

Faculté des sciences

Least-Squares Monte Carlo Method and Machine Learning for Solvency Capital Requirement Calculation.

Auteur·es : Prince Agogue De Tetio

Promoteur·rices : Jérôme Barbarin

Lecteur·rices : Donation Hainaut

Année académique 2025-2026

LOUVAIN SCHOOL OF STATISTICS, BIOSTATISTICS AND
ACTUARIAL SCIENCES

Master en Sciences Actuarielles, à finalité spécialisée

Least-Squares Monte Carlo Method and Machine Learning for Solvency Capital Requirement Calculation

Written by:
Prince Agogue De Tetio

Under the supervision of:
Jérôme Barbarin

Academic Year : 2025-2026

Abstract

The Solvency II directives require of an insurance company to hold sufficient capital to fulfill its underwritings in a one in two hundred years adverse change in their own funds, called the Solvency Capital Requirements(SCR). When an internal model is used, obtaining this capital implies the valuation of the balance sheet of the insurance company in every possible future state of the world leading to a nested simulation problem whose computational cost can quickly become impractical for large portfolios with guarantees, profit sharing mechanisms and options. Least-Squares Monte Carlo (LSMC) the classical method addresses this by fitting a regression function to a number of rough valuations and then evaluating it over many scenarios. The regression function is usually polynomial and with a large set of predictors the regression coefficients increase rapidly and its accuracy especially in the stressed capital region is not guaranteed.

This thesis asks whether the regression step can be improved using machine learning models that stay transparent enough and to answer this, a stylized Universal Life portfolio is projected through a seven-driver ALM model that includes two Gaussian rate factors, stochastic equity volatility, a persistent credit spread, dynamic surrenders and Gompertz–Makeham mortality with a stochastic index, fitted with the same data and judged against the same nested reference on the three proxies : polynomial LSMC as the benchmark, an Explainable Boosting Machine(EBM) and LocalGLMnet. Each proxy is tested twice, once on the full best estimate liability and once on the smaller future discretionary benefits component and is measured not only by average accuracy but also by bias, capital-region error and stability across repeated fits.

The comparison produces no outstanding winner as LocalGLMnet fits best overall, the polynomial benchmark lands closest to the reference capital in the final sample and the EBM gives the steadiest capital estimate while also explaining its own predictions through readable shape functions. Because the SCR rests on only a few of extreme scenarios, the main lesson is that a single accuracy score is a poor way to choose a solvency proxy and that flexibility can be gained without losing interpretability.

Acknowledgment

First I would like to express my deepest gratitude to my supervisor, Professor Jérôme Barbarin, for his continuous support and invaluable guidance throughout the preparation of this thesis. His expertise, insightful remarks have been essential to successfully complete this thesis. I am truly grateful for the time and dedication he put into my research.

I would also like to thank the professors and staff of the Institute of Statistics, Biostatistics, and Actuarial Sciences at UCLouvain for providing the best academic environment and a high-quality education. Their recognized commitment to teaching and research has significantly contributed to my personal intellectual and professional development.

My sincere thanks also go to my friends and colleagues, whose support, motivation and fellowship made this journey both enriching and enjoyable.

Above all, I am Immensely grateful to my family for their unconditional love, endless encouragement and unwavering faith in me even when I doubted. A special tribute to my parents: Tetio Samuel and Sidjie Georgette, whose sacrifices, guidance and constant support have shaped who I am today. Their belief in my abilities has been my greatest source of strength and this achievement is as much theirs as it is mine.

Contents

1	Introduction	1
2	Solvency II, Solvency Capital Requirement and the Nested Valuation Problem	3
2.1	Introduction	3
2.2	Solvency Capital Requirement under Solvency II	3
2.2.1	Net asset value and own funds	3
2.2.2	The one-year 99.5% Value-at-Risk criterion	4
2.3	SCR calculation approaches under Solvency II	4
2.3.1	The standard formula	4
2.3.2	The internal model approach	5
2.4	Risk drivers and balance-sheet sensitivities	5
2.4.1	Financial and actuarial risk drivers	5
2.4.2	Stresses and their role in solvency analysis	5
2.5	From current valuation to one-year solvency assessment	6
2.5.1	Valuation at time $t=0$	6
2.5.2	Valuation at time $t=1$ and the distribution of future NAV	6
2.6	Scenario structure for internal model SCR	6
2.6.1	Real-world and risk-neutral measures	6
2.6.2	Outer and inner scenarios	7
2.7	Nested simulations and Computational burden	7
2.8	Motivation for proxy modeling	7
2.9	Conclusion	8
3	Least-Squares Monte Carlo (LSMC)	9
3.1	Motivation and Principle	9
3.2	Mathematical Formulation	10
3.3	LSMC Algorithm	11
3.3.1	Choice of risk drivers	11
3.3.2	Construction of fitting scenarios	12
3.3.3	Regression calibration	13
3.3.4	Validation	13
3.3.5	5. Aggregation and capital extraction	14
3.4	Regression Specification and Empirical Evidence from the Literature	15
3.4.1	Classical polynomial basis functions	15
3.4.2	Why interaction terms matter	16
3.4.3	Evidence from realistic insurance portfolios	16
3.4.4	Adaptive selection and parsimonious specification	16
3.5	Limitations of Classical LSMC in Practice	17
3.5.1	Explosion of candidate terms	17
3.5.2	Sensitivity to basis specification	17
3.5.3	Extrapolation and the capital region	17
3.5.4	Implication for the thesis	17
3.6	Conclusion	18

4	Explainable Machine Learning Proxy Models	19
4.1	Motivation: From Polynomial LSMC to Interpretable Machine Learning	19
4.2	Feed-Forward Neural Networks for Tabular Proxy Regression	20
4.2.1	Mathematical setup	20
4.2.2	Why feed-forward networks are attractive	21
4.2.3	Why they remain insufficient for the present thesis	21
4.3	LocalGLMnet	21
4.3.1	General idea	21
4.3.2	Model structure	22
4.3.3	Why LocalGLMnet is attractive for LSMC proxy modelling	22
4.3.4	Interpretability and variable selection	23
4.4	Explainable Boosting Machines	23
4.4.1	General idea	23
4.4.2	Cyclic boosting and shape functions	24
4.4.3	Bagging and interaction selection in InterpretML	24
4.4.4	Why EBM is relevant for this thesis	25
4.4.5	Strengths and caveats	26
4.5	Positioning the Models within the LSMC Framework	26
5	Numerical Implementation and Empirical Results	28
5.1	Introduction	28
5.2	Universal Life Portfolio and ALM Projection Model	28
5.2.1	Portfolio representation	28
5.2.2	Asset portfolio and economic dynamics	29
5.2.3	Real-world and valuation dynamics	33
5.2.4	Liability cash flows, guarantees and profit sharing	34
5.2.5	High-dimensional risk-driver vector	35
5.2.6	Parameter specification	36
5.3	Nested Simulation and Validation Design	37
5.3.1	One-year valuation and SCR definition	37
5.3.2	Present value of an inner path	37
5.3.3	Fitting scenarios	38
5.3.4	Validation and capital calculation	39
5.3.5	Numerical budget and BEL_0 convergence	39
5.3.6	Performance measures	41
5.4	Implementation of the Proxy Models	42
5.4.1	Polynomial LSMC benchmark	42
5.4.2	Explainable Boosting Machine	42
5.4.3	LocalGLMnet	42
5.4.4	Hyperparameter selection	43
5.5	Main Experiment: Full Time-1 Best Estimate Liability	43
5.5.1	Global validation accuracy	43
5.5.2	One-dimensional validation sweeps	44
5.5.3	Adverse and capital-region validation	45
5.5.4	SCR accuracy	45
5.6	Robustness and Sensitivity Analysis	46

