Annals of Operations Research (2024) 334:157–184
https://doi.org/10.1007/s10479-022-04845-7

ORIGINAL RESEARCH

A semi-supervised learning approach for variance reduction in life insurance

Martin Jimenez[1] · **Yahia Salhi**[2]

Accepted: 16 June 2022 / Published online: 30 June 2022

Abstract
Monte-Carlo based valuation in life insurance involves the simulation of various components of the balance sheet: portfolios, guarantees, assets mix, market conditions and many other specific risk factors; which can be time-consuming. In view of the time needed to achieve this task, insurers are then facing a trade-off of balancing the number of simulations against the uncertainty surrounding the estimated quantity. In the current paper, we propose a variance-reduction methodology using a machine learning technique. It roots from the unsupervised learning literature in conjunction with the quantization of random processes. The goal is to reduce the number of simulated Brownian paths, using auxiliary scenarios that can be seen as path clusters, which efficiently and accurately approximate the initial ones with regards to an adequate measure of distance. Moreover, we introduce penalty to accommodate for various inputs of the initial conditions of risk factors. By doing so, we implicitly assign labels to the scenarios and thus advocate using a semi-supervised learning to enhance the performance of the scenarios reduction by imposing an additional impurity condition on the clusters based on these labels. This is made possible thanks to a decomposition property of the insurers cash-flows, which allows to disentangle the initial conditions of risk factors from the Brownian motions driving their dynamics. The training of the proposed learning algorithm is based on an adaptation of the well-known k-means algorithm. An intensive numerical study is carried out over a range of simulation setups to compare the performances of the proposed methodology. We show, numerically, that the proposed methodology outperforms some classical variance reduction approach. Also, using a real-life dataset, we show that our methodology outperforms some conventional variance reduction used by life insurance practitioners.

Keywords Scenarios reduction · Semi-supervised learning · Insurance liabilities

Mathematics Subject Classification 81P16 · 90B90 · 62P05

✉ Yahia Salhi
yahia.salhi@univ-lyon1.fr

1 FRACTALES, 64 Rue Tiquetonne, 75002 Paris, France

2 UCBL, ISFA LSAF EA2429, Univ Lyon, 50 Avenue Tony Garnier, 69007 Lyon, France

Introduction

Recent developments in both prudential (Solvency II) and financial reporting (MCEV, IFRS) regulations highlighted the need for consistent market valuation as a universal framework to the assessment of insurance liabilities, see (Thérond, 2016). The aim of such a framework is twofold. First, the market-consistent valuation is used as the basis for assessing the risks faced by the insurers making it compatible with the market practice. This allows, for instance, to ease the comparative assessment of investment decisions among different market assets and other types of investments on the markets. Therefore, market-consistent valuation allows risks to "be valued at the amount for which they could be exchanged between knowledgeable willing parties in an arms length transaction". Secondly, there is an objective to increase the level of harmonization among the market stakeholders. The advent of such a practice has fundamentally changed the way the solvency of insurance companies is assessed. For instance, in terms of quantitative capital requirements, the purpose of introducing the Solvency Capital Requirement (SCR) is to be more consistent with the risks actually supported by an insurer. It is considered as a key element in the risk management process, and corresponds to the economic capital required for a company to be able to meet its commitment up to a certain level of tolerance. Therefore, insurers run into the problem of being able to assess the impact of a decision (the launch of a new offer, a change of the strategic allocation of investments, the introduction or a modification of a risk transfer provision, the determination of contracts revaluation) that may affect their risk profile and thus the solvency ratio. This is a difficult and thorny topic, particularly, in life insurance where economic capital, like the solvency capital requirement (SCR), is extremely sensitive to changes in the economic environment, e.g. yield curve, tighter credit spreads, movements of equity markets and other relevant factors, see (Bignon et al., 2019). If this exercise is mandatory for supervisory reporting, on an annual basis, insurance companies generally impose constraints on their surplus or solvency ratio, in terms of effective management of their risks. It is, thus, necessary to frequently monitor the indicators prescribed by the regulators and needed for risk management purposes. From a practical point of view, the operational implementation of a market consistent valuation consists of a complex process that requires precise information and significant calculation times in order to obtain their full loss distribution based on an assessment of their future uncertain profits. This uncertainty arises from the development of a number of influencing factors, such as equity returns, interest rates, credit spreads or biometric risks. Due to the wide range of guarantees and option-like features, closed-form solutions for the considered cash flows are only available in special cases. As a consequence, in practice, the balance-sheet is usually simulated based on Monte-Carlo schemes, see (Floryszczak et al., 2016) and (Borel-Mathurin & Vedani, 2019). To achieve this, massive simulations of the insurer's balance sheet are required, which takes into account the stochastic dynamics of the underlying risk factors and the more complex interactions between the assets and liabilities.

In practice, insurers use the so-called table of scenarios, which is composed of simulated paths of Brownian motions. It allows to disentangle the economic risk factors from the uncertainty driving them, see (Vedani et al., 2017). The insurer can coherently change or apply shocks on these factors in order to assess, for instance, its capital adequacy and resilience to adverse scenarios. Even though insurance companies use distributed calculation for simulations to produce results faster, the required runtime to achieve such calculations is usually estimated in hours. This makes practitioners prefer to reduce drastically the amount of scenarios used in order to decrease the computation time. However, this give arise to a sampling bias. So, it is important, from a risk management point of view, to control such an uncertainty. A natural strategy consists in finding an alternative Monte-Carlo estimator providing smaller variance. This is referred to as variance reduction, see (Glynn, 1994) for a discussion. Two classical methods of variance reduction are suited to such a problem: control variates and antithetic variables methodologies. In insurance, a general description and practical aspects of these methods can be found in Borel-Mathurin and Vedani (2019). Still, these methods may strongly depend on the cash flows projection model or on the underlying risk factors and may require significant changes in the practical implementation of the Monte Carlo simulation. Therefore, most practitioners do not use the most advanced methods except for marginal cases. On the other hand, some reduced-form methods are proposed in the literature, which aim at representing the insurer balance sheet using a simple regressive model that capture its main component. However, as discussed by Bignon et al. (2019) such models are very sensitive the initial economic situation, such as the asset mix, the interest rate curve, etc. To cope with these drawbacks, in the current paper, we propose a methodology that roots from the unsupervised learning literature in conjunction with the quantization of random processes. The goal is to reduce the number of simulated Brownian paths, using auxiliary scenarios that can be seen as clusters. Those efficiently and accurately approximate the initial ones with regards to an adequate measure of distance. The proposed methodology in this paper has an appealing interpretation in the machine learning literature related notably to the vector quantization, see (Gray, 1984). It is an unsupervised scheme that allows to aggregate the economic scenarios according to their proximity. However, we should note that the complex behavior of the insurer's balance sheet represents an additional challenge as soon as the liabilities would react adversely with regard to the generated scenario. For instance, if the equity risk is considered, its impact follows intimately the extreme scenarios of the Brownian paths in the sense that consecutive draw-downs give rise to losses on the asset side. On the other hand, when the inflation or property risks are inspected, cash flows do not seem to give a clear pattern in terms of the path of the underlying Brownian motion. To cope with this stylized fact, in this paper, we will add a guiding component to the unsupervised learning process. For instance, in Sect. 1, we consider a convenient decomposition which allows to disentangle the Brownian path from the initial state of the risk factors. By doing so, we implicitly assign labels to the scenarios and thus advocate using a semi-supervised learning to enhance the performance of the scenarios reduction by imposing an additional impurity condition on the clusters based on these labels. Hence, we introduce a methodology to accommodate for various inputs of the initial conditions of risk factors. This will inherently capture the behavior of the insurer's balance-sheet with regard to changing market and portfolios conditions. Accordingly, we address most of the drawbacks discussed in Borel-Mathurin and Vedani (2019) and Bignon et al. (2019), which makes the Monte-Carlo schemes less dependent on the number of simulated paths. We assess its performance using some classic illustrative examples. For instance, we consider two financial cashflows related, respectively, to a call and a lookback options. These allow for a comparative study to assess the effectiveness of the variance reduction. Moreover, we discuss a real-world example based on an insurer simulated balance-sheet.

The remainder is organized as follows. In Sect. 1, we introduce the mathematical framework used to model the insurer's economic balance sheet, mostly described through the observed risk factors. We also discuss the practical implementation of the scenario-based approach used to estimate the expected future cash flows of profits. In Sect. 2, we introduce an adequate decomposition of these cash flows allowing to disentangle the so-called scenarios' table from the initial values of the risk factors. This will be of paramount interest in introducing, first, a supervised approach that aims at reducing the number of the required scenarios while keeping them closer to the initial ones. Finally, we propose a semi-supervised approach that will take into account the additional information stemming from the different initial situations and allows to label the scenarios. The scenarios reduction is thus formulated as a constrained optimization problem, which can be tackled in a similar fashion as for the well celebrated k-means algorithm. Finally, in Sect. 3, we illustrate the proposed methodology based on simulation study and a real-world data set.

1 Mathematical framework and problem formulation

In this section, we introduce the mathematical framework as well as the notation related to the modeling of life insurers balance sheets. We will use an abstract representation to describe the dynamics of risk factors driving the assets and liabilities. Based on a decomposition property described below, we will be able to disentangle the so-called economic scenario from the initial state of the risk factors, which will be crucial for the scenarios aggregation discussed later on this paper.

1.1 Market-consistent valuation

We consider a filtered probability space $(\Omega, \mathbb{F}, \mathbb{P})$ satisfying the usual assumptions, where Ω represents the space of all possible states in the financial and insurance market, $\mathbb{P}$ is the reference probability measure, also called the *real world* or *historical* probability measure, and $\mathbb{F} = (\mathcal{F}_t)_{t\geq 0}$ represents all available information about the risk factors that affect the profitability of a portfolio or the solvency of the insurer company over time. This σ-algebra gathers information related to the financial assets, insurance liabilities, policyholder and insurer behaviors, among other risk factors. For valuation purposes, we take for granted the existence of a risk-neutral probability measure $\mathbb{Q}$ equivalent to $\mathbb{P}$ under which payment streams can be valuated as expected discounted cash flows. Moreover, we assume that the insurer balance sheet is driven by $d \in \mathbb{N}$ risk factors and let $X = (X_t)_{t\geq 0}$ be the d-dimensional process describing their dynamics. In the sequel, we assume that the dynamics of X obey to the following stochastic differential equation:

