar{L}_{\text{in}}^{(k)} = N_{\text{in}}^{-1} \sum_{l=1}^{N_{\text{in}}} L_{(\text{in})}^{(l)} \mathbb{1}_{\{L_{(\text{in})}^{(l)} > \tau\}}$ and add $\rho^{(k)} \bar{L}_{\text{in}}^{(k)}$ in list Λ_i^x
 - 25: **end for**
 - 26: compute \hat{x}_i and s_i^x for the list Λ_i^x , and \hat{y}_i , and s_i^y for the list Λ_i^y , and s_i^{xy} between Λ_i^x and Λ_i^y
 - 27: compute s_i^f using (20) for $E[L|L > \tau]$
 - 28: **end for**
 - 29: **if** (*pilot* = true) **then** set *pilot* = false and go to line 6
 - 30: return $\hat{y}^* = I^{-1} \sum_{i=1}^I \hat{y}_i$ for $P(L > \tau)$ and $\hat{r}^* = \sum_{i=1}^I \hat{x}_i / \sum_{i=1}^I \hat{y}_i$ for $E[L|L > \tau]$
-

4 Conditional excess simulation

We described our new algorithm for tail loss probability computation in Sect. 3. This section explains how a similar methodology can be used for the computation of conditional excess simulation.

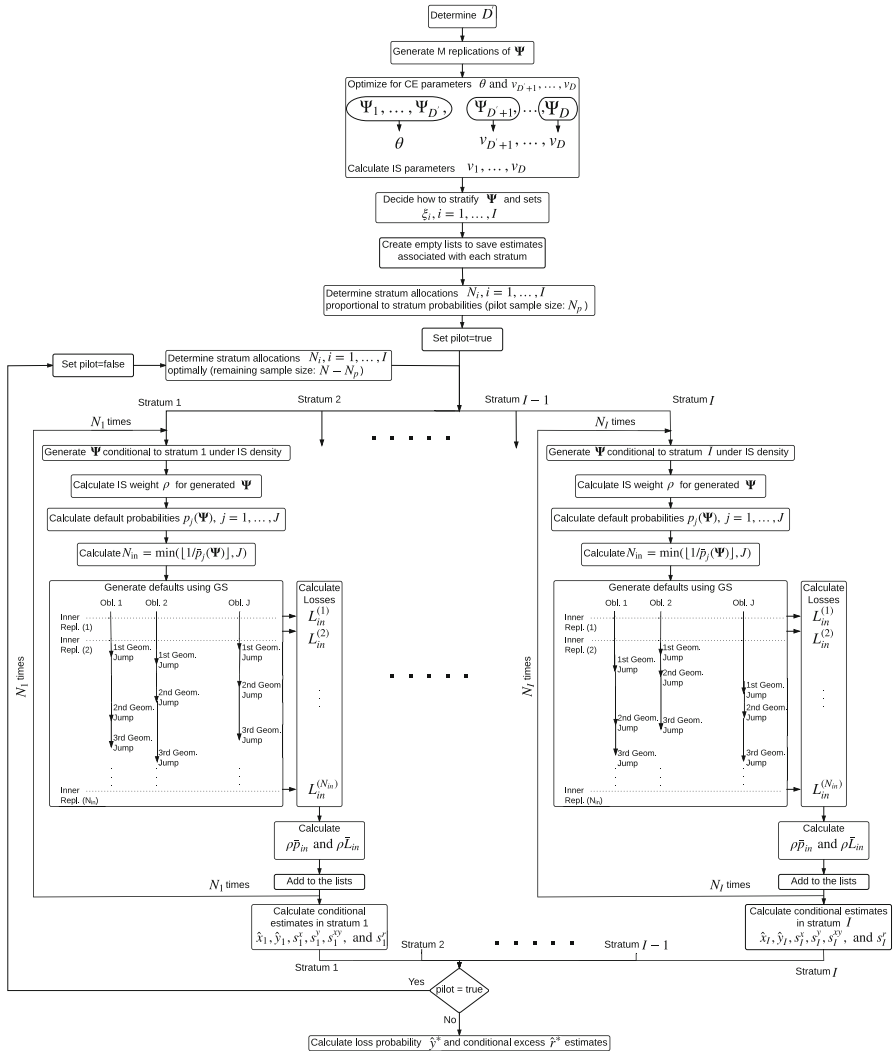


Fig. 1 The flowchart of STCEGS method provided in Algorithm 4

If we assume that $P(L > \tau) > 0$, the conditional excess $r = E[L|L > \tau]$ can be written as $r = x/y$ where $x = E[L \mathbf{1}_{\{L > \tau\}}]$ and $y = P(L > \tau)$. This ratio can be estimated as $\hat{r} = \hat{x}/\hat{y}$ with the approximate variance:

$$Var[\hat{r}] \approx x^2 y^{-4} Var[\hat{y}] - 2xy^{-3} Cov[\hat{x}, \hat{y}] + y^{-2} Var[\hat{x}], \tag{15}$$

which is found by using multivariate Taylor series expansion of the variance of the ratio (see, e.g., Glasserman 2004).

The ratio estimator \hat{r} is biased and the bias has the form

$$Bias[\hat{r}] = xy^{-3} Var[\hat{y}] + y^{-2} Cov[\hat{x}, \hat{y}] + O(N^{-2}),$$