5.6.1	EBM interactions and bagging	46
5.6.2	Sensitivity to the number of inner simulations	46
5.6.3	Sensitivity to the number of fitting scenarios	47
5.6.4	Stability across random repetitions	48
5.6.5	Economic parameter sensitivity	49
5.7	Extension: Future Discretionary Benefits	50
5.7.1	Actuarial motivation	50
5.7.2	Construction of GB_1 and FDB_1	50
5.7.3	Global and capital-region results	51
5.7.4	Comparison of the BEL_1 and FDB_1 proxy targets	52
5.8	Interpretability and Actuarial Validation	53
5.8.1	EBM main effects for the full liability	53
5.8.2	Term importance and selected interactions	54
5.8.3	Interpretability of the FDB target	55
5.8.4	Interpretability as a model-risk control	57
5.9	Discussion and Limitations	57
5.10	Conclusion	58
6	Conclusion	60
A	Appendix	64
A.1	Numerical parameterisation	64
A.1.1	Purpose of the parameterisation	64
A.1.2	Profit-sharing moneyness and initial solvency	64
A.2	Random seeds	65
A.3	Proxy Hyperparameter Tuning	65
A.3.1	Tuning protocol	65
A.3.2	Polynomial LSMC	65
A.3.3	Explainable Boosting Machine	66
A.3.4	LocalGLMnet	66
A.4	Final selected hyperparameters	68
A.5	Additional Numerical Sensitivity Results	68
A.5.1	BEL_0 convergence	68
A.5.2	Number of inner simulations used for fitting	68
A.5.3	Number of outer fitting scenarios	68
A.5.4	Stability across fitting repetitions	69
A.5.5	Economic parameter sensitivity	69

1 Introduction

The Solvency Capital Requirement (SCR) under Solvency II is intended to ensure that an insurer holds sufficient capital to absorb severe losses over a one-year horizon. For a life insurer using an internal model, its calculation can become computationally demanding. A large number of possible economic and actuarial situations must be generated at the end of the first year, and the remaining liabilities must be valued in each of these situations. When this valuation itself requires Monte Carlo simulation, the calculation takes the form of a nested simulation problem [1, 2].

This problem is particularly relevant for long-term life insurance contracts. Their value may depend not only on interest rates and investment returns, but also on guarantees, future profit sharing, surrender behaviour and mortality. Performing an accurate market-consistent valuation for every one-year scenario may therefore require a very large number of simulations.

Least-Squares Monte Carlo (LSMC) offers a way to reduce this computational cost. Instead of carrying out a large inner simulation for every scenario, the liability value is approximated by a regression proxy depending on a smaller set of risk drivers. Once estimated, the proxy can be evaluated very quickly over a large number of real-world scenarios. Polynomial regression is a common choice for this purpose and provides a transparent and well-established benchmark [2, 3].

However, the relationship between the risk drivers and the liability value can be strongly nonlinear. Profit-sharing mechanisms, policyholder behaviour and interactions between financial and actuarial variables can make the choice of a suitable polynomial basis increasingly difficult as the number of risk factors grows. More flexible machine learning methods may help to capture these relationships, but flexibility alone is not sufficient in an actuarial context. Since the resulting proxy is used for capital calculation, it should also remain sufficiently understandable and capable of being validated.

This thesis therefore studies two interpretable machine learning approaches as alternatives to classical polynomial LSMC: Explainable Boosting Machine (EBM) and LocalGLMnet. EBM represents the prediction through learned one-dimensional effects and selected pairwise interactions [4], while LocalGLMnet combines a generalized linear model structure with coefficients that may vary with the risk factors [5]. Both approaches aim to introduce additional flexibility without completely losing the link between the prediction and the underlying risk factors.

The main objective of the thesis is to compare these three proxy approaches for the estimation of a life-insurance SCR. More specifically, the analysis addresses three questions. First, can EBM and LocalGLMnet improve the approximation of a complex liability compared with polynomial LSMC? Secondly does a model that is more accurate on average also give a more accurate SCR? And lastly, does the choice of the quantity being approximated influence the quality of the final capital estimate?

The numerical study is based on a stylised Universal Life portfolio. The asset-liability model contains seven financial and actuarial risk drivers, covering interest rates, portfolio returns, equity volatility, credit spreads, surrender behaviour and mortality. The three proxies are trained and validated on common simulation

data and are compared against the same nested Monte Carlo reference. Their performance is evaluated not only through global prediction errors, but also in adverse scenarios and in the region surrounding the 99.5 % capital quantile. Stability across repeated fittings and changes in selected liability assumptions is also considered.

Two proxy targets are studied. The first is the complete best estimate liability at the end of the one-year horizon. The second is the value of future discretionary benefits, which is approximated separately before the full liability is reconstructed. This second experiment allows the analysis to examine whether the way in which the liability is decomposed can affect the final SCR approximation.

The remainder of the thesis is organised as follows. Chapter 1 introduces the Solvency II framework and the nested simulation problem associated with internal-model SCR calculation. Chapter 2 presents the LSMC methodology and the construction and validation of regression proxies. Chapter 3 introduces LocalGLMnet and Explainable Boosting Machines and discusses their use as interpretable proxy models. Chapter 4 presents the Universal Life model, the numerical implementation and the empirical comparison of the three approaches. Additional numerical and implementation details are provided in the Appendix.

2 Solvency II, Solvency Capital Requirement and the Nested Valuation Problem

2.1 Introduction

Solvency II is the prudential framework governing insurance and reinsurance undertakings in the European Union [6]. Its main objective is to ensure that insurers hold enough eligible own funds to absorb adverse developments and continue meeting their obligations to policyholders. In contrast to older rule-based solvency regimes, Solvency II adopts a prospective and risk-sensitive approach based on an economic valuation of assets, liabilities and risk exposures.