$$dX_t = \mu(X_t)dt + \sigma(X_t)dW_t, \tag{1}$$

where $W = (W^j)_{j=1,\ldots,d}$,, with $W^j = (W_t^j)_{t\geq 0}$ for $j = 1, \ldots, d$, is an $\mathbb{R}^d$-valued vector of (correlated) standard Brownian motions. Here, $\mu : \mathbb{R}^d \mapsto \mathbb{R}^d$ denotes the drift process and $\sigma : \mathbb{R}^d \mapsto \mathbb{R}^{d\times d}$ the volatility process. Here, for the sake of simplifying the notation, we assume throughout this paper that the coefficients of the diffusion do not depend on time t. The general case can be reduced to that one by considering stochastic differential equation satisfied by the process $Y_t = (t, X_t)$. Moreover, in practical applications most of the involved models are broadly covered by the above considered time-homogeneous process. Hence, in order to valuate insurance liabilities, we let $\Upsilon : \mathcal{C}([0, T], \mathbb{R}^d) \to \mathbb{R}$, for $T > 0$, be a measurable function representing the sum of discounted future profits due to the in-force business. This mapping relates the risk factors X to the insurer balance sheet. In most cases, this mapping is based on the insurer's own simulation-based model that takes into account the underlying liabilities as well as the interaction between those and the asset side. The use of such a modeling approach has become mandatory for several years for insurers in order to meet regulatory, financial and accounting standards which require a fair and market consistent value assessment of their portfolios. Based on the stochastic risk factors and their dynamics in (1), we consider a projection of the cash flows over a horizon T and introduce the market-consistent value of the cash flows at time $t < T$ as the $\mathcal{F}_t$-measurable r.v. $\mathbb{E}\left[\Upsilon(X)\middle|\mathcal{F}_t\right]$. In the sequel, we will assume that X is a Markov process so that the market-consistent value of future profits, denoted V, can be written as

$$V(x) = \mathbb{E}\left[\Upsilon(X)\middle|X_t = x\right]. \tag{2}$$

In other words, the market-consistent value V reduces to a conditional expectation with respect to a σ-field generated by the random vector X_t, for $t \geq 0$. Notice that in the above definition we use $\Upsilon(X)$ instead of $\Upsilon((X_s)_{s\in[t,T]})$ to alleviate the notation. In fact, as we can see in the sequel, the cash flows projection model will inevitably be given in terms of a collection of points drawn from the continuous process X. In that case, we will explicit the dependency of the mapping in terms of these discretized processes.

Remark 1 (Markov Property) In line with practical use of the market-consistent valuation, the process X is assumed be Markov in the sense that only the current state of the market and liability positions impact the future profits. Hence, the market-consistent value V does not depend on the entire history of the risk factors, see (Bauer et al., 2010) for a discussion. However, in some applications, in particular in life insurance, we should keep track of relevant information related to the contracts: account value, cash-surrender value, current death benefit, etc. In order to accommodate for such a case we can introduce state variables gathering this past information as suggested, for example, in Bauer et al. (2010, b). In that case V can incorporate an additional dimension. Alternatively, one can consider a more general dynamics such (1) including the evolution of the state factors. Also, it is possible to extend the models so that they include more risk sources than solely asset and liability risk, see (Zenios, 1995).

The $\mathcal{F}_t$-measurable r.v. $V(X_t)$ is of paramount importance in assessing the insurer's profitability as well as most of the regulatory quantities, e.g. the embedded value (MCEV), best estimate of liabilities (BEL) and the solvency capital requirement (SCR) among others. It is intrinsically linked to the discounted cash flows process $\Upsilon(X)$, where the map Υ refers to the so-called cash flow projection model that derives the future profits from the development of the risk factors X up to time t. This takes into account the interrelations between the assets and liabilities due, for example, to the discretionary profit sharing mechanism and accountancy rules, among other things. The use of such a projection model is very common in life insurance and allows to avoid an isolated analysis of assets and liabilities and reflects legal and regulatory requirements as well as management rules, see Bignon et al. (2019). In fact, such a model also considers the matching and diversification effects of the two sides of the insurer's balance sheet, see (Bignon et al., 2019) for a discussion. Although no simple expression holds for Υ, in general, various examples can be considered related to specific contracts with simplified accountancy rules as well as asset and liabilities interactions: the valuation of SCR for variable annuities or participating life contracts as in Hilli et al. (2007), Cathcart and Morrison (2009) and Hainaut et al. (2018), among others. Hence, in order to make our framework readily applicable to more complex contracts and insurer's balance sheets, we keep an abstract presentation of the map Υ. Therefore, the following aims at focusing on maps that are not accessible and to a certain degree can be regarded as the output of the cash flows projection model. The latter is, in most cases, a *black box*-like procedure which implements the the cash flows projection model and gives simulations of the discounted cash flows, see (Floryszczak et al., 2016) and (Felix, 2016).

1.2 Asset and liability model

In order to valuate the future expected discounted profits $V(x)$, we will suppose that the drift and the volatility of the dynamics of risk factors in Eq. (1) are uniformly Lipschitz continuous functions and σ is of class $\mathcal{C}^1$. Under these assumptions, the stochastic differential equation (1) has a unique solution, see (Durrett & Durrett, 1984). Furthermore, the Yamada-Watanabe theorem, see (Durrett & Durrett, 1984), ensures the existence of a measurable mapping Ψ on $\mathcal{C}([0, T], \mathbb{R}^d)$ such that $X_s = \Psi(W, X_t)$ for $s \geq t$, where W stands for the path of the process between valuation date t and the projection horizon T. Although no simple expression holds for this map Ψ in general, various schemes (Euler scheme, Milstein scheme, etc.) allow one to construct such an approximation. In general, the latter is not a function of the whole path of W between t and T but only of a part of it in a similar way to the Monte-Carlo scheme discussed in the previous section. For instance, if we use an Euler scheme with a discretization grid $t_k = t + k(T-t)/n$, for $k = 0, \ldots, n$, the (approximated) mapping Ψ only depends on the vector $Z = (W_{t_0}, W_{t_1}, \ldots, W_{t_n})$. Hence, we can write the market consistent value in Eq. (2) on the discretized grid as follows

$$V(x) = \mathbb{E}\left[f(Z, X_t)\middle|X_t = x\right], \tag{3}$$

where $f = \Upsilon \circ \Psi : \mathbb{R} \times \mathbb{R}^d \mapsto \mathbb{R}$ represents the projection model, which is decomposed into two distinct sub-modules. This decomposition is very appealing as it allows to distinguish the asset from the liability component in the insurer balance sheet related, respectively, to the mapping Υ and Ψ. Moreover, this is in line with practical applications as the economic scenarios are generally summarized by the set of random standard variables Z and then an initial state for the vector X_t is injected in the cash flows projection model, see (Felix, 2016). Moreover, as discussed in Borel-Mathurin and Vedani (2019), the common practice in life insurance is to disentangle the so-called *scenarios' table*, i.e. Z, from the initial levels of the risk factors X_t. The main aim of this decomposition is to allow for assessing the impact of economic scenarios on X_t on the asset-liability management (ALM) model used by practitioners when achieving an economic valuation. Indeed, when $t = 0$, for example, insurers are called to assess the present value of their future profits using not only a baseline condition but also marginally shocked economic conditions associated to the risk factors X_0. For instance, these are aimed to assess the insurers resilience to some specific endogenous stress tests. In our case, this decomposition will help label the scenarios Z taking into account their impacts on the cash flows. More formally, each scenario Z_i involved in the calculation of the Monte-Carlo estimate of $V(x)$ will generate a given cash flow conditionally on the initial state x. In the following, for the sake of simplicity, we will mainly focus on an initial valuation at time $t = 0$ in presence of various initial scenarios $X_0 = x_1, \ldots, x_d$ as discussed above.

Remark 2 In the case of a Black-Scholes model (geometric Brownian motion), where the risk factors evolve following the dynamics

$$dX_t = rX_t dt + \sigma X_t dW_t \tag{4}$$

where r and σ in Eq. (1) are some constants, the map Ψ can fully characterized using the initial condition $X_0 = x_i$ and the vector of Gaussian increments Z. In fact, using the Euler scheme with a discretization grid t_k, with time step $\Delta t = T/n$, Ψ can be explicitly written as follows

$$\Psi(Z, x_i) = x_i \prod_{k=1}^{n} \left(1 + r\Delta t + \sigma(W_{t_k} - W_{t_{k-1}})\right).$$

In our case, the knowledge of Ψ is not required and we will keep the abstract formulation of the problem to cope with the real-world applications. In fact, even if the latter can be fully described, depending the model under consideration as well as the discretization scheme, the resulting function of interest $f = \Upsilon \circ \Psi$ is still inaccessible.

1.3 Monte-carlo simulation and economic scenarios generation

In practice, the valuation formula in Eq. (2) is estimated using a Monte-Carlo scheme based on the simulation of the random variable Z. Using a scenario table, one diffuses the impact of the initial levels of the risk factors X_t over the projection horizon thanks to the ALM model. Even if this procedure is quite straightforward for some classic financial toy examples, it is not the case when dealing with insurance liabilities. Since life insurance contracts present some complex mechanisms, such as profit sharing, target crediting rate and dynamic lapses behaviors of policy holders, computing the present value of future profits, among others, involves a projection model that can become cumbersome very often. Accordingly, the Monte-Carlo valuation can be very time-consuming.

2 Machine learning approach to variance reduction

The goal of the following is to propose an estimation procedure in the same spirit as the Monte-Carlo while attempting to reduce the variance and thus the computation time. Formally, we aim at reducing the number of simulated scenarios used for calculation white keeping the variance of the estimation at a satisfactory level. The scenarios aggregation methodology is constructed in a such a way to adapt to different risk factors initial levels.

2.1 Monte-carlo estimation and uncertainty

Recall that the estimation of $V(x)$ using the Monte-Carlo relies on the set of economic scenarios, which are represented by a set of random vectors z_i, for $i = 1, \ldots, N$, drawn from the law F of Z. Hence, the Monte-Carlo approximation of $V(x)$, denoted $V_{\mathrm{MC}}^N(x)$ can be rewritten as follows

$$V_{\mathrm{MC}}^N(x) = \sum_{i=1}^{N} p_i f(z_i, x), \tag{5}$$

such that $p_i = \mathbb{P}(Z = z_i)$. When the samples z_i are equiprobable, which is the case in practical applications, this probability of the mass points of F can be simply written as $p_i = 1/N$. In such a case, the variance of the estimate is given by $\sigma_{\mathrm{MC}} = N^{-1}\sqrt{\mathbb{V}\mathrm{ar}(f(Z, x))}$. Although the convergence of the estimate $V_{\mathrm{MC}}^N(x) \to V(x)$ in probability as $N \to +\infty$ is granted by the Law of Large Numbers, it is desirable to reduce the variance of this estimate instead of focusing on scenarios reduction as discussed later on this paper. In fact, the classic

Central Limit Theorem tells thats

$$\mathbb{P}\left(V(x) \in \left[V_{\mathrm{MC}}^N(x) - z_{\alpha/2}\frac{\sigma_{\mathrm{MC}}}{\sqrt{N}}, V_{\mathrm{MC}}^N(x) + z_{\alpha/2}\frac{\sigma_{\mathrm{MC}}}{\sqrt{N}}\right]\right) \to 1-\alpha \text{ as } N \to +\infty, \quad (6)$$

where $z_{\alpha/2}$ is the $(1-\alpha/2)$ percentile point of the normal distribution, i.e. $1-\Phi(z_{\alpha/2}) = \alpha/2$ with Φ being the standard normal cumulative distribution function. Note that σ_{MC} may differ significantly for different scenarios x, i.e. under different realizations of the risk factors. No matter when the valuation is performed the uncertainty surrounding the estimated value based on basic Monte-Carlo techniques can be large. Thus, in order to decrease the width of the confidence intervals one can increase N, but this also increases the computational cost. Even though insurance companies use distributed calculation on simulation and on sensitivities to produce results faster, the total runtime is usually estimated in hours. Hence, a better strategy is to reduce variance σ_{MC}^2 by finding an alternative Monte-Carlo estimator involving random variables with smaller variance. This is referred to as variance reduction. Two classical methods of variance reduction are adapted to this problem: control variates and antithetic variables methodologies. A general description of the methods can be found in Borel-Mathurin and Vedani (2019). However, these methods may strongly depend on the theoretical behavior of the cash flows projection model, i.e. the functional Υ, or the underlying model (1) for the risk factors X and require significant changes in the practical implementation of the Monte Carlo simulation. For instance, in our case we do not require any further knowledge aside the simulated paths of the Brownian motions. This makes the methodology developed henceforth different in the spirit from most of the variance reduction techniques which, as already mentioned, require in most cases the full accessibility of the mapping f. Therefore, most practitioners do not use the most sophisticated methods except for marginal cases.