see, e.g., Fishman (1996, p. 109). Here, the leading term is of order $O(N^{-1})$. It is possible to reduce the bias by subtracting the estimate of the leading term from the ratio estimate. However, the squared bias is of order $O(N^{-2})$, thus small compared to the variance. Therefore, it is enough to use the simple ratio estimate without the bias correction.

Let \hat{x}^{NV} and \hat{y}^{NV} denote the naive estimators for x and y . The naive estimator for r is:

$$\hat{r}^{NV} = \hat{x}^{NV} / \hat{y}^{NV} = \left(\sum_{k=1}^N L^{(k)} \mathbb{1}_{\{L^{(k)} > \tau\}} \right) / \left(\sum_{k=1}^N \mathbb{1}_{\{L^{(k)} > \tau\}} \right), \tag{16}$$

and its variance can be estimated using (15). Algorithm 1 gives all the details of how to use this estimate.

Following Glasserman (2005) and Sak and Hörmann (2012), for the simulation of $E[L|L > \tau]$, we use the IS distribution computed for the simulation of $P(L > \tau)$. If we use CEGS, our new estimate of conditional excess is

$$\hat{r}^{CEGS} = \hat{x}^{CEGS} / \hat{y}^{CEGS} = \left(\sum_{k=1}^N \rho^{(k)} \bar{L}_{in}^{(k)} \right) / \left(\sum_{k=1}^N \rho^{(k)} \bar{p}_{in}^{(k)} \right), \tag{17}$$

where

$$\bar{L}_{in}^{(k)} = \left(N_{in}^{(k)} \right)^{-1} \sum_{l=1}^{N_{in}^{(k)}} L^{(k,l)} \mathbb{1}_{\{L^{(k,l)} > \tau\}}$$

denotes the average of the inner replications $L^{(k,l)} \mathbb{1}_{\{L^{(k,l)} > \tau\}}$, $l = 1, \dots, N_{in}^{(k)}$,

$$\bar{p}_{in}^{(k)} = \left(N_{in}^{(k)} \right)^{-1} \sum_{l=1}^{N_{in}^{(k)}} \mathbb{1}_{\{L^{(k,l)} > \tau\}}$$

denotes the average loss probability, and $N_{in}^{(k)}$ is the number of inner repetitions for the k th outer replication.