Within this framework, the Solvency Capital Requirement (SCR) plays a central role. It represents the level of capital that an insurer must hold in order to remain solvent over a one-year horizon with a high degree of confidence. For undertakings using an internal model, the SCR is derived from the stochastic distribution of future own funds, or equivalently from the future net asset value. This naturally requires a joint analysis of assets and liabilities, which is particularly important in life insurance because of long maturities, embedded guarantees, discretionary management rules and path-dependent cash flows [7].

From a computational perspective, the direct evaluation of the SCR under an internal model leads to a nested valuation problem. One first has to generate possible one-year states of the world under the real-world measure and then, conditional on each of these states, revalue the balance sheet using market-consistent principles. This two-level structure explains why brute-force Monte Carlo methods quickly become computationally prohibitive and why proxy techniques, such as Least-Squares Monte Carlo, are needed in practice [1, 2].

This chapter introduces the regulatory and quantitative foundations that motivate the modeling approaches developed in the rest of the thesis. It first defines the SCR and its interpretation under Solvency II. It then briefly presents the two main calculation routes, namely the standard formula and the internal model approach. After that, it introduces the notions of risk drivers, stresses and scenario structures needed to formulate the solvency problem in a dynamic asset-liability setting. Finally, it explains how the transition from current valuation at time $t = 0$ to solvency assessment at time $t = 1$ gives rise to nested simulations and motivates the use of proxy models.

2.2 Solvency Capital Requirement under Solvency II

2.2.1 Net asset value and own funds

Let $V(t, A)$ denote the market-consistent value of assets at time t , and let $V(t, L)$ denote the market-consistent value of liabilities at the same date. The net asset value is then defined by

$$NAV_t = V(t, A) - V(t, L).$$

This quantity measures the residual economic value available after liabilities have been covered and therefore provides a natural basis for solvency assessment. In the

context of internal models, the future distribution of NAV_t , and in particular of NAV_1 , is the key object from which the capital requirement is derived [7].

The importance of the net asset value follows directly from the economic balance-sheet logic of Solvency II. Solvency is not assessed solely through book values or simplified accounting indicators, but through a market-consistent comparison between the value of assets and the value of obligations. This is why asset-liability modeling is central in life insurance applications [2].

2.2.2 The one-year 99.5% Value-at-Risk criterion

Under Solvency II, the SCR is calibrated so as to reflect extreme but plausible adverse developments over a one-year horizon. In the internal model framework, it is linked to the 99.5% Value-at-Risk of the future net asset value, or equivalently of the associated loss distribution. In practical terms, this means that the insurer should remain solvent over one year with a probability of at least 99.5% [1, 7].

A common formalization is

$$SCR = v(0, 1) \left(\mathbb{E}^{\mathbb{P}}[NAV_1] - \text{VaR}_{0.995}(NAV_1) \right),$$

where $v(0, 1)$ denotes the one-year risk-free discount factor, $\mathbb{P}$ the real-world probability measure, and $\text{VaR}_{0.995}(NAV_1)$ the 0.5th percentile of the future net asset value distribution under $\mathbb{P}$. Here we adopt the convention that

$$\text{VaR}_{\alpha}(X) := \inf \{x \in \mathbb{R} : \mathbb{P}(X \leq x) \geq 1 - \alpha\},$$

so that $\text{VaR}_{0.995}(NAV_1)$ refers to the *lower* tail of the NAV distribution, consistent with a loss-based interpretation. This expression highlights that the SCR represents the present value of the shortfall between the expected future economic own funds and their worst-case realization at the 1-in-200 confidence level.

In this thesis, the SCR is not studied as a purely regulatory output. Rather, it is viewed as the result of a quantitative asset-liability modeling exercise. Its accuracy therefore depends not only on the specification of the ALM model itself, but also on the numerical method used to approximate future liability values and future net asset values.

2.3 SCR calculation approaches under Solvency II

2.3.1 The standard formula

Solvency II provides a standardized route for the calculation of the SCR through the standard formula. This approach relies on a modular decomposition of risks and on prescribed aggregation rules, usually based on regulatory correlations. Its main advantage is its relative simplicity and operational ease [2].

However, the standard formula is not designed to capture in a granular way the specific interactions between assets and liabilities, dynamic management rules, or the path-dependent features of many life insurance liabilities. For undertakings with complex ALM structures, it may therefore provide only an approximate representation of the actual risk profile. For this reason, the standard formula is not the main focus of the present thesis, but it remains an important benchmark within the Solvency II framework.

2.3.2 The internal model approach

As an alternative to the standard formula, Solvency II allows insurers to use an internal model, subject to supervisory approval. The purpose of the internal model is to assess the undertaking's risks in a way that better reflects the actual structure of its business and its overall solvency needs [7].

This is especially relevant in life insurance and ALM applications. Long-dated contracts, participation mechanisms, surrender behavior, profit-sharing rules and embedded guarantees all create liability cash flows whose value depends on a large number of interacting financial and actuarial factors. In such a setting, a realistic solvency assessment requires a stochastic framework that can jointly represent the behavior of assets and liabilities [1, 7].

Moreover, the internal model is not intended to be a purely regulatory tool. Solvency II requires that it be meaningfully integrated into the insurer's decision-making and risk-management process, which is usually referred to as the use test [6]. This is consistent with the objective of the present thesis, which is to study and compare proxy-based approaches for SCR assessment within an ALM framework.

2.4 Risk drivers and balance-sheet sensitivities

2.4.1 Financial and actuarial risk drivers

The solvency position of an insurer is affected by a wide range of uncertainties. In practice, these uncertainties are represented through a set of risk drivers, or risk factors, that summarize the main variables influencing the value of assets, liabilities and net asset value. In life insurance, financial risk drivers typically include interest rates, equity levels and credit variables, while actuarial risk drivers include mortality, longevity, lapse and contract options like profit-sharing.

Introducing risk drivers is important because, first it makes it possible to describe the evolution of the balance sheet in a structured and economically interpretable way and secondly, it provides the explanatory variables on which proxy models can later be built on. In the present thesis, the approximation methods work on a simplified summary of the system, made of carefully chosen risk factors that matter for solvency.

2.4.2 Stresses and their role in solvency analysis

Once the relevant risk drivers have been identified, solvency analysis requires the study of how adverse changes in these variables affect the balance sheet. These changes are usually referred to as stresses or shocks. A stress may be understood as a deterioration in one or several financial or actuarial conditions, leading to a reduction in future net asset value [2].

In an internal-model framework, stresses are not restricted to deterministic regulatory shocks. They are embedded in the stochastic evolution of the risk drivers and appear as realized deviations from their baseline trajectories. This is particularly important in ALM modeling, where the effect of a given stress depends not only on one isolated variable, but also on the joint behavior of the full asset-liability structure.

2.5 From current valuation to one-year solvency assessment

2.5.1 Valuation at time $t=0$

At time $t = 0$, the solvency position of the insurer is obtained by valuing the current balance sheet under prevailing market conditions. This consists in computing the market-consistent value of assets and liabilities and deriving the corresponding value of NAV_0 . Although this valuation may already be complex in life insurance, it remains a single-date pricing exercise.