2.2 Unsupervised learning and scenarios reduction

Recall that the realization of random vector Z are defined by the distribution function $F : \mathbb{R}^n \to [0, 1]$. Here, we want to find a random vector $\widehat{Z}$ which takes its values in a finite set of points $\{\widehat{z}_1, \ldots, \widehat{z}_K\}$, for $K < N$, in such a way that the value function V can be approximated by

$$V_{\mathrm{ML}}^K(x) = \sum_{j=1}^{K} \widehat{p}_j f(\widehat{z}_j, x), \quad (7)$$

where $\widehat{p}_j = \mathbb{P}(\widehat{Z} = \widehat{z}_j)$. Hence, we are willing to find a finite number of trajectories which "summarize" the behavior of the Brownian motions and thus the random vector Z. By doing so, the resulting estimation of the value function $V(x)$ requires less scenarios than the initial Monte-Carlo approximation. To do so, Bally and Pages (2003) propose an approach to construct scenario lattices for Brownian motions based on a minimization of the distance between the initial scenarios and its representation $\widehat{Z}$. In other words, the optimal representation of the $\mathbb{R}^d$ random vectors Z consists in finding the best possible approximation by a measurable function that takes at most K values in $\mathbb{R}^d$.

In a more general setting, we wanted to choose the r.v. $\widehat{Z}$ with distribution function $\widehat{F} : \mathbb{R}^K \to [0, 1]$ in such a way that the computation of the above quantity requires a reduced number of samples K compared to the initial sample of size N. Also, the distribution $\widehat{F}$ should be close to the initial one. In order to optimally choose this vector we aim at minimizing the distance to the initial set of scenarios Z. To do so, various metrics can be chosen, see (Liu

& Pagès, 2020). Here, we will consider the Wasserstein distance to measure the proximity between the two samples, in the sense that for each component Z_i and $\widehat{Z}_j$ of the two vectors, we define

$$d_p(F, \widehat{F}) = \left(\inf_{g \in M(F,\widehat{F})} \int \|z - \widehat{z}\|^p dg(z, \widehat{z}) \right)^{1/p}$$

where $\|\cdot\|$ is the $\mathbb{L}_p$-norm on $\mathbb{R}^n$ and $M(F, \widehat{F})$ refers to the set of all finite measures having marginal distributions F in the first component and $\widehat{F}$ in the second component. Letting $\delta_{i,j} = \mathbb{P}(Z_i = z_i, \widehat{Z}_j = \widehat{z}_j)$, then the distance $d_p(F, \widehat{F})$ can be written in a discretized form as follows

$$d_p(F, \widehat{F}) = \left(\min_{\delta_{i,j} \in [0,1]} \left\{ \sum_{i=1}^{N} \sum_{i=1}^{K} \delta_{i,j} \|z_i - \widehat{z}_j\|^p \right\} \right)^{1/p}, \tag{8}$$

such that $\sum_{j=1}^{K} \delta_{i,j} = p_i$ and $\sum_{i=1}^{N} \delta_{i,j} = \widehat{p}_j$. We can see that the Wasserstein distance can be viewed as a minimum-cost transportation problem, where $\delta_{i,j}$ represents the amount of probability mass shipped from z_i to $\widehat{z}_j$ at unit transportation cost $\|z_i - \widehat{z}_j\|^p$. Thus, $d_p(F, \widehat{F})$ quantifies the minimum cost of moving the initial distribution F to the target distribution $\widehat{F}$. In the literature, this approach is referred to as the mass transport problem, which amounts to construct approximate mass probabilities, which redistribute the mass probabilities of the initial generated scenarios Z while minimizing a probability distance between them and the representative scenarios, see (Villani, 2008). In other words, the aim is to look for a set of mass points $(\widehat{z}_j, \widehat{p}_j)$, for $j = 1, \ldots, K$, such that the distance in (8) is minimized. As noted before, this boils down to finding the best summary of the initial economic scenarios by a smaller number of *synthetic* scenarios.

In the following, we consider the Euclidean distance, i.e. $p = 2$ in (8), and we will assume a set of equiprobable initial sample scenarios, i.e. $p_i = 1/N$, so that the optimization problem amends to finding the set of mass points $(\widehat{z}_j, \widehat{p}_j)$ that minimize the distance $d(F, \widehat{F})$ under the simplified constraints $\sum_{i=1}^{N} \delta_{i,j} = N\widehat{p}_j$ and $\sum_{j=1}^{K} \delta_{i,j} = 1$, i.e.

$$\min_{\widehat{z}_1,\ldots,\widehat{z}_K} \left\{ \left(\min_{\delta_{i,j} \in \{0,1\},\, \widehat{p}_j \geq 0} \left\{ \sum_{i=1}^{N} \sum_{j=1}^{K} \delta_{i,j} \|z_i - \widehat{z}_j\|^2 \text{ s.t. } \sum_{j=1}^{N} \delta_{i,j} = N\widehat{p}_j, \sum_{j=1}^{K} \delta_{i,j} = 1 \right\}^{1/2} \right) \right\}. \tag{9}$$

This optimization problem belongs to the so-called NP-hard problems, see (Aloise et al., 2009) and thus requires an adequate algorithm that will look for a local minimum instead of a global one as the problem (9) is nonconvex. In fact, exhaustive search for the optimal clusters is intractable even for $K = 2$. However, when $p = 2$, the above strategy can be formulated as the Lloyd's method I, which is better known as the k-means clustering algorithm, MacQueen (1967). In fact, we are looking for clusters of scenarios that best matches with regards to their Euclidean distance. To see this, one can first start from an initial guess for scenarios $\widehat{z}_j$ then looking for the local optimum over $\delta_{i,j}$ and $\widehat{p}_j$. Also, in order to cope with the classic k-means procedure, we will set $\delta_{i,j} = 1$ in the minimization (9) if $j = \text{argmin}_k\{\|z_i - \widehat{z}_k\|\}$. By doing so, the decision variable $\delta_{i,j}$ becomes a binary variable and be regarded as a cluster assignment variable. In that case, the problem perform a hard assignement of data to each cluster. In other words, each vector z_j of the original scenarios can be only assigned to one cluster. Hence, the representative scenario $\widehat{z}_j$ simplifies to the mean of all z_i where $\delta_{i,j} = 1$. Formally, if we denote $C(Z)$ the j-th partition of the initial scenarios space, which contains all scenarios closest to the representative one $\widehat{z}_j$, such that

$$C_j(\widehat{z}_j) = \left\{ z_i : j = \underset{k=1,\ldots,K}{\operatorname{argmin}} \left\{ \|z_i - \widehat{z}_k\|^2 \right\}, i = 1, \ldots, N \right\},$$

for $j = 1, \ldots, K$; then the minimization problem can be formulated as follows

$$\min_{\widehat{z}_1,\ldots,\widehat{z}_K} \left\{ \sum_{j=1}^{K} \sum_{z_i \in C_j(\widehat{z}_j)} \|z_i - \widehat{z}_j\|^2 \right\}. \tag{10}$$

Remark 3 The problem of finding the closed approximation in (8) can be linked to literature on optimal quantization problem, which arises in electrical engineering in connection with signal processing and data compression. More recently, it became an efficient tool in numerical probability to compute regular and conditional expectations. The quantization approach for solving this problem considers the minimization of a quadratic distortion, see (Liu & Pagès, 2020) and (Bally & Pages, 2003) for discussion. In our case, since we use the Euclidean distance and we impose that a hard assignment rule, i.e. $\delta_{i,j} = 1$ for neighboring scenarios, we fall naturally into the quantization approach. Indeed, the k-means clustering approach has a close connection with quadratic optimal quantization, see (Pollard, 1982).

Solving the optimization problem in Eq. (10) provides the representative scenarios which result in a good way to define influential patterns. Hence, in order to set up the approximation in Eq. (7), we should assign the mass probability $\widehat{p}_j = \mathbb{P}(\widehat{Z}_j = \widehat{z}_j)$ to each scenario $\widehat{z}_j$. The latter depends on the amount of scenarios assigned to each cluster. Thus, by computing the ratio between the amount of data points assigned to each cluster and the total amount of points in the dataset, the probability $\widehat{p}_j$ can be defined, i.e. $\widehat{p} = |C_j(\widehat{z}_j)|/N$.

2.3 Stress tests and scenarios labeling

The unsupervised learning approach for scenarios aggregation only uses the information provided by the discrete distribution F of the random variable Z. So far, we ignored the information gathered using the various initial levels used by the insurer. As we already mentioned, the decomposition $f = \Upsilon \circ \Psi$ brings in additional information through the separation of the random scenarios from the initial levels X_0 as provided by the stress tests used to assess the liability. This allows to assign multiple labels $f(z_j, x_i)$ to each scenario z_j and thus enrich the initial datasets. By doing so, we can enhance the clustering based on the sole distance between the initial scenarios. Indeed, the unsupervised technique, i.e. k-means, defined in the last section advocates grouping similar scenarios together by optimizing some similarity measure, Euclidean distance, for each cluster such as within group variance. Therefore, it is not necessarily guaranteed to group the same type of scenarios together. On other words, the same paths of Z could not lead to the same impact on the insurer balance sheet. This will undoubtedly depend on the complex mechanisms inherited in the asset and liability projection model as well as the initial levels of the risk factors. Accordingly, when introducing the labels $f(z_j, x_i)$, we shall incorporate some degree of supervision to the learning scheme through, which is used to guide the clustering process. The idea is to find a set of aggregated scenarios $\widehat{z}_j$ that minimize the distance measure in (10) under some constraints, which can be used to penalize misplaced known labels. To achieve this, recent work has looked at extending the k-means algorithm in (10) to incorporate such an additional information in the form of instance level must-link and cannot-link constraints, see (Davidson & Ravi, 2005). Formally, a must-link constraint enforces two scenarios z_l and z_k to be placed in the same cluster $C_j(\widehat{z}_j)$ as soon as $f(z_l, x_i) = f(z_k, x_i)$. Conversely, a cannot-link constraint enforces that two scenarios must not be placed in the same cluster as soon as their labels are different. These techniques have been widely explored and applied in *single-label* settings that are concerned with learning from a set of examples associated with a single label from a disjoint labels. We should mention that this is also applied for datasets that gather both labeled and unlabeled samples.