To estimate the accuracy of (17), we use a general result for ratio estimators given on p. 234 of Glasserman (2004). Since the values $(\bar{L}_{in}^{(k)}, \bar{p}_{in}^{(k)})$ for $k = 1, \dots, N$ are independent and identically distributed, the variance of the ratio estimator under the CEGS method is calculated as:

$$Var[\hat{r}^{CEGS}] \approx \left(\sum_{k=1}^N \rho^{(k)} \bar{p}_{in}^{(k)} \right)^{-2} \sum_{k=1}^N \left(\rho^{(k)} \bar{L}_{in}^{(k)} - \hat{r}^{CEGS} \rho^{(k)} \bar{p}_{in}^{(k)} \right)^2. \tag{18}$$

Equations (17) and (18) can also be found in Sak and Hörmann (2012) and Sak (2010). The full algorithm of simulating tail loss probability and conditional excess using CEGS method is given in Algorithm 3.

When we add stratification to CEGS, the ratio estimate can be calculated as

$$\hat{r}^* = \hat{x}^* / \hat{y}^* = \sum_{i=1}^I \hat{x}_i / \sum_{i=1}^I \hat{y}_i.$$

Here, $\hat{x}_i = N_i^{-1} \sum_{k=1}^{N_i} \rho^{(i,k)} \bar{L}_{in}^{(i,k)}$ and $\hat{y}_i = N_i^{-1} \sum_{k=1}^{N_i} \rho^{(i,k)} \bar{p}_{in}^{(i,k)}$ are estimators conditional to the i th stratum, and $\bar{L}_{in}^{(i,k)}, \bar{p}_{in}^{(i,k)}$ are the same as in (17) but computed conditional to the i th stratum.

The variance of the stratified ratio estimator can be estimated again using (15). However, the variances and the covariance of the estimators must be replaced by

$$Var[\hat{x}^*] = I^{-2} \sum_{i=1}^I N_i^{-1} (s_i^x)^2, \\ Var[\hat{y}^*] = I^{-2} \sum_{i=1}^I N_i^{-1} (s_i^y)^2,$$

and

$$Cov [\hat{x}^*, \hat{y}^*] = I^{-2} \sum_{i=1}^I N_i^{-1} s_i^{xy}.$$

Here, s_i^x denotes the sample standard deviation of $\rho^{(i,k)} \bar{L}_{in}^{(i,k)}$, $k = 1, \dots, N_i$, s_i^y denotes the sample standard deviation of $\rho^{(i,k)} \bar{p}_{in}^{(i,k)}$, $k = 1, \dots, N_i$, and s_i^{xy} denotes the covariance between these two samples. We plug these formulas into (15) and get:

$$Var [\hat{r}^*] = I^{-2} \sum_{i=1}^I N_i^{-1} (s_i^r)^2, \tag{19}$$

where

$$s_i^r = \left(y^{-4} x^2 (s_i^y)^2 - y^{-3} 2x s_i^{xy} + y^{-2} (s_i^x)^2 \right)^{1/2}, \quad i = 1, \dots, I, \tag{20}$$

is the conditional standard deviation of the ratio estimator corresponding to the i th stratum.

Similar to what we describe in Sect. 3.3, the variance given in (19) can be minimized by allocating the sample sizes N_i proportional to conditional standard deviations s_i^r (see [Başoğlu and Hörmann 2014](#)):

$$N_i = \left[s_i^r N / \sum_{i=1}^I s_i^r \right], \quad i = 1, \dots, I. \tag{21}$$

We again use a pilot study to estimate s_i^r , $i = 1, \dots, I$ and determine the allocation sizes for the main run. The final estimate is calculated by combining the samples generated during both the pilot study and the main run. The full algorithm of simulating tail loss probability and conditional excess using STCEGS method is given in Algorithm 4. As stated earlier, we use the same IS strategy to estimate tail loss probability and conditional excess. However, in stratification, the optimal sample allocations are found with different formulas for these two estimates.