2.5.2 Valuation at time $t=1$ and the distribution of future NAV

The SCR, however, cannot be obtained from the current value NAV_0 alone. It depends on the distribution of the change in economic own funds over the one-year risk horizon and therefore on the full probability distribution of NAV_1 . In the loss convention introduced above, NAV_0 provides the risk-free accumulated reference level, while the stochastic variation comes from the future balance sheet at time 1. Instead of valuing a single balance sheet under observed market conditions, one must therefore determine future balance-sheet values under a wide range of possible financial and actuarial scenarios [1, 2].

The main difficulty comes from the fact that the liabilities at time $t = 1$ must themselves be valued conditionally on the information available at that future date. In other words, the insurer must first project the relevant risk drivers over one year under the real-world measure $\mathbb{P}$ and then, for each resulting future state of the world, compute a new market-consistent valuation of the remaining liability cash flows under the risk-neutral measure $\mathbb{Q}$ [7].

A useful summary is the following: while valuation at time $t = 0$ is a single market-consistent pricing exercise, the SCR requires the full probability distribution of the net asset value at time $t = 1$. This is what makes the problem much more demanding and naturally leads to nested simulations.

2.6 Scenario structure for internal model SCR

2.6.1 Real-world and risk-neutral measures

A central feature of internal model SCR valuation is the coexistence of two probabilistic perspectives. On the one hand, the future solvency position must be assessed under the real-world measure, since the SCR is based on the actual probability distribution of future balance-sheet outcomes over one year. On the other hand, the valuation of liabilities at future dates must remain market-consistent, which requires the use of a risk-neutral pricing framework [2, 7].

This distinction is essential. The real-world measure governs the stochastic evolution of risk drivers over the solvency horizon, while the risk-neutral measure is used conditionally at time $t = 1$ to price the remaining cash flows consistently with financial market valuation principles.

2.6.2 Outer and inner scenarios

This dual role of projection and valuation naturally gives rise to two types of scenarios. The first type consists of one-year scenarios generated under the real-world measure. These are the *outer scenarios*. They describe the possible economic and actuarial states of the world at time $t = 1$ and represent the future distribution of the insurer's balance sheet.

The second type consists of conditional valuation scenarios generated under the risk-neutral measure from each outer state. These are the *inner scenarios*. Their purpose is to compute the market-consistent value of future asset and liability cash flows once the outer scenario has materialized. In life insurance, inner scenarios are particularly important because embedded guarantees and options generally prevent closed-form valuation [1].

2.7 Nested simulations and Computational burden

The direct computational response to the SCR problem is the nested simulation approach. Its first layer consists of generating a large number of one-year scenarios for the risk drivers under the real-world probability measure. These outer scenarios describe the future distribution of the insurer's solvency position. Conditional on each outer scenario, the insurer must then perform a second-stage valuation of liabilities under market-consistent principles. This requires simulations under the risk-neutral measure and gives rise to the inner scenarios. The liability value attached to a given outer scenario is then obtained by averaging the discounted future cash flows across the corresponding inner simulations.

Although conceptually straightforward, the nested simulation approach is computationally very expensive in realistic applications [1, 2]. A robust SCR calculation may require thousands of outer scenarios and, for each of them, a large number of inner simulations. The total number of valuations can therefore reach extremely high levels, which makes brute-force implementation impractical for realistic life insurance ALM models.

This computational difficulty is the main reason why approximation methods have become necessary in practice. Rather than fully revaluing liabilities in every outer scenario, one seeks to replace the valuation map by a sufficiently accurate proxy. This naturally motivates the use of Least-Squares Monte Carlo and, or better still, the proxy-modeling approaches studied in this thesis.

2.8 Motivation for proxy modeling

The previous discussion shows that internal model SCR calculation involves a trade-off between conceptual rigor and computational feasibility. On the one hand, a proper solvency assessment requires the future distribution of net asset values and therefore a nested valuation framework. On the other hand, the direct implementation of such a framework is too costly in realistic insurance applications [1].

Among the available approximation methods, Least-Squares Monte Carlo [8] has emerged as one of the most widely used approach. Its main idea is to perform a limited number of full valuations and then use them to calibrate an approximation of

the liability value or the net asset value as a function of the underlying risk drivers. Once such a proxy is available, a large number of outer scenarios can be evaluated at a much lower computational cost.

The present thesis builds on this idea and studies several proxy-modeling approaches within an ALM setting for SCR evaluation. Least-Squares Monte Carlo provides the natural starting point, while more recent interpretable machine-learning extensions, namely LocalGLMnet and Explainable Boosting Machines, are introduced in the following chapters.

2.9 Conclusion

This section has established the main regulatory and quantitative concepts underlying the thesis. It has shown that the SCR under Solvency II is a one-year risk measure derived from the future distribution of the insurer's net asset value, and that the internal model approach is particularly relevant for life insurance and ALM applications. It has also explained why the transition from current valuation at time $t = 0$ to solvency assessment at time $t = 1$ naturally leads to a nested simulation problem involving outer scenarios under the real-world measure and inner scenarios under the risk-neutral measure. Finally, it has highlighted the computational limits of brute-force nested simulations and thereby motivated the use of proxy-based methods, which form the core methodological focus of the remainder of the thesis.

3 Least-Squares Monte Carlo (LSMC)

3.1 Motivation and Principle

As discussed in Chapter 1, the Solvency Capital Requirement (SCR) under Solvency II is derived from the one-year distribution of losses or, equivalently, from the adverse evolution of own funds over a one-year horizon [6]. In an internal model framework, this requires the insurer to project the relevant risk factors over one year under the real-world measure and, conditional on each resulting scenario, to revalue assets and liabilities on a market-consistent basis. For life insurance portfolios, such a valuation is rarely available in closed form because liabilities are long-dated, path-dependent, and frequently include guarantees, policyholder options and management rules. In practice, this naturally leads to a nested simulation problem.

The direct nested Monte Carlo approach is conceptually straightforward. One first generates a large number of outer scenarios under the real-world measure in order to describe the possible one-year states of the world. Then, for each of these outer scenarios, one performs inner simulations under the pricing or risk-neutral measure to obtain market-consistent values of the balance-sheet components at time 1. The resulting one-year distribution of net asset value (NAV) or available capital can then be used to estimate the SCR. However, the literature is very clear that this brute-force approach is computationally prohibitive in realistic insurance applications. Koursaris [3] notes that the total scenario requirement can easily reach millions or even hundreds of millions once outer and inner simulations are combined. Floryszczaka et al. 2015 make the same point at company level [9], emphasizing that nested simulation is theoretically the most general approach but operationally unrealistic because of the required computation resources.