2.4 A semi-supervised learning approach

In order to deal with the aforementioned labels, a semi-supervised clustering could be the best candidate. We will combine the benefits of supervised and unsupervised learning methods. We advocate the use of a semi-supervised technique instead of a fully supervised one that could be based either on a direct learning of the multiple labels $f(z_j, x)$ or the value function $V(x)$, for $x = x_1, \ldots, x_d$, to avoid any over-fitting of the underlying model. This was the case, for instance, in Bignon et al. (2019) where a fully parametrized model is constructed describing the value function $V(x)$ in terms of the initial levels x. First, the training dataset used required a tremendous time using various and varied scenario's levels x. Also, it is shown that such an approach is efficient as soon as the insurer balance sheet, in particular the asset mix, remains stable over periods. In our case, we will add a supervision that could be handled adequately in such a way to aggregate the scenarios with having similar paths but also comparable impact in the cash flows. As discussed in Chauvigny et al. (2011), each path z_i in the scenarios table amend to cash flows that impact differently the valuation function. They advocate measuring the risk associated to each simulation and to order them. Thus, most adverse simulations in terms of solvency are considered. This classification of scenarios leads to build an execution area with the points that seem the most risky. Hence, we can consider the *impurity* present in the clusters with regards to their associated risks. Formally, in the presence of these multiple labels, we advocate measuring the the impurity of a cluster according to its composition of available labeled samples, i.e. impact on the cash flows. For instance, we can use the variance of the cash flows within the clusters. This impurity measure is low if all the scenarios of a cluster have comparable impacts on the liability. Finally, we aim at finding clusters that minimize a linear combination of the cluster dispersion and cluster impurity as follows

$$\min_{\widehat{z}_1,\ldots,\widehat{z}_K} \left\{ \sum_{j=1}^{K} \sum_{z_i \in C_j(\widehat{z}_j)} \left(\|z_i - \widehat{z}_j\|^2 + \alpha \| f(z_i, x) - \widehat{m}_j \|^2 \right) \right\}. \tag{11}$$

where α is positive regularization parameter, $x = (x_1, \ldots, x_d)$ and $m_j = \sum_{z_i \in C_j(\widehat{z}_j)} \sum_{x=x_1}^{x_d} f(z_i, x)$. If $\alpha = 0$, then the result is a purely unsupervised clustering algorithm. Notice that in this optimization we can use the $\| f(z_i, x)\|^2$ as a constraint instead of the variance, which requires an adequate reduction of the cash flows for each cluster. In fact, for each cluster we compute the $\sum_{z_i \in C_j(\widehat{z}_j)} \sum_{x=x_1}^{x_d} (f(z_i, x))^2$ as a way to measure the overall within-cluster variability of cash flows. The choice of the variance is a way to measure the impurity in the extent that the clusters are constructed in such a way to decrease the disparities in terms of the labels between the scenarios. Of course, we can also consider other monetary risk measures to cope with the application concerns that are mainly looking the impact of each scenario on the solvency of the insurer. We should note that these optimization problems, are generally difficult to solve, since the objective function (10) is highly discontinuous with many local minima.

2.5 Optimization algorithms

There are several algorithms for optimizing the unconstrained (unsupervised) problem in (10). Here, we will focus on the k-means algorithm, also known as nearest centroid sorting, and its variants. Such an algorithm involves an iterative scheme that operates over a fixed number K of clusters, while attempting to satisfy the following properties: i) each class has a center $\widehat{z}_j$ which is the mean position of all the scenarios in that class, i.e. $\widehat{z}_j = \sum_{z_i \in C_j(\widehat{z}_j)} z_i / |C_j(\widehat{z}_j)|$; ii) each scenario z_i is in the class whose center it is closest to. The classic algorithm, referred to as the batch k-means, starts with some initial configuration of the scenarios, then it iteratively improves its estimate of the cluster centroids $\widehat{z}_j$, for $j = 1, \ldots, K$, until no further changes are possible. It does this by alternating between assigning all points to their closest cluster centers and updating the cluster centers, see (MacQueen, 1967). However, recall that the objective problem in (10) can have many local minima and thus such an algorithm does not minimize the cost function in general. Specifically, it is shown that the local minimum found strongly depends on the initial values of the cluster centers $\widehat{z}_j$. To improve the quality of the clustering and avoid suboptimal solutions, it is generally recommended, in practice, to run repeatedly using random initializations. This, however, is expensive for large sets of scenarios. A variant called *mini-batch*, introduced in Sculley (2010), proposes to assign a uniformly selection fraction of data at each iteration. The advantage of this algorithm is to reduce the computational cost by not using all the data-set each iteration but a sub-sample of a fixed size. This strategy reduces the number of distance computations per iteration at the cost of lower cluster quality. There is also the balanced iterative reducing and clustering algorithm using hierarchies (Birch) which is a hierarchical clustering and memory-efficient algorithm, see (Zhang et al., 1996). It offers an alternative to the mini-batch algorithm that can also be used to accelerate k-means clustering. We can also mention the agglomerative algorithms that work in the opposite way. In fact, agglomerative methods perform hierarchical clustering using a bottom up approach. A typical algorithm begins with a given fixed number of clusters and successively merge together similar clusters.

These algorithms can be used in our case with little impact of the constraint. In fact, most of these algorithms alternate between assignment of the vectors Z_i to clusters and centers updates. Therefore, at each step the variance of the cash flows can be computed over each cluster and serves to update the dispersion of the cluster while keeping the centeroids unchanged. By doing so, we will undoubtedly increase the computation time needed to achieve a given degree of convergence. To handle such undesired behavior, we can make use of algorithms that are well suited to constrained optimization. For instance, Demiriz et al. (1999) advocate using *genetic algorithm* for such problems.

3 Empirical and numerical analyses

In this section, we illustrate the methodology developed in the previous sections. We use two examples in which we compare our method on toy examples with the analytical outputs to investigate its performance. Hence, we will analyze a real-world example based on a life insurer's data set.

3.1 Illustrative examples

We will dedicate this section to well studied example in order to assess the efficiency of the proposed methodology. To this end, we consider a simple yet practical case which will serve as a reference. Here, we consider a Vanilla put option, whose payoff is given as $\Upsilon(X_T) = e^{-rT}(X_T - C)^+$, where X is given as a solution of the Black & Scholes dynamics, see Table 1. Here, we will focus on the analytical form of the process $X_T = \Psi(Z, X_0)$ as a function of a Gaussian variable Z and the initial value X_0, where $\Psi(Z, X_0) = X_0 \exp\left((r - \sigma^2/2)T + \sigma\sqrt{T}Z\right)$, with $r = 0.01, \sigma = 0.2, C = 100$ and $T = 2$. In this example, the mapping $(z, x) \mapsto f(z, x)$ can be given analytically as discussed previously, see Sect. A.1. Let us, first, consider the case where $\alpha = 0$, which corresponds to an unsupervised scheme based on the sole observations of the simulated random Gaussian variable Z. Here, we let $N = 1000$ and $\lambda = 0$ and thus consider the optimization problem in Eq. (10). Recall that in this case, the original distribution corresponds to the empirical Gaussian distribution with N copies, whereas the targeted distribution has M elements. These M elements correspond to the number of scenarios aimed at representing the original distribution. In Fig. 1, we check graphically the quality of the representative scenarios for $K = 20$ obtained after convergence of the method by drawing the weight function $\widehat{p}_j = \mathbb{P}(\widehat{Z} = \widehat{z}_j) = |C_j(\widehat{z}_j)|/N$. For each cluster, the representative scenarios $\widehat{z}_j$ corresponds to the average over each bar of the histogram in the bottom panel of Fig. 1. Based on these clusters, we estimate the price $V^K_{\mathrm{ML}}(x) = \sum_{j=1}^{K} \widehat{p}_j f(\widehat{z}_j, x)$, where $f(\widehat{z}_j, x)$ are given in the top panel of Fig. 1. We compare the prices obtained from this unsupervised method with those obtained by the Monte Carlo simulations on reduced sample size $M < N$, denoted $V^M_{\mathrm{MC}}(x)$, for various values $x = 70, 80, 90, 110, 120$ and 130. In this setting the true prices $V_{\mathrm{BS}}(x)$ are available and will serve us as a support for comparisons, see Sect. A.1. Letting $x = 100$, numerical results are printed in the first panel of Table 1. In order to compare the proposed methodology in terms of variance reduction to the reference, we also compute the variance for the small sized Monte-Carlo replicates with $M = 20$. The variance of these Monte-Carlo experiment are derived based on semi-analytical formula given in Sect. A.1. First, we notice that the clustering approach provide variances that close the initial simulations whereas the reduced sized Monte-Carlo experiment behaves poorly. The latter produced high variances, which is in line with had been expected. The same conclusions can be drawn when considering, for instance, the case of an increased volatility σ ranging from 10% to 40%, see the second panel of Table 1, cash flows related to deeply out of the money call option and so on. As soon as the bias is considered,we can see that the prices provided by the unsupervised learning always belong to the confidence interval of the true prices. One can note that it is true even for values of M, which means that the methodology gives quickly a good price approximation, in particular given the reduced variance. Moreover, the price given by the unsupervised learning is very close of the theoretical price in most of the configurations discussed above. Exception is made for cases where the option is deeply out-the-money. To understand this, we plot in Fig. 1 the level (green hashed line) at which the call option is at-the-money. Below this level, the cash flows do not contribute to price as they are equal to 0. When the call option is deeply out-of-the-money, we are left with small amount of scenarios leading a positive cash flow. In other terms, this implicitly comes down to moving the vertical line in the top panel of Fig. 1 while keeping the left-hand side values equal to 0. Consequently, the aggregated scenarios still keep the same distribution and thus provide little information on the behavior of the cash flows for this particular case. To cope with this, we should assigns high probability to regions in the distribution of Z on which the cash flows are positives. This appeals to the