5 Numerical results

In this section, we compare the efficiency of the new methods CEGS and STCEGS with other available algorithms in literature. The efficiency of a simulation method is inversely proportional to the product of the variance of its estimator and the execution time (TM) of the simulation. We, therefore, report as a main result of our comparison, the efficiency ratio of the STCEGS estimator

$$ER(\hat{y}^*) = Var [\hat{y}^{BM}] TM(\hat{y}^{BM}) (Var [\hat{y}^*] TM(\hat{y}^*))^{-1},$$

where \hat{y}^{BM} is the benchmark estimator that corresponds to the best method existing in literature.

For the first model, we use the numerical example given in [Glasserman and Li \(2003\)](#). It is a portfolio with $J = 1000$ obligors and an exposure level of $c_j = 0.04 + 0.00196j$ for $j = 1, \dots, J$. Furthermore, $a_{j0} = 0.002$ and $a_{jd} = 0.0002$ for $d = 1, \dots, D = 10$. We assume $\sigma_d = 9$ for $d = 1, \dots, D$.

For the Gaussian copula model, we use the first numerical example given in [Glasserman and Li \(2005\)](#). It is a portfolio of $J = 1000$ obligors in a 10-factor model. The marginal default probabilities $p_j = 0.01(1 + \sin(16\pi j/J))$ for $j = 1, \dots, J$, thus, vary between 0 and 2%; the exposures $c_j = (\lceil 5j/J \rceil)^2$ for $j = 1, \dots, J$, take the values 1, 4, 9, 16, and

25, with 200 obligors at each level. These parameters represent a significant departure from a homogeneous model. The factor loadings a_j are generated independently and uniformly from the interval $(0, 1/\sqrt{10})$; the upper limit of this interval ensures that the sum of squared entries for each obligor does not exceed 1. Note that this upper limit also implies that, for some of the obligors, the sum of the squares of the elements of a_j are close to 1 indicating that this credit portfolio contains strongly correlated obligors.

For the t -copula model, we use the third numerical example of [Sak and Hörmann \(2012\)](#) and [Sak \(2010\)](#). It is a 5-factor model with 1200 obligors. Default probabilities, p_j , are generated independently and uniformly from the interval $[0, 0.02]$ and exposure levels are defined by $c_j = (\lceil 20j/J \rceil)^2$ for $j = 1, \dots, J$. To define the factor loadings, the obligors are separated into six segments of size 200. For each segment, the factors are generated uniformly from the interval $(0, \max)$. For the structure of the matrix and the max values, we refer the reader to [Sak and Hörmann \(2012\)](#) (Table 3, p. 1566).

For each specific model, we tabulate the half length of the 95% confidence intervals as a percentage of the tail loss probability and conditional excess estimators for the available methods in literature (see Table 1). These methods are naive simulation (NV), two-step IS of [Glasserman and Li \(2003\)](#) (TS-IS), the geometric shortcut (GS), the combination of IS and GS given in [Sak and Hörmann \(2012\)](#) (ISGS), and finally, the new methods CEGS and STCEGS introduced in this paper. We indicate the benchmark method for each specific model and give the efficiency ratio of STCEGS method in the last column of Table 1.

The number of equiprobable strata is $I = I_1 = 50$ for CreditRisk⁺ and the Gaussian copula model, and $I = 240$ ($I_1 = 30$, $I_2 = 8$) for the t -copula model. The sample size that we use for the pilot study in STCEGS is $N_p = 30,000$. This leaves approximately 70,000 drawings (N in (13) and (21)) to be allocated optimally in the main run. For CEGS and STCEGS, the number of replications used for determining IS parameters is $M = 10,000$.

Summarizing the numerical results given in Table 1, the efficiency of STCEGS is higher than the efficiency of the methods proposed on those specific credit risk models. The main source of this efficiency improvement comes from the stratification which allows us to allocate more replications to regions with high variances. This aspect of stratification contributes to the objective of the IS method which is to move the sampling process to important regions. Moreover, our experiments showed that stratification yields better results when it is combined with IS.