This is precisely the context in which the Least-Squares Monte Carlo method becomes useful. The basic idea of LSMC is to replace repeated inner valuations by a proxy function calibrated on a comparatively small but carefully designed set of fitting scenarios. Instead of computing a full inner valuation under every outer scenario, one evaluates the expensive liability model only on selected points and then uses regression to approximate the conditional relationship between the risk drivers and the target quantity. Once the proxy has been calibrated and validated, it can be evaluated very quickly on a large set of outer scenarios to recover the full loss distribution and the associated SCR. LSMC is not merely a computational shortcut, it is a full approximation framework designed to make solvency calculations feasible without abandoning the economic logic of nested valuation. Krah et al. 2020 [10] summarize this well by describing LSMC as a method that uses a comparatively small number of nested Monte Carlo simulations and translates their outputs into a proxy function of the insurer's loss with respect to the underlying risk factors.

The relevance of LSMC for this thesis is therefore twofold. First, it provides the standard benchmark framework for replacing full nested simulations in Solvency II applications. Second, it makes explicit that the quality of the final SCR estimate depends not only on the simulation model but also on the quality of the regression-based proxy. This second aspect is particularly important for the present work, because the later chapters focus precisely on improving the regression step while preserving interpretability.

3.2 Mathematical Formulation

Let $(\Omega, \mathcal{F}, \mathbb{P})$ denote the real-world probability space governing the one-year evolution of the relevant financial and actuarial risk factors. Let $X \in \mathbb{R}^d$ be the vector of state variables observed at time 1, for example changes in interest rates, equity, credit, mortality, morbidity, lapse and expense levels over the one-year period. Conditional on X , the market-consistent value of future liabilities may be represented as

$$Y_1(X) = \mathbb{E}^{\mathbb{Q}} \left[\sum_{t>1} v(1, t) CF_t \middle| X \right],$$

where CF_t denotes the future liability cash flows and $v(1, t)$ the appropriate risk-neutral discount factor. In this formulation, $\mathbb{P}$ governs the outer one-year evolution of the state variables, whereas $\mathbb{Q}$ is used for the inner market-consistent valuation. This distinction is essential in solvency modelling: the outer distribution reflects the insurer's one-year risk profile, while the inner valuation reflects market-consistent values.

If $V_1(A)$ denotes the market value of assets at time 1, then the net asset value at that horizon is

$$NAV_1 = V_1(A) - Y_1(X).$$

The SCR is then derived from the one-year distribution of this quantity through the loss convention introduced in Chapter 1. Defining

$$L_1 = NAV_0(1 + r_0) - NAV_1,$$

the capital requirement used in the numerical framework is

$$SCR_1 = \text{VaR}_{0.995}^{\mathbb{P}}(L_1).$$

The corresponding amount expressed at time 0 is $v(0, 1)SCR_1$. This formulation is consistent with simulation-based LSMC approaches in which the SCR is obtained from the empirical distribution of the adverse change in available capital [1, 2].

Alternative normalisations of the future NAV distribution are also used in the literature, including formulations centred around its expected value [7]. The numerical analysis in this thesis uses the loss-based convention above throughout. The essential structure of the LSMC problem remains unchanged: the liability target is a conditional expectation under $\mathbb{Q}$, whereas the one-year loss distribution from which capital is extracted is generated under $\mathbb{P}$.

The computational constraint arises because $Y_1(X)$ is generally not known analytically and must be approximated numerically for each outer realization of X . If one uses N outer scenarios and M inner simulations for each outer scenario, then the total number of required liability valuations is of order $N \times M$. In realistic life insurance settings, both numbers must be large: N must be sufficient to identify the tail of the loss distribution, while M must be sufficient to stabilize the inner valuation. This is exactly why nested simulations become infeasible in practice and why proxy methods such as LSMC are introduced. Krah et al. [10] explicitly note that life insurers may need several hundred thousand to several million cash-flow projection

runs under a direct nested approach and that current computational capacities are often not sufficient for such tasks.

Within the LSMC framework, the objective is therefore not to evaluate $Y_1(X)$ exactly for all possible outer scenarios, but to approximate the map

$$X \mapsto \mathbb{E}^Q[Y \mid X]$$

by a proxy function $\hat{Y}(X)$ calibrated on a manageable number of fitting points. The rest of the chapter explains how this approximation is constructed in practice.

3.3 LSMC Algorithm

The LSMC methodology can be described as a sequence of steps linking the simulation model to the final SCR estimate. Although implementations vary across practitioners and research groups, the overall logic remains the same. This subsection goes through each step in detail, because the quality of the final capital estimate depends on decisions made at every stage, not only on the regression step itself.

3.3.1 Choice of risk drivers

The first step is to identify a set of risk drivers $X = (X^{(1)}, \dots, X^{(d)}) \in \mathbb{R}^d$ that captures the main variation of the target quantity over the one-year horizon. In principle, the liability value at time 1 depends on a very large number of financial and actuarial variables. In practice, the modeller must work with a much smaller set of explanatory factors. Koursaris [3] describes this as an exercise in dimensionality reduction: the objective is to find a parsimonious collection of variables that explains the main risk characteristics of the liabilities without making the approximation problem unnecessarily large.

In life insurance and ALM applications, typical candidates include interest-rate factors such as the level, slope and curvature of the yield curve, equity price indices, credit spreads, implied volatility surfaces, lapse rate deviations and mortality stresses. The number of retained drivers varies across implementations, but Krah et al.'s [10] realistic setting with fifteen risk factors gives a sense of the scale that practitioners face.

If important drivers are omitted, the proxy model will be biased, and adding more simulation scenarios will not fix this problem because the missing variables create patterns in the errors that the model simply cannot capture. If too many variables are included that are not very important, the model becomes difficult to estimate and unnecessarily complex. The number of possible combinations of variables increases very quickly, which makes it easier for the model to fit the noise in the data instead of the true relationships (overfitting). Therefore, instead of including everything, the goal is to find a good balance between having enough relevant variables and keeping the model simple and manageable.

There is also a deeper issue. Representing Y as a function of the time 1 values of X assumes that those values form a sufficient summary of everything that affects the liability. In reality, life insurance liabilities are path-dependent: profit sharing, dynamic lapse behaviour, ratchet guarantees and management overlays all depend on

the trajectory of financial variables over the year, not only their endpoint. Collapsing this trajectory into a vector of terminal values is a real modeling assumption with real consequences, and its adequacy depends on the specific portfolio.

3.3.2 Construction of fitting scenarios

Once the risk drivers are fixed, a set of fitting scenarios $\{x_i\}_{i=1}^n$ must be constructed. Each fitting scenario is a realisation of the risk driver vector at the one-year horizon, and for each of these points an inner market-consistent valuation is performed to obtain an estimate of the target quantity. This produces a fitting dataset

$$\{(x_i, \hat{y}_i)\}_{i=1}^n,$$

where $x_i \in \mathbb{R}^d$ is the vector of risk-driver realisations and $\hat{y}_i$ is the corresponding nested Monte Carlo estimate of the liability value.