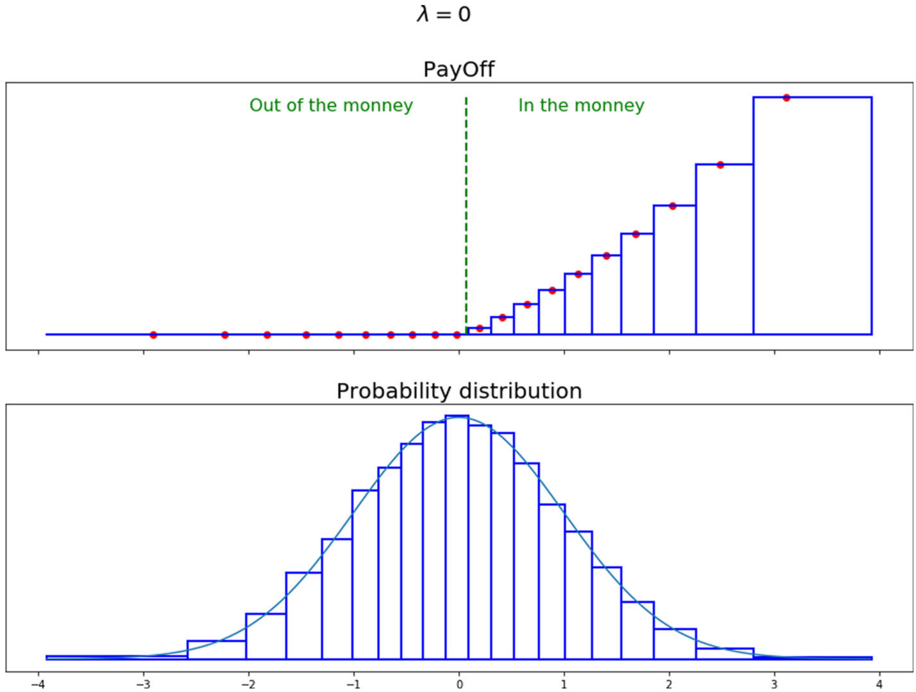


Fig. 1 Bottom panel: The mass probabilities $\widehat{p}_j$ of the representative clusters $C(\widehat{z}_j)$ as a function of the representative components $\widehat{z}_j$ for $j = 1, \ldots, K$, with $K = 20$. Top panel: Cash flows representation over clusters as well as their respective representative $f(\widehat{z}_j, x)$ in red circles. (Color figure online)

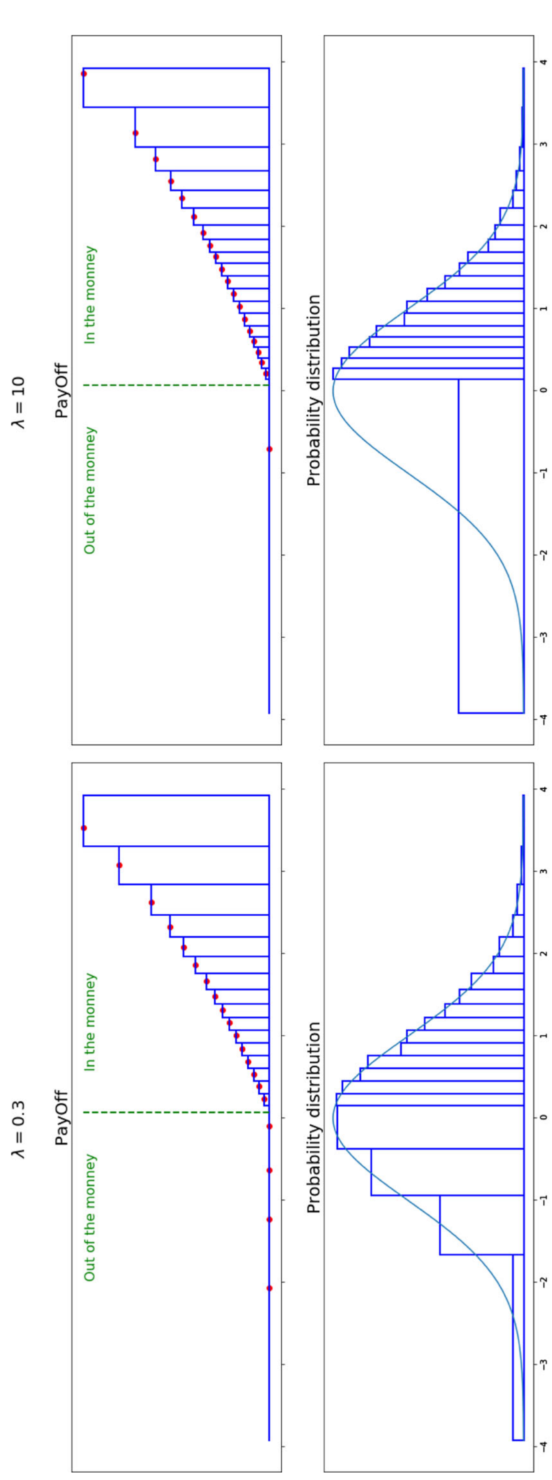
importance sampling techniques that aims to concentrate simulation on sample paths that contribute most to estimating the expected payoff for instance.

In view of these primary results, we add a supervised layer using the constrained optimization in (9). In order to perform the optimization algorithm, we will use the k-means algorithm to solve the problem. More precisely, we will enlarge the support of the parameter α in order to penalize adequately the variance of the labels. Thus, we will assume that the k-means algorithm is performed over the vector $(Z_1, Z_2, \ldots, Z_N, \lambda f(Z_1, x), \ldots, \lambda f(Z_N, x))$, with $\lambda = \sqrt{\alpha}$. The choice of λ will be done in such a way to avoid any alternation of the initial vector of the scenarios. Of course, we should note that we can implement an adequate algorithm as discussed in Sect. 2.5. Here, we train our model using the estimated values based on the initial value $x = 100$. In Fig. 2, we depict the mass probabilities against the clusters representatives (bottom panel) as well as the corresponding cash flows distribution (top panel) for different values of λ. First, as discussed earlier λ increases, the learning approach tends to give more weight for the labels among the cluster instead of the scenarios. Ultimately, this will correspond to a supervised learning looking representative scenarios that fully describe the resulting cash flows. In fact, as we can see in the right panel of Fig. 2, for $\lambda = 10$ the representative scenarios at the left side of the distribution are aggregated in the same cluster. This is due to the fact that the cash flow for these scenarios is null as the call options is deeply out-the-many. However, the remaining $K - 1$ clusters are split over the points for which the cash flows take positive values (top panel). By doing so, we concentrate the information on scenarios that account for the price V. Hence, reducing the importance of the supervision, i.e. the cash flows (e.g. $\lambda = 0.3$) decreases this impact and, as we can see in the left panel of Fig. 2, the scenarios on the left hand side of the distribution take more importance even if the corresponding cash flows do not contribute to the computation of the desired prices at this initial level $x = 100$. Consequently, when we train the clustering algorithm on this particular case and applying it to other values not included in the training set, we will undoubtedly recover a bias that will decrease with the degree of importance we assign the supervision, i.e. α. For instance, in the right panel of Table 1, we reported the bias as well as the variance for the test set composed of cash flows for initial values $x = 70, 80, 90, 110, 120$ and 130. Here, we can clearly see the aforementioned impact as soon as the bias is relatively smaller for $\lambda = 0.3$ compared to $\lambda = 10$. Notice that the desired impact is, however, too small to draw immediate conclusions. This is principally due to the training set which only consists of a single initial value. In practical applications, we should therefore include a representative and dense panel of initial values x in order to span as much as possible all possible outcomes. This will be very beneficial for the learning behavior and will enhance the out-of-sample predictions (Fig. 3).

3.2 Path-dependent cash-flows

In this section we use the machine learning scheme to price a (lookback) path-dependent put option over the running supremum of the equity, i.e. $V(x) = \mathbb{E}[e^{-rT}(\max(\max_{0\le t\le T}(X_t), X_0) - C)^+ | X_0 = x]$, where X's dynamics is as a solution to Black & Scholes dynamics. In this case, we consider the following set of parameters: $r = 0.01, \sigma = 0.3, T = 1, N = 5000$ and $C = 150$. To implement the estimation, we consider an Euler scheme to discretize the dynamics using a time step $\Delta t = T/n$ such that $n = 250$. This is a typical example where the mapping f does depend on the whole dynamics of the underlying process and thus using the discretization can be written in terms of the initial value x as well as the random vector Z. Here, we aim to exhibit the impact of the supervision in learning the aggregation mechanism more suited to complex cases. First, we consider the unsupervised approach that consists in aggregating the paths of the Brownian motion over the discrete grid, i.e. $\lambda = 0$. In Fig. 4a we depict the representative scenarios for $K = 10$, which as we can see are smooth representative paths that span the domain of the support for the random vector Z. This is a typical representation similar to the quantization discussed in Remark 3. Adding a supervision, with $\lambda = 10$, based on the knowledge of the outcomes for $x = 150$ tends to distort the representative scenarios in order to adapt to this latter information, see Fig. 4b. When can draw similar conclusions as in the above example, however, it is not easy to have a graphical underpin in that sense. Next, we show in Fig. 4c another example of supervision with an enlarged set of labels covering initial values in $\mathcal{X} = \{100, 105, 110, \ldots, 200\}$.

In Table 2, we report the outputs of comparison in terms of the bias as well as the variances. In the top panel, we investigate the bias for different configuration detailed in Fig. 4, whereas in the bottom panel we reported the corresponding variance. In order to exhibit the benefit of the methodology, we consider small amount of Monte-Carlo simulations, $M = 20$, and the same amount of the representative scenarios, $K = 20$. First, let us focus on option that are at-the-money, $S_0 = 150$ and 200. Without the supervision constraint the representative scenarios succeed to provide a satisfying estimation of the option price compared to the Monte-Carlo estimation. These estimations perform better than the Monte-Carlo experiments based on $M = 20$ Brownian paths. This is particularly appealing if we regard the induced variances. In fact, these Monte-Carlo simulations reproduce more ten times the initial uncertainty while the scenario reduction technique keeps the variance closer to the initial level. On the other hand,



(a) $\lambda = 0.3$

(b) $\lambda = 10$

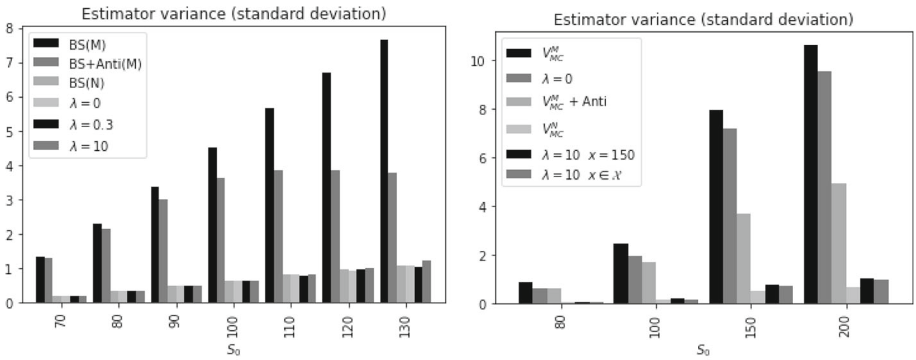
Fig. 2 Illustration of the supervision impact on the clustering mechanism with $\lambda = 0.3$ (left panel) that corresponds to a slight degree of supervision and $\lambda = 10$ (right panel) which provides a near supervised clustering of the random variable Z

Table 1 Comparison of the estimated prices of European call option based on Monte-Carlo simulations and scenarios reduction scheme with and without supervision