We report the results of two different IS methods, ISGS and CEGS. The difference between ISGS ([Sak and Hörmann 2012](#); [Sak 2010](#)) and CEGS is the set-up that selects the IS parameters. While ISGS uses the mode of the zero-variance IS function, this paper uses cross-entropy method with a dimension reduction for finding the IS parameters. For both methods, the geometric shortcut is common. Although, the half lengths produced by CEGS are no better than the ones produced by ISGS, CEGS has the advantage of being more general as it can be used for all three specific models. This is the main reason why we combine stratification with CEGS in the STCEGS algorithm.

Throughout the paper, we give algorithms to compute tail loss probabilities. For practical purposes, we may need to calculate risk measures like Value-at-Risk (VaR), which is simply the quantile of the loss distribution. Computing VaR requires simulating tail loss probabilities for a series of threshold values. An efficient algorithm for the estimation of multiple tail loss probabilities in a single stratified simulation is given in [Başoğlu et al. \(2013\)](#).

Table 1 Performance comparisons of the methods under the three models

Model	Interest	τ	Result	Half lengths of estimates as percentage of results						
				NV	TS-IS	GS	ISGS	CEGS	STCEGS	ER
10-factor Gamma	$P(L > \tau)$	8.0	0.1014	1.85	0.99*	0.97		0.40	0.21	13.09
		13.8	0.0101	6.13	1.44*	2.85		0.66	0.41	7.29
	$E[L L > \tau]$	19.7	0.0009	22.62	1.79*	7.75		0.97	0.68	3.74
		8.0	10.52	0.46	0.20*	0.21		0.07	0.04	17.12
10-factor Gaussian	$P(L > \tau)$	13.8	16.27	0.90	0.17*	0.32		0.05	0.04	10.89
		19.7	22.14	2.69	0.15*	0.53		0.06	0.05	5.33
	$P(L > \tau)$	250.0	0.1096	1.77	0.83	1.59	0.59*	0.59	0.12	23.22
		950.0	0.0981	6.27	1.08	5.60	0.83*	0.84	0.17	23.54
	$E[L L > \tau]$	2,000.0	0.00084	22.33	1.12	20.53	1.00*	1.00	0.21	20.70
		250.0	521.74	1.21	0.43	1.13	0.33*	0.33	0.07	17.95
5-factor t-copula	$P(L > \tau)$	950.0	1366.54	2.00	0.28	1.81	0.21*	0.21	0.06	13.15
		2,000.0	2478.75	4.04	0.17	4.33	0.15*	0.15	0.04	10.02
	$P(L > \tau)$	4,500.0	0.0992	1.87		1.70	0.59*	0.56	0.17	11.21
		16,500.0	0.0102	6.10	5.59	0.78*	0.79	0.79	0.25	9.11
	$E[L L > \tau]$	34,000.0	0.0010	19.30	17.57	0.92*	0.90	0.90	0.31	8.25
		4,500.0	9,509.98	1.29	1.19	0.33*	0.33	0.33	0.10	11.00
	$P(L > \tau)$	16,500.0	23,875.57	2.04	1.86	0.21*	0.21	0.21	0.08	7.70
		34,000.0	42,273.97	3.99	3.49	0.14*	0.14	0.14	0.06	5.43

The benchmark methods are indicated with asterisks. Total sample size for each method is approximately 100,000

6 Conclusion

In this paper, we considered the problem of efficient estimation of tail loss probability and conditional excess for the Bernoulli mixture model of credit risk. We presented a new efficient simulation method which is a combination of stratification, importance sampling based on cross-entropy, and inner replications using the geometric shortcut method. It is an important contribution as all the credit risk models proposed in literature can be represented as a Bernoulli mixture model and it is more convenient for statistical fitting purposes compared to threshold models. We also formulated the optimal sample allocation for stratification of ratio estimators. Thus, we obtain further variance reduction for conditional excess estimators. We evaluated the efficiency of our method on three different credit risk models: CreditRisk⁺, the Gaussian, and the *t*-copula model. Numerical results suggest that the proposed general algorithm is more efficient than the benchmark methods for these specific models.