The construction of this dataset involves two distinct decisions: where to place the fitting points in the risk-driver space, and how many inner simulations to allocate to each point. These decisions interact and both matter.

Scenario design. Several approaches are possible in practice as one may place scenarios along economically meaningful stress directions. What matters most is that the fitting design should cover sufficiently well the region in which the proxy will later be evaluated. Barrie & Hibbert [3] explicitly emphasize that the fitting range should include all likely liability scenarios together with some allowance for more extreme situations, precisely because the proxy may later be used under stressed conditions.

This point is particularly important in the Solvency II context. The region that matters most for SCR estimation is the tail of the real-world distribution of the risk drivers. A fitting design concentrated only around central scenarios may therefore lead to a proxy that behaves well in average situations but extrapolates poorly in stressed regions. Since regulatory capital is ultimately determined by tail events, this would be an undesirable failure mode.

Inner simulation budget per fitting point. The second design choice concerns the number of inner simulations used to compute each target value $\hat{y}_i$. A characteristic feature of LSMC is that one often uses many fitting scenarios but only a moderate number of inner simulations for each of them. Best practice often simulate around 10,000 outer scenarios and for each one, about 2 additional inner simulations. Altogether, this gives 20,000 scenarios to analyze. Fitting and validation points serve different purposes: fitting points should cover the input space well, whereas validation points should be valued more accurately in order to serve as reliable benchmarks [10].

This distinction has an important consequence. The regression is calibrated not on exact conditional expectations, but on Monte Carlo estimates of these expectations. Hence, the fitting targets contain simulation noise in addition to the usual approximation error making the number of inner simulation and the number of fitting points jointly constrained design parameters and not independent choices.

3.3.3 Regression calibration

The next step is to fit a proxy function to the dataset of fitting points. In classical LSMC, the proxy is written as a linear combination of basis functions,

$$\hat{Y}(X) = \sum_{k=0}^{K-1} \beta_k e_k(X),$$

where $e_0(X) = 1$ and the remaining e_k are chosen by the modeller. The coefficients are estimated by ordinary least squares:

$$\hat{\beta} = \arg \min_{\beta \in \mathbb{R}^K} \sum_{i=1}^n \left(\hat{y}_i - \sum_{k=0}^{K-1} \beta_k e_k(x_i) \right)^2.$$

Choice of basis functions. The most common choice in actuarial practice is polynomial basis functions, possibly including interaction terms between risk drivers. Orthogonal polynomial families such as Laguerre [8] or Legendre [3] polynomials are generally preferred over raw monomials for numerical stability: because they are orthogonal over their support, the resulting design matrix is better conditioned and the regression coefficients are less sensitive to small perturbations in the fitting data.

The OLS criterion and its implications. OLS weighs all fitting scenarios equally, regardless of where they fall in the distribution of X . A central scenario and a tail scenario contribute exactly the same amount to the calibration objective. This means the proxy is implicitly optimised for average accuracy across the fitting distribution, not for accuracy in the tail. For SCR estimation, this is a structural misalignment: the regulatory objective is a 99.5% quantile, and the OLS criterion gives no special status to the scenarios that determine it. This does not make OLS wrong as an estimation method, but it means that the quality of the proxy in the capital region is not directly controlled by the calibration objective.

Adaptive model selection. In realistic insurance applications, the calibration is rarely done by running OLS on a fixed large set of basis functions. Instead, adaptive procedures are used to select a parsimonious subset of terms. Starting from a small initial model, candidate terms are added progressively according to a model selection criterion. The Akaike Information Criterion (AIC) is the most common choice, although Bayesian Information Criterion (BIC) and cross-validation are also used. The stepwise direction (forward, backward or bidirectional) affects which terms are retained and in what order. More broadly, all of the calibration choices reflect the same idea: the proxy should be flexible enough to capture the main nonlinearities of the liability function, but sufficiently parsimonious to remain stable out of sample.

3.3.4 Validation

Once calibrated, the proxy must be validated before it is used to estimate the SCR. This step is essential: a good in-sample fit does not guarantee reliable out-of-sample performance, especially in the tail region that is most relevant for capital estimation.

Krah’s framework distinguishes between several types of validation sets, each serving a different diagnostic purpose.

Global out-of-sample validation. A broad validation set of scenarios, not used in calibration and valued with more inner simulations than the fitting points, is used to assess the overall predictive accuracy of the proxy. Standard metrics such as mean absolute error, relative error and R^2 are appropriate here. This type of validation checks whether the proxy generalises beyond the fitting data, but it does not directly address whether it works in the tail.

Capital-region validation. A second validation set focuses specifically on scenarios in or near the 99.5% loss quantile. This is the most relevant diagnostic for SCR purposes because it tests the proxy precisely where the regulatory number is determined. Krah introduces this as a separate validation exercise precisely because a proxy can score well on global metrics and still perform poorly in the capital region. The implication is that global validation is necessary but not sufficient: the capital-region performance must be assessed independently.

The statistical limits of tail validation. There is a fundamental difficulty with capital-region validation that is worth being explicit about. At the 99.5th percentile, only 0.5% of outer scenarios fall in the relevant region by definition. Even with $N = 10,000$ outer scenarios, that is approximately 50 points. Validating the proxy on 50 points produces wide confidence intervals for any tail error metric. A five percent error in the capital region may or may not be statistically significant depending on the shape of the loss distribution near the threshold. This limitation is not specific to any implementation: it follows from the rarity of the event being estimated and cannot be resolved by improving the proxy alone. It implies that reported SCR estimates should always be accompanied by an honest assessment of the uncertainty attributable to proxy approximation in the tail, not only to Monte Carlo sampling variability in the outer scenario generation.

Capital-metric validation. The most directly relevant check would be to compare the proxy-induced SCR estimate with a high-fidelity reference value obtained by brute-force nested simulation. This is called capital-metric validation and it is also the most demanding, because it requires exactly the kind of computation that LSMC is designed to avoid. In practice it is performed on simplified or reduced portfolios where the reference computation is feasible, and the results are extrapolated to the full model under assumptions that are themselves difficult to verify.

3.3.5 5. Aggregation and capital extraction

Once the proxy has been calibrated and validated, it is evaluated on a large set of outer scenarios $\{x_i^*\}_{i=1}^N$ generated under the real-world measure $\mathbb{P}$. This produces approximate liability values $\{\widehat{Y}(x_i^*)\}$ for each outer scenario, from which

$$\widehat{NAV}_1^{(i)} = V_1(A; x_i^*) - \widehat{Y}(x_i^*)$$

is calculated.