S_0	C	σ (%)	Estimator variance						Estimator mean			
			BS (M)	BS+Anti (M)	BS (N)	$\lambda = 0$	$\lambda = 0.3$	$\lambda = 10$	BS	$\lambda = 0$	$\lambda = 0.3$	$\lambda = 10$
70	100	20	1.337	1.303	0.189	0.190	0.191	0.189	1.332	1.311	1.326	1.328
80	100	20	2.273	2.138	0.321	0.320	0.322	0.320	3.450	3.424	3.441	3.442
90	100	20	3.367	2.977	0.476	0.473	0.473	0.474	7.037	7.005	7.027	7.026
100	100	20	4.517	3.608	0.639	0.633	0.631	0.636	12.153	12.118	12.124	12.127
110	100	20	5.642	3.857	0.798	0.792	0.772	0.806	18.638	18.600	18.393	**17.726**
120	100	20	6.696	3.855	0.947	0.938	0.956	0.990	26.226	26.183	26.081	**23.325**
130	100	20	7.662	3.756	1.084	1.072	1.042	1.216	34.632	34.585	34.381	**31.133**
100	100	10	2.153	1.562	0.304	0.302	0.301	0.304	6.627	6.614	6.509	6.511
100	100	20	4.517	3.608	0.639	0.633	0.631	0.636	12.153	12.118	12.124	12.127
100	100	30	7.312	6.157	1.034	1.027	1.026	1.030	17.642	17.572	17.624	17.623
100	100	40	10.633	9.300	1.504	1.492	1.493	1.493	23.053	22.931	23.017	23.027
100	80	20	5.752	3.049	0.813	0.805	0.809	0.892	24.275	24.238	24.100	**21.052**
100	100	20	4.517	3.608	0.639	0.633	0.631	0.636	12.153	12.118	12.124	12.127

Table 1 continued

S_0	C	σ (%)	Estimator variance						Estimator mean			
			BS (M)	BS+Anti (M)	BS (N)	$\lambda = 0$	$\lambda = 0.3$	$\lambda = 10$	BS	$\lambda = 0$	$\lambda = 0.3$	$\lambda = 10$
100	120	20	3.151	2.913	0.446	0.443	0.445	0.444	5.369	5.336	5.358	5.358
70	100	20	1.337	1.303	0.189	0.190	0.191	0.189	1.332	1.311	1.328	1.330
80	100	20	2.273	2.138	0.321	0.320	0.321	0.320	3.450	3.424	3.437	3.442
90	100	20	3.367	2.977	0.476	0.473	0.474	0.474	7.037	7.005	7.001	**6.134**
100	100	20	4.517	3.608	0.639	0.633	0.627	0.643	12.153	12.118	12.070	**8.825**
110	100	20	5.642	3.857	0.798	0.792	0.780	0.823	18.638	18.600	18.522	**11.521**
120	100	20	6.696	3.855	0.947	0.938	0.935	1.136	26.226	26.183	26.105	**20.000**
130	100	20	7.662	3.756	1.084	1.072	1.059	1.229	34.632	34.585	34.508	**29.835**
70	100	20	1.337	1.303	0.189	0.190	0.191	0.189	1.332	1.311	1.324	1.325
80	100	20	2.273	2.138	0.321	0.320	0.322	0.320	3.450	3.424	3.438	3.439
90	100	20	3.367	2.977	0.476	0.473	0.472	0.474	7.037	7.005	7.022	7.022
100	100	20	4.517	3.608	0.639	0.633	0.631	0.637	12.153	12.118	12.135	12.136
110	100	20	5.642	3.857	0.798	0.792	0.788	0.797	18.638	18.600	18.621	18.623
120	100	20	6.696	3.855	0.947	0.938	0.929	0.944	26.226	26.183	26.176	26.178
130	100	20	7.662	3.756	1.084	1.072	1.088	1.093	34.632	34.585	34.077	33.953



(a) Lookback option **(b)** Call option

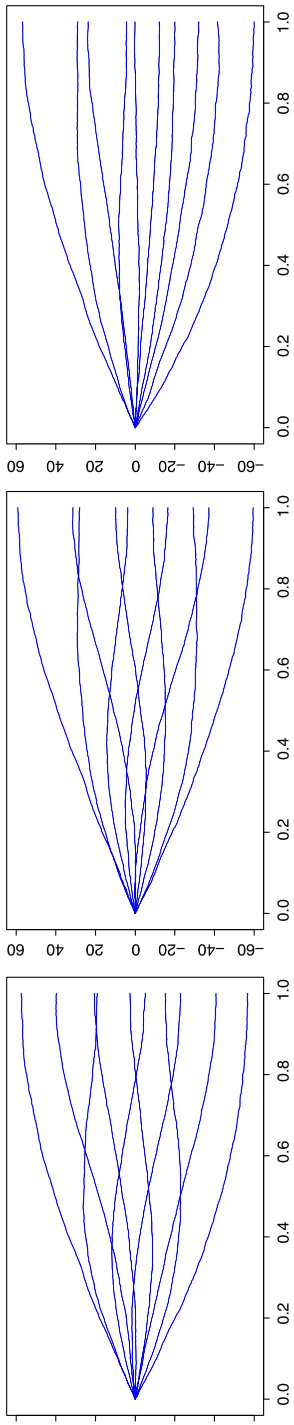
Fig. 3 Performance of the considered variance reduction methodologies reported in Table 1 for different values of S_0

Table 2 Comparison of the estimated prices of the path-dependent option based on Monte-Carlo simulations and scenarios reduction scheme with and without supervision

S_0	Variance (standard deviation)						Mean			
	V^N_{MC}	V^M_{MC}	Anti	$\lambda = 0$	$\lambda = 10$		BS	$\lambda = 0$	$\lambda = 10$	
					$x = 150$	$x \in \mathcal{X}$			$x = 150$	$x \in \mathcal{X}$
80	0.054	0.855	0.600	0.616	0.067	0.065	0.434	0.175	0.433	0.429
100	0.156	2.469	1.682	1.934	0.203	0.168	3.039	1.867	3.012	3.009
150	0.504	7.965	3.687	7.165	0.746	0.729	37.844	31.887	37.730	37.774
200	0.672	10.620	4.916	9.554	0.994	0.972	99.962	92.018	99.810	99.868

when we look at the outputs of the out-of-the money case, $S_0 = 100$, we see that the learning scheme only based on the knowledge of the labels for the initial value case $x = 150$ provides satisfying estimations compared to the theoretic based on a closed formula, see (Conze, 1991). In fact, compared to column (BS) in Table 2, all the schemes allows to estimate the desired price with a controlled bias. When we enlarge the labels set by introducing additional outputs, i.e. $\mathcal{X}$, we enhance the estimation as the learning constraint takes into account extreme scenarios and assign Brownian paths reproducing the latter to specific clusters. However, it is still insufficient to capture the behavior of the cash flows, which requires to increase the number of the clusters.

Additionally, we also compare the performance of the proposed methodology with a conventional variance reduction technique. Here, we use antithetic variates method taking two simulated paths of the Brownian motion of opposite signs, see (Glynn, 1994). In Table 1, we report the analytic results for such a methodology which provides the quantification of the variance for different sizes, i.e. M, of the underlying sample, see Sect. A.3. In the left panel of Fig. 3, we also depict the performance of this methodology with regard to the machine learning based one. We can observe that the variance of the Monte-Carlo based estimation of the lookback option with small amount of simulated paths, $M = 20$, has six times greater uncertainty compared to the proposed methodology with identical number of retained clusters, $K = 20$. This remains valid even if we use an antithetic variance



(a) $\lambda = 0, x = 100$

(b) $\lambda = 10, x = 100$

(c) $\lambda = 10, x = 50, \cdots, 250$

Fig. 4 Illustration of the supervision impact on the clustering mechanism with $\lambda = 0.3$ (left panel) that corresponds to a slight degree of supervision and $\lambda = 10$ (right panel) which provides a near supervised clustering of the random variable Z

reduction method. The latter succeeds to decrease the variance compared to a naive simulation scheme but has performance that less efficient compared to the our methodology. The same conclusions can be drawn when looking at the call option case, see the right panel of Fig. 3.

3.3 Real-world dataset

In this section, we consider a real-world data set drawn from a French insurer's projection model. The insurer's assets are mainly constituted of fixed income securities (78%), UCITS[1] (13%), equities (5%) and properties (4%). The composition of assets and liabilities remained stable over the considered period of the current study, which is very common in life insurance. In our case, major changes can come from financial market movements, which later be captured through the estimation of sensitivities. We have at our disposal cash flows for 3 consecutive valuation dates, i.e. 2017, 2018 and 2019, with $N = 10{,}000$ simulated paths of 6 economic factors; i.e. equity index, property index, short and long term interest rates, dividends and inflation. As noted in Sect. 1, these paths rely on the specification of the economic scenarios generator and are based on simulations of dicretized dynamics of correlated Brownian motions as in Eq. (1). Here, an Euler scheme were used based on a monthly grid over the projection period. Hence, the insurer liabilities are projected over $T = 40$ and various initial levels x were investigated. These correspond to shocks on the main determinant risk factors. Formally, these are based on the set of all information required for the valuation on the initial date: asset information (the set of all the financial assets of the company, usually more than a thousand asset products); liability information (the set of all insurance liabilities, which can be over million policies); current economic situation (interest rate curve, equity and property volatility term structure, swaption volatilities, etc.); policyholders behavior; insurer behavior; etc. Over all, we have $d = 14$ different shocks on these factors together with combined shocks that encompass the potential mixed effects that can impact the balance sheet. These variants correspond the Solvency 2 standard shocks. In Fig. 5, we depict the simulated paths of the random vector Z, which we separated into the 6 economic factors. Hence, a ranking based on the impact on the cash flows $f(Z_i, x)$ was added to visualize the complex structure of these scenarios. We observe that some regions of the factors are strongly related to higher or lower values of cash flows, which is in line with the initial intuition on the relation between the scenarios Z_i and $f(Z_i, x)$ that we will try to capture. For instance, this relation is very clear for equity where high equity implies high impact on the cash flows. The same expected behavior is seen for property and long term interest rate as well as dividends but on the opposite direction. For short term interest rates, central values give higher values of cash flows while extreme values give lower values of $f(Z, x)$. On the other hand, inflation does not seem to give a clear pattern. This makes the isolation of the impact of the scenarios difficult to address. In fact, it is not clear how to measure the risk associated to each scenario and to order them. The machine learning, will then serve to capture the proximity between these scenarios and thus allow for an adequate aggregation. However, as noted above, we shall need a supervised component to direct the learning of the risk associated to each scenario. Also, as discussed in the preceding section, we should enrich the set of the initial values x in order to span as much as possible their domain of definition and thus allow for an efficient computation of the variance constraint and avoid any bias.