Acknowledgements This work was supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK) Research Fund Project 111M108 and Xi'an Jiaotong- Liverpool University Research Fund Project RDF-14-01-33, and partially supported by Boğaziçi Scientific Research Fund Project 6923.

References

- Baçoğlu, İ., & Hörmann, W. (2014). Efficient stratified sampling implementations in multiresponse simulation. In A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, & J. A. Miller (Eds.), *Proceedings of the 2014 winter simulation conference* (pp. 757–768).
- Baçoğlu, İ., Hörmann, W., & Sak, H. (2013). Optimally stratified importance sampling for portfolio risk with multiple loss thresholds. *Optimization*, 62(11), 1451–1471.
- Bassamboo, A., Juneja, S., & Zeevi, A. (2008). Portfolio credit risk with extremal dependence: Asymptotic analysis and efficient simulation. *Operations Research*, 56(3), 593–606.
- Chan, J. C. C., & Kroese, D. P. (2010). Efficient estimation of large portfolio loss probabilities in *t*-copula models. *European Journal of Operational Research*, 205(2), 361–367.
- Credit Suisse Financial Products. (1997). *CreditRisk+: A CreditRisk management framework*. London: Credit Suisse Financial Products.
- Fishman, G. S. (1996). *Monte Carlo: Concepts, algorithms, and applications*. New York: Springer.
- Frey, R., & McNeil, A. J. (2002). VaR and expected shortfall in portfolios of dependent credit risks: Conceptual and practical insights. *Journal of Banking & Finance*, 26, 1317–1334.
- Geweke, J. (1989). Bayesian inference in econometric models using Monte Carlo integration. *Econometrica*, 57–6, 1317–1339.
- Glasserman, P. (2004). *Monte Carlo methods in financial engineering*. New York: Springer.
- Glasserman, P. (2005). Measuring marginal risk contributions in credit portfolios. *Journal of Computational Finance*, 9(2), 1–41.
- Glasserman, P., & Li, J. (2003). Importance sampling for a mixed Poisson model of portfolio credit risk. In S. Chick, P. J. Sánchez, D. Ferrin, & D. J. Morrice (Eds.), *Proceedings of the 2003 winter simulation conference* (pp. 267–275).
- Glasserman, P., & Li, J. (2005). Importance sampling for portfolio credit risk. *Management Science*, 51(11), 1643–1656.
- Gupton, G. M., Finger, C. C., & Bhatia, M. (1997). *Creditmetrics technical document*. New York: J.P. Morgan & Co.
- Kang, W., & Shahabuddin, P. (2005). Fast simulation for multifactor portfolio credit risk in the *t*-copula model. In M. E. Kuhl, N. M. Steiger, F. B. Armstrong, & J. A. Joines (Eds.), *Proceedings of the 37th conference on winter simulation* (pp. 1859–1868).
- McNeil, A. J., Frey, R., & Embrechts, P. (2005). *Quantitative risk management*. New Jersey: Princeton University Press.
- Rubinstein, R. Y., & Kroese, D. P. (2008). *Simulation and the Monte Carlo method*. New Jersey: Wiley.
- Sak, H. (2010). Increasing the number of inner replications of multifactor portfolio credit risk simulation in the *t*-copula model. *Monte Carlo Methods and Applications*, 16, 361–377.

- Sak, H., & Hörmann, W. (2012). Fast simulations in credit risk. *Quantitative Finance*, *12*(10), 1557–1569.
- Sak, H., Hörmann, W., & Leydold, J. (2010). Efficient risk simulations for linear asset portfolios in the t-copula model. *European Journal of Operational Research*, *202*, 802–809.