Using the loss convention introduced in Chapter 1, the corresponding proxy loss for scenario i is

$$\widehat{L}_1^{(i)} = NAV_0(1 + r_0) - \widehat{NAV}_1^{(i)}.$$

The empirical proxy SCR is then

$$\widehat{SCR}_1 = \widehat{\text{VaR}}_{0.995}(\widehat{L}_1).$$

Equivalently, this selects the adverse 1-in-200 region of the simulated one-year own-funds distribution. If the result is required in time-0 monetary units, it can be multiplied by the deterministic discount factor $v(0, 1)$. Chapter 4 reports the time-1 loss quantile consistently for the reference and all proxy models.

This is where the computational advantage of LSMC becomes decisive. Evaluating $\widehat{Y}$ on $N = 100,000$ or even $N = 1,000,000$ outer scenarios is essentially instantaneous once the proxy coefficients are fixed, regardless of the complexity of the original inner valuation model. The method allows a very large number of real-world simulations for the capital calculation itself, since the proxy evaluation is so cheap. This means that once the fitting and validation effort is complete, the tail of the loss distribution can be estimated with very high precision from the proxy alone.

Nevertheless The quality of this final capital estimate inherits all the approximation errors introduced in the previous steps. A large N at the evaluation stage does not compensate for a poor proxy: it only reduces the Monte Carlo sampling error in the quantile estimation, not the proxy approximation error itself. This is why the calibration and validation steps are not secondary details but the most consequential part of the LSMC framework.

3.4 Regression Specification and Empirical Evidence from the Literature

In the classical LSMC framework, the main modelling choice concerns the regression basis. The standard approach in actuarial applications is to use polynomial basis functions, possibly enriched with interaction terms. This subsection briefly presents this classical specification and then shows, through results from the literature, why these choices matter in practice for insurance liability valuation and SCR estimation.

3.4.1 Classical polynomial basis functions

Polynomial basis expansions are attractive because they are simple, flexible, and easy to combine with ordinary least squares. They allow the proxy to capture not only linear effects, but also curvature and interactions between risk drivers. In practitioner-oriented discussions of LSMC, Koursaris [3] note that orthogonal polynomial families such as Legendre polynomials are often preferred to raw monomials because they improve numerical stability, while still providing sufficient flexibility for many liability functions.

At the same time, polynomial bases immediately raise a practical issue: the number of candidate terms grows quickly with both the number of risk drivers and the degree of the expansion. This means that even in the classical setting, regression specification is closely linked to model complexity control.

3.4.2 Why interaction terms matter

An important empirical lesson from the literature is that interaction terms may substantially improve proxy quality. This is especially relevant in insurance applications, where liabilities often depend jointly on several risk factors.

Floryszczak et al. [9] provide a clear illustration in a Solvency II NAV/SCR setting. They show that a fourth-degree polynomial proxy without interaction terms may fit reasonably in central regions, but fails to reproduce the NAV surface satisfactorily in more stressed parts of the distribution. Once interaction terms are included, the approximation becomes much more accurate in the high-quantile region relevant for capital estimation. This result is particularly useful for the present thesis because it shows that the specification of the regression basis directly affects the quality of the approximation where SCR estimation is most sensitive.

3.4.3 Evidence from realistic insurance portfolios

The practical relevance of classical polynomial LSMC is also confirmed by applications to realistic insurance portfolios. Koursaris [3] report a case study on a with-profits liability portfolio embedded in a full ALM model. Their proxy is fitted in eight risk dimensions and calibrated on around 20,000 ALM scenarios. The validation compares proxy values with more accurate Monte Carlo valuations and shows good agreement both under individual stresses and under simultaneous joint stresses.

This is useful for Chapter 2 because it shows that polynomial LSMC is not only a theoretical tool. It has already been implemented in realistic insurance settings, where the liability function is generated by a genuine ALM model and depends on several financial and actuarial risk drivers.

3.4.4 Adaptive selection and parsimonious specification

The literature also makes clear that polynomial LSMC is rarely used as a fully unrestricted expansion. In practice, candidate basis functions are usually selected through adaptive procedures, often guided by criteria such as AIC. Koursaris [3] explicitly note that when the number of potential polynomial terms becomes large, subset selection is required in order to retain a parsimonious and stable specification.

Krah develops this point in a more formal way in a realistic life insurance setting. Her framework starts from monomial basis functions but imposes restrictions on the admissible terms and on the interaction structure in order to preserve numerical stability and keep the final proxy manageable. For the present thesis, the main message is therefore that in classical LSMC, regression specification cannot be separated from model selection.

3.5 Limitations of Classical LSMC in Practice

Although the literature confirms the usefulness of classical polynomial LSMC, it also highlights a number of practical limitations. These limitations do not invalidate the framework itself, but they do explain why more flexible regression engines are increasingly considered.

3.5.1 Explosion of candidate terms

The first limitation is combinatorial. As the number of risk drivers increases, the number of polynomial terms grows very quickly, especially once higher-order effects and interaction terms are included. Koursaris [3] point out that even for a moderate number of risk drivers, the number of candidate terms may become very large, making subset selection unavoidable.

Krah et al. [10] provide a stronger actuarial illustration of this issue. In their realistic life insurance setting with 15 risk factors, they explain that the candidate-term space may become so large that standard forward or bidirectional procedures are no longer practical without additional structure. This is directly relevant for the present thesis because it shows that the difficulty of classical LSMC is not only to fit a proxy, but also to define a feasible function class.

3.5.2 Sensitivity to basis specification

A second limitation is the sensitivity of the method to the chosen basis. The results of Floryszczak, Le Courtois and Majri [9] already illustrate this clearly: excluding interaction terms can lead to a much poorer approximation in stressed regions, whereas including them improves the fit materially. This means that classical LSMC should not be viewed as a single fixed benchmark. Its performance depends strongly on modelling choices such as the polynomial degree, the interaction structure, and the selection procedure.

3.5.3 Extrapolation and the capital region

A third limitation concerns extrapolation. In solvency applications, the scenarios that matter most for the SCR are often stressed scenarios that are only sparsely represented in the fitting sample. As a result, the proxy may need to extrapolate beyond the region that was well covered during calibration.

Krah's work [2] is especially useful on this point. She shows that improved global out-of-sample performance does not necessarily imply improvement in the capital region, and that residual bias in the SCR-relevant region may remain even when the model performs better overall. For the present thesis, this is an important reminder that good average fit is not sufficient: in a solvency context, the key issue is the behaviour of the proxy near the tail quantile.

3.5.4 Implication for the thesis

Taken together, these results justify the next step of the thesis. Classical LSMC remains the natural benchmark framework because it is interpretable, well estab-

lished, and directly linked to the nested simulation problem. At the same time, the literature shows that its regression step becomes strained when dimensionality increases, when interactions matter strongly, and when high accuracy is needed in the stressed tail region. This is exactly why Chapter 3 retains the LSMC framework while replacing the classical polynomial regression engine by more flexible but still interpretable machine learning methods.