We will perform the optimization in Eq. (9) based on the sole information stemming from 2017 and will use the remaining data for cross validation and test. Since the optimization in this real world case can take time to converge to a stable local minima, we will modify the

[1] Undertakings for the Collective Investment of Transferable Securities

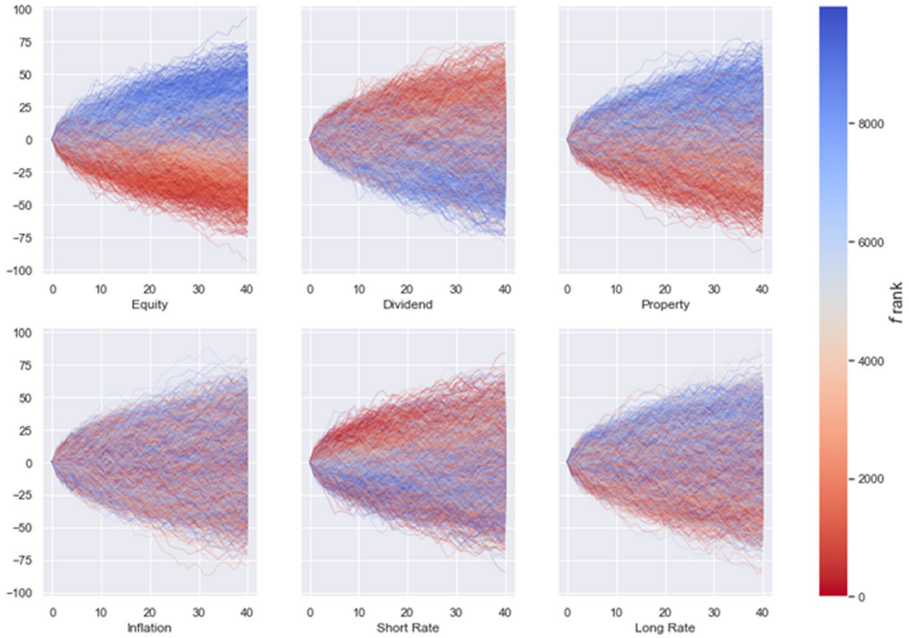


Fig. 5 Economic scenarios and their impact on the cash flow map $f(Z, x)$ as of 2017, for different risk factors: short and long term interest rates, inflation, equity, dividend and property risks

criterion in Eq. (9) without altering the underlying reasoning. Indeed, we propose to consider the vector $Y = (Z_1, \ldots, Z_n, \lambda f(Z_1, x_1), \ldots, \lambda f(Z_n, x_d))$ over which we will apply the unsupervised clustering algorithm as in Eq. (10). By doing so, we should adequately choose the weight parameter λ in order to diminish the impact of the cash flows amplitude. In other words, the parameter λ is used to balance the order of magnitude of the two component of Y. In our case, Z has values between -100 and 100, while $f(Z, x)$ has values between $-20{,}000$ and 30,000. It is then of paramount importance to tune this parameter with respect to the domain space of Z and $f(Z, x)$ when we perform the optimization. To this end, we train the model on the basis of 2017, white keeping 2018 information to tune hyperparameters and will do the final test on the 2019. We will consider $K = 500$ representative scenarios and compare the outputs to a Monte-Carlo scheme based on the same amount of randomly drawn scenarios.

In Fig. 6 we depict the densities of the simulated cash flows $f(Z, x)$ using respectively 500 and 3000 paths and compare the latter to the one recovered using the unsupervised scheme using $K = 500$ aggregated scenarios. We also compare three clustering algorithms discussed in Sect. 2.5, i.e. Birch, agglomerative clustering and k-means. In this figure, we check graphically the quality of representative scenarios by drawing the weight function. This allows to investigate the bias and the variance of the methodologies. First, we observe that the three considered algorithm provide similar results, which is consistent with the earlier discussion. In fact, the main difference between the three is the time needed to converge toward a local or global optimum. Next, when we compare their densities to the Monte-Carlo scheme, we se clearly that those resulting from the unsupervised schemes with $K = 500$

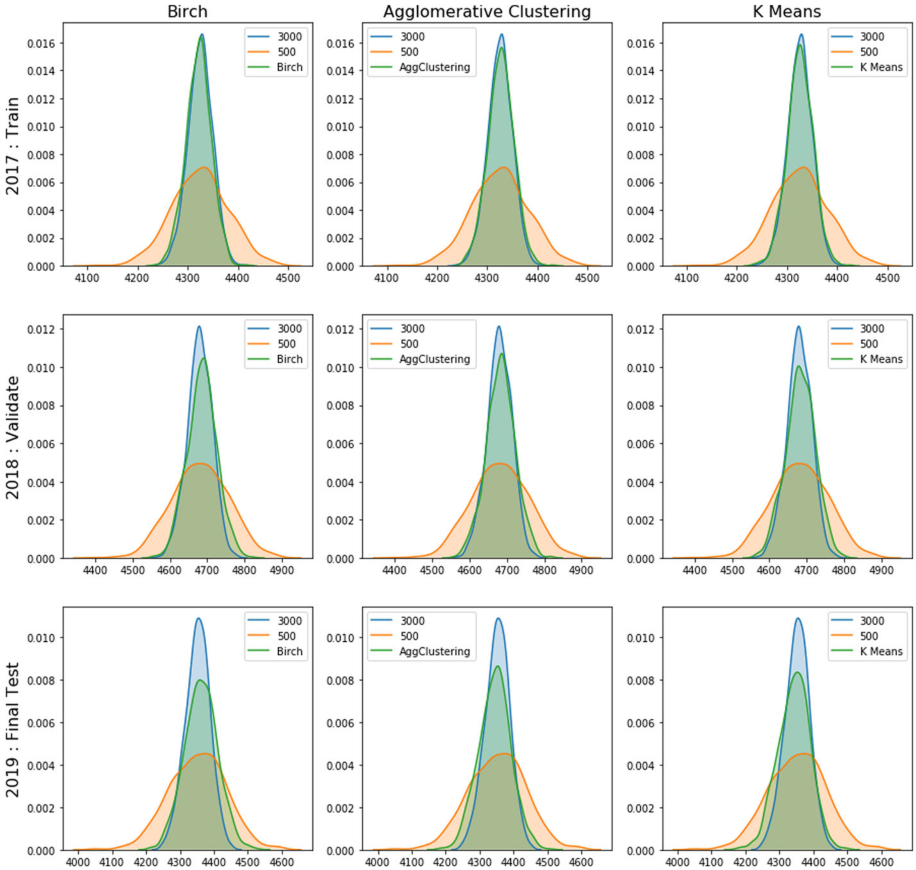


Fig. 6 Comparison of the insurer's simulated cash flows density using $M = 3000$ scenarios (blue) with those based recuded number of scenarios $M = 500$ (orange) and unsupervised learning with $K = 500$ (green) for the training (top), validation (middle) and test (bottom) sets. (Color figure online)

clusters are close to the Monte-Carlo simulations with 3000 scenarios. This is for the training set as well as the validation and more interestingly the testing set.

We should highlight that the behavior of the unsupervised learning may be altered in situations where the insurer's balance sheet undergoes structural changes in its components. It is in fact possible that some determinants time evolution, e.g. the asset mix, may change drastically the outcomes and thus induce a biased estimation in which case the estimated value to be taken carefully, see (Bignon et al., 2019). Therefore, any changes in the assets mix or other structural variables may have definitely implications on the insurer's financial cash flows and thus the conclusions of the above discussion. It is, however, possible to adapt the model while taking into account this change of the asset mix. This will allow, for instance, to learn the behavior resulting from such a component. We should, however, highlight that this may be complicated to implement in practice.

3.4 Practical implications

The variance reduction technique presented in this article requires a sufficiently large set of simulations as an initial step, which serves to train and validate the model. Once this first calculation is done and the reduced number of simulations chosen, a substantial gain in computing time is acquired. For such a technique to have a practical interest these two steps must be considered. In fact, insurance companies have different calculations that are done during the year. Usually, at the end of each year, results are of capital importance as they are the main figures transmitted to investors and regulators. Depending on the size of the company and its internal policy, Monte-Carlo based calculation can be also done on a quarterly or monthly basis. Thus, our variance reduction technique can be, for example, calibrated at the end of the year using the full set of simulations, and used for periodic calculations for the next year. This implicitly assume that the yearly assumptions are sufficiently robust and accurate while the quarterly of monthly could be less precise. Moreover, such a technique can also be used in sensitivity studies or in the analysis of change, which is required whenever the insurer results exhibit a variation from one period to another. Those are more frequently implemented so that fast calculation times are more than required. On the other hand, such a methodology can be very useful when faced with a nested stochastic problem such as asset portfolio optimization, where the double stochastic run can gain enormously by using a reduced set of secondary simulations. As the insurance company is continually evolving, it is recommended to calibrate the variance reduction technique at least once a year.

4 Concluding remarks

In this paper, we tackled the problem of scenarios and variance reduction, which are two sides of the same coin, in life insurance liabilities valuation under different initial balance sheet situations. The latter is characterized using various risk factors which are solutions to some stochastic differential equations. To this end, we proposed a machine learning-based methodology which is based on an unsupervised clustering of the discrete Brownian paths used to generate the insurer cash flows. This is made possible thanks to a decomposition of the cash flows allowing to disentangle the scenario table from the risk factors. By doing so, we can adequately reduce the dimensionality of the underlying scenarios while keeping the variance closer to the original setting. This desired result is investigated based on an experimental example using a single initial risk factor. While the clustering approach is shown to behave efficiently as regards to a comparable Monte-Carlo approach, the changing initial and endogenous risk factors alter its performance. Hence, a constraint is added to the problem in order to help assign the scenarios to the clusters that best describe their impact on the cash flows. This supervised component make the methodology falls into the semi-supervised learning algorithms which aggregate the economic scenarios aimed at representing the original ones while taking into account the labels assigned to each ones. These labels stems from the decomposition of the cash flows mapping and are related to the initial level of the underlying risk factors. This is very convenient as the algorithm takes into account various outcomes resulting from different labels. Therefore, the trained algorithm would adapt to new situation. The proposed methodology is discussed using a toy example allowing to understand the impact of the supervision constraint as well as its limitation. We show, numerically, that this scheme outperforms a naive Monte-Carlo approach. Finally, we perform a more realistic experiment based on real-world data sets to investigate the practical aspect of the proposed methodology.

Acknowledgements The authors would like to thank the anonymous referees for their constructive comments that improved the quality of this paper.