3.6 Conclusion

This chapter introduced the Least-Squares Monte Carlo method as the benchmark framework for approximating the nested simulation problem arising in Solvency II internal models. The central idea is to replace repeated inner valuations by a regression-based proxy function linking the target quantity to a finite set of risk drivers. This makes the overall calculation much more tractable while preserving the economic logic of real-world outer simulation and market-consistent inner valuation.

The chapter also showed that the regression step is a central component of the method. The literature confirms both the usefulness of classical polynomial LSMC and its practical limitations. In particular, it shows that interaction terms may be crucial, that realistic life insurance applications quickly become high-dimensional, and that good global fit does not necessarily imply good performance in the capital region.

For the purpose of this thesis, the conclusion is therefore twofold. First, classical LSMC provides the right conceptual framework for replacing full nested simulations in SCR estimation. Second, its practical limitations justify the search for improved proxy models. The next chapter takes up this question by introducing interpretable machine learning alternatives, namely LocalGLMnet as extensions of the classical regression step within the LSMC framework and Explainable Boosting Machines as a glass-box additive proxy.

4 Explainable Machine Learning Proxy Models

4.1 Motivation: From Polynomial LSMC to Interpretable Machine Learning

Chapter 2 explained how the LSMC framework provides a practical solution to the nested simulation problem arising in Solvency II internal models. Its central idea is to replace repeated inner valuations by a proxy function linking the quantity of interest to a finite set of risk drivers. Within this framework, the traditional choice is to use polynomial regression. This choice has clear advantages: it is transparent, computationally light, and naturally compatible with least-squares estimation. At the same time, the literature also shows that polynomial LSMC becomes increasingly difficult to calibrate when the number of risk drivers grows or when the liability surface is strongly nonlinear and path dependent [2].

This limitation is particularly relevant in life insurance and ALM applications. The value of liabilities may depend simultaneously on several financial and actuarial variables, and the corresponding relationship can be highly nonlinear. In such settings, forcing the proxy into a fixed polynomial structure may lead either to insufficient flexibility or to a rapid explosion of candidate terms. As discussed in the previous chapter, this creates a tension between approximation quality, numerical stability, and interpretability.

A natural response is to consider more flexible machine learning methods. In recent years, neural networks and other statistical learning techniques have been increasingly used in insurance because they can capture complex relationships and often improve predictive performance. The recent literature on solvency proxy modelling confirms that replacing polynomial regression by deep learning can improve accuracy in approximating the mapping between risk drivers and liability values [7]. However, this gain in flexibility comes at a cost. Standard deep neural networks are typically black-box models: they provide predictions, but little direct insight into how those predictions are formed or which risk drivers dominate them.

This lack of explainability is not a minor inconvenience in the present context. In the Solvency II framework, internal models are not only technical devices used to compute a number. They also serve as risk management tools and must remain understandable enough to support decision-making and model governance. Recent work on explainable LSMC stresses precisely this point: while deep learning can improve approximation quality, the highly regulated nature of the insurance industry requires models whose behaviour can be interpreted and linked back to the underlying risk profile. In other words, accuracy alone is not sufficient.

This motivates the present chapter. The objective is not to abandon the LSMC framework developed in Chapter 2, but to revisit its regression step. More specifically, this chapter introduces two interpretable machine learning proxy families that can be embedded within the LSMC logic: LocalGLMnet and Explainable Boosting Machines. The first remains close to neural networks while recovering a GLM-like additive interpretation. The second belongs to the generalized additive model family and relies on cyclic boosting to obtain flexible yet directly interpretable shape functions. Together, these models offer two different ways of improving the flexibility of

the proxy while preserving interpretability.

Before presenting them, it is useful to briefly recall why standard neural networks are attractive for tabular proxy regression and why, in their usual form, they remain insufficient for the purposes of this thesis.

4.2 Feed-Forward Neural Networks for Tabular Proxy Regression

Before introducing specifically interpretable proxy models, it is useful to briefly discuss the role of standard feed-forward neural networks in the present setting. Their importance here is not only practical but also conceptual. On the one hand, they provide a natural extension of classical polynomial LSMC when the mapping from risk drivers to liability values becomes too complex to be captured parsimoniously by a fixed basis expansion. On the other hand, they make clear why interpretability becomes a central issue in solvency proxy modelling.

4.2.1 Mathematical setup

Let $X_1 \in \mathbb{R}^{q_0}$ denote the vector of risk drivers observed at time 1, and let $\mu(X_1)$ denote the conditional quantity to be approximated, for example the market-consistent value of liabilities or another economic balance-sheet variable relevant for SCR estimation. In the classical LSMC framework, one approximates the map

$$X_1 \mapsto \mu(X_1)$$

through a regression-based proxy calibrated on fitting scenarios.

A standard fully connected feed-forward neural network replaces the classical polynomial basis by a composition of nonlinear feature transformations. Following the notation used in the explainable LSMC literature, let $h \in \mathbb{N}$ be the number of hidden layers and let $q_k \in \mathbb{N}$, for $1 \leq k \leq h$, denote the width of each layer. The network defines a mapping

$$x \mapsto z_W(x) = z_{h+1}(x) = (z_h \circ \cdots \circ z_1)(x) \in \mathbb{R}^{q_h},$$

where each layer $z_k : \mathbb{R}^{q_{k-1}} \rightarrow \mathbb{R}^{q_k}$ is an affine transformation followed by a nonlinear activation function, and where W collects all network weights [7]. The vector $z_W(x)$ can be interpreted as a learned representation of the original features.

The final output layer then maps this learned representation into a scalar approximation of the target quantity. In the notation of Perla, Scognamiglio, Spadaro and Zanetti, the resulting proxy takes the form

$$\hat{\mu}^{(\text{DL})}(X_1) = \sum_{k=0}^{q_h} \beta^{(k)} z_W^{(k)}(X_1),$$

where $z_W^{(0)}(X_1) = 1$ plays the role of a bias term and the coefficients $\beta^{(k)}$ belong to the output layer.

4.2.2 Why feed-forward networks are attractive

The main reason for using feed-forward neural networks in this setting is their flexibility. In contrast to polynomial regressions, they do not require the modeller to specify in advance which interactions or nonlinear transformations should be included. Instead, these structures are learned from the fitting data through the hidden layers. This is particularly attractive when the liability surface is high-dimensional and complex, which is often the case in life insurance and ALM applications.

Krah [10] also points out that artificial neural networks are among the stochastic machine learning alternatives capable of capturing the complex nature of cash-flow projection models, although their successful implementation requires careful hyperparameter selection. In this sense, feed-forward networks provide a natural response to some of the limitations of classical polynomial LSMC discussed in Chapter 2.

4.2.3 Why they remain insufficient for the present thesis

Despite their flexibility, standard feed-forward neural networks have an important drawback in the context of this thesis: they are difficult to interpret. Richman and Wüthrich stress that deep learning models perform representation learning internally they are difficult to interpret. Richman and Wüthrich [5] stresses that in a nontransparent