A Appendices

A.1 Analytical variance of the call option

We will derive the theoretic variance of the cash flow $f(Z,x) = e^{-rT}(x\exp((r-\sigma^2/2)T + \sigma\sqrt{T}Z) - C)^+$, where Z is a random Gaussian variable. First, we consider

$$\begin{aligned}\mathbb{E}[f(Z,x)] &= \mathbb{E}\left[e^{-rT}(xe^{(r-\sigma^2/2)T+\sigma\sqrt{T}Z} - C)^+\right] \\ &= \mathbb{E}\left[(x\exp(-\sigma^2T/2 + \sigma^2\sqrt{T}Z) - Ce^{-rT})1_{Z+d_2\geq 0}\right],\end{aligned}$$

where $d_2 = \left(\log(C/x) + (r-\sigma^2/2)T\right)/\sigma\sqrt{T}$. Hence, we have the classic Black & Scholes analytic formula

$$\begin{aligned}\mathbb{E}[f(Z,x)] &= \int_{-\infty}^{d_2}\left(xe^{-\sigma\sqrt{T}y-\sigma^2T/2} - e^{-rT}C\right)\frac{1}{\sqrt{2\pi}}e^{-y^2/2}dy \\ &= \int_{-\infty}^{d_2} xe^{-\sigma\sqrt{T}y-\sigma^2T/2}\frac{1}{\sqrt{2\pi}}e^{-y^2/2}dy - \frac{e^{-rT}C}{\sqrt{2\pi}}\int_{-\infty}^{d_2} e^{-y^2/2}dy\end{aligned}$$

Hence, letting $u = y + \sigma\sqrt{T}$, we come up the well-celebrated Black & Scholes formula

$$\begin{aligned}\mathbb{E}[f(Z,x)] &= \frac{x}{\sqrt{2\pi}}\int_{-\infty}^{d_2} e^{-\sigma\sqrt{T}(u-\sigma\sqrt{T})-\dfrac{\sigma^2T}{2}-\dfrac{(u-\sigma\sqrt{T})^2}{2}}du - Ce^{-rT}\phi(d_2) \\ &= \frac{x}{\sqrt{2\pi}}\int_{-\infty}^{d_1} e^{-u^2/2}du - Ce^{-rT}\phi(d_2) \\ &= x\phi(d_1) - Ce^{-rT}\phi(d_2),\end{aligned}$$

where $d_1 = d_2 + \sigma\sqrt{T}$, and ϕ is the standard normal cumulative distribution function. In order to come with the variance of cash flows, we need to compute the following quantity

$$\begin{aligned}\mathbb{E}\left[f(Z,x)^2\right] &= \mathbb{E}\left[\left(e^{-rT}(xe^{(r-\sigma^2/2)T+\sigma\sqrt{T}Z} - C)^+\right)^2\right] \\ &= \mathbb{E}\left[\left(x\exp(-\sigma^2T/2 + \sigma\sqrt{T}Z) - Ce^{-rT}\right)^2 1_{Z+d_2\geq 0}\right], \\ &= \int_{-\infty}^{d_2}\Bigg[\Bigg(x^2\exp(-2\sigma^2T + 2\sigma\sqrt{T}z) + C^2e^{-2rT} \\ &\qquad - 2xC\exp(-\sigma^2T/2 + \sigma\sqrt{T}z - rT)\Bigg)\frac{e^{-z^2/2}}{\sqrt{2\pi}}\Bigg]dy.\end{aligned}$$

Using the same arguments as the above formula up to a change of variable, we can write

$$\mathbb{E}\left[f(Z,x)^2\right] = x^2e^{\sigma^2T}\phi(d_3) - 2xCe^{-rT}\phi(d_1) + C^2e^{-2rT}\phi(d_2),$$

where $d_3 = d_2 + 2\sigma\sqrt{T}$. Hence, the variance of the cash-flows can be given explicitly, which will be used to quantify the variance underlying the Monte-Carlo simulations as follows

$$\operatorname{Var}\left(\frac{1}{M}\sum_{i=1}^{M} f(Z^i, x)\right) = \frac{1}{M}\operatorname{Var}(f(Z, x)).$$

A.2 Variance reduction with antithetic variables technique

The antithetic variable techinque estimates the expected value of $f(Z, x)$ by

$$\frac{2}{M}\sum_{i=1}^{M/2} \frac{f(Z^i, x) - f(-Z^i, x)}{2}.$$

The variance of this estimator is given by:

$$\begin{aligned}
&\operatorname{Var}\left(\frac{2}{M}\sum_{i=1}^{M/2} \frac{f(Z^i, x) - f(-Z^i, x)}{2}\right) \\
&= \frac{2}{M}\operatorname{Var}\left(\frac{f(Z^i, x) - f(-Z^i, x)}{2}\right) \\
&= \frac{1}{M}\Big(\operatorname{Var}(f(Z, x)) + \operatorname{Cov}(f(Z, x) - f(-Z, x))\Big)
\end{aligned}$$

where

$$\operatorname{Cov}(f(Z, x), f(-Z, x)) = \mathbb{E}[f(Z, x) f(-Z, x)] - \mathbb{E}[f(Z, x)]^2,$$

and

$$\begin{aligned}
&\mathbb{E}[f(Z, x) f(-Z, x)] \\
&\quad = \mathbb{E}\Big[e^{-rT}\Big(xe^{(r-\sigma^2/2)T + \sigma\sqrt{T}Z} - C\Big)^{+} e^{-rT} \\
&\qquad \Big(xe^{((r-\sigma^2/2)T - \sigma\sqrt{T}Z)} - C\Big)^{+}\Big] \\
&\quad = \Big((x^2 e^{-\sigma^2 T} + C^2 e^{-2rT})(\phi(d_2) \\
&\qquad - \phi(-d_2)) - 2xCe^{-rT}(\phi(d_1) - \phi(d_4)\Big)1_{d_2>0}
\end{aligned}$$

with $d_4 = -d_2 + \sigma\sqrt{T}$.

A.3 Analytical variance of the fixed strike lookback call option

Consider a fixed strike Lookback Call Option that starts at $t = 0$. Define $X = \max(C, x)$ then

$$\begin{aligned}
\mathbb{E}[f(W_t, x)] &= \mathbb{E}\left[e^{-rT}\left(\max_{0\le t\le T}(xe^{(r-\sigma^2/2)t + \sigma W_t}) - C\right)^{+}\right] \\
&= e^{-rT}(X - C) + x\phi(d_1) - Xe^{-rT}\phi(d_2)
\end{aligned}$$

$$+xe^{-rT}\frac{\sigma^2}{2r}\left(-\left(\frac{x}{X}\right)^{\frac{-2r}{\sigma^2}}\phi\left(d_1-\frac{2r}{\sigma}\sqrt{T}\right)+e^{rT}\phi(d_1)\right)$$

$$\begin{aligned}\mathbb{E}[f(W_t,x)^2] &= e^{-2rT}(X-C)^2-e^{-rT}xC\left(2+\frac{\sigma^2}{r}\right)\phi(d_1)+e^{-2rT}X(2C-X)\phi(d_2)\\ &+e^{-2rT}x\left(X\frac{\sigma^2}{2r+\sigma^2}-(X-C)\right)\frac{\sigma^2}{r}\left(\frac{x}{X}\right)^{\frac{-2r}{\sigma^2}}\phi\left(d_1-\frac{2r\sqrt{T}}{\sigma}\right)\\ &+x^2e^{\sigma^2T}\frac{2r+3\sigma^2}{2r+\sigma^2}\phi\left(d_1+\sigma\sqrt{T}\right)\end{aligned}$$

References

Aloise, D., Deshpande, A., Hansen, P., & Popat, P. (2009). Np-hardness of Euclidean sum-of-squares clustering. *Machine Learning, 75*(2), 245–248.

Bally, V., & Pages, G. (2003). Error analysis of the optimal quantization algorithm for obstacle problems. *Stochastic Processes and Their Applications, 106*(1), 1–40.

Bauer, D., Bergmann, D., & Kiesel, R. (2010). On the risk-neutral valuation of life insurance contracts with numerical methods in view. *ASTIN Bulletin: The Journal of the IAA, 40*(1), 65–95.

Bauer, D., Bergmann, D., & Reuss, A. (2010b). Solvency II and nested simulations: A least-squares Monte Carlo approach. In *Proceedings of the 2010 ICA congress*.

Bignon, A., Ndjeng-Ndjeng, A., Salhi, Y., & Thérond, P.-E. (2019). A reduced-form model for a life insurance's net asset value. *Bankers, Markets & Investors, 157,* 1–13.

Borel-Mathurin, F. & Vedani, J. (2019). Market-consistent valuation: A step towards calculation stability.

Cathcart, M. & Morrison, S. (2009). Variable annuity economic capital: The least-squares Monte Carlo approach. *Life & Pensions*, pp. 36–40.

Chauvigny, M., Devineau, L., Loisel, S., & Maume-Deschamps, V. (2011). Fast remote but not extreme quantiles with multiple factors: Applications to solvency ii and enterprise risk management. *European Actuarial Journal, 1*(1), 131–157.

Conze, A. (1991). Path dependent options: The case of lookback options. *The Journal of Finance, 46*(5), 1893–1907.

Davidson, I. & Ravi, S. (2005). Clustering with constraints: Feasibility issues and the k-means algorithm. In *Proceedings of the 2005 SIAM international conference on data mining*, pp. 138–149. SIAM.

Demiriz, A., Bennett, K. P., & Embrechts, M. J. (1999). Semi-supervised clustering using genetic algorithms. *Artificial Neural Networks in Engineering*, pp. 809–814.

Durrett, R. & Durrett, R. (1984). *Brownian motion and martingales in analysis*. Wadsworth Advanced Books & Software California.

Felix, J.-P. (2016). Cash flow projection models. In *Modelling in life insurance: A management perspective*, pp. 63–79. Springer.

Floryszczak, A., Le Courtois, O., & Majri, M. (2016). Inside the Solvency 2 black box: Net asset values and solvency capital requirements with a least-squares monte-carlo approach. *Insurance: Mathematics and Economics, 71*, 15–26.

Glynn, P. W. (1994). Efficiency improvement techniques. *Annals of Operations Research, 53*(1), 175–197.

Gray, R. (1984). Vector quantization. *IEEE Assp Magazine, 1*(2), 4–29.

Hainaut, D., Devolder, P., & Pelsser, A. (2018). Robust evaluation of SCR for participating life insurances under Solvency II. *Insurance: Mathematics and Economics, 79*, 107–123.

Hilli, P., Koivu, M., Pennanen, T., & Ranne, A. (2007). A stochastic programming model for asset liability management of a finnish pension company. *Annals of Operations Research, 152*(1), 115–139.

Liu, Y., Pagès, G., et al. (2020). Characterization of probability distribution convergence in wasserstein distance by l^p-quantization error function. *Bernoulli, 26*(2), 1171–1204.

MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, pp. 281–297. Oakland, CA, USA.

Pollard, D. (1982). Quantization and the method of k-means. *IEEE Transactions on Information Theory, 28*(2), 199–205.

Sculley, D. (2010). Web-scale k-means clustering. In *Proceedings of the 19th international conference on World wide web*, pp. 1177–1178.

Thérond, P.-E. (2016). About market consistent valuation in insurance. In *Modelling in life insurance: A management perspective*, pp. 43–60. Springer.

Vedani, J., El Karoui, N., Loisel, S., & Prigent, J.-L. (2017). Market inconsistencies of market-consistent European life insurance economic valuations: Pitfalls and practical solutions. *European Actuarial Journal, 7*(1), 1–28.

Villani, C. (2008). *Optimal transport: Old and new*, vol. 338. Springer

Zenios, S. A. (1995). Asset/liability management under uncertainty for fixed-income securities. *Annals of Operations Research, 59*(1), 77–97.

Zhang, T., Ramakrishnan, R., & Livny, M. (1996). Birch: An efficient data clustering method for very large databases. *ACM SIGMOD Record, 25*(2), 103–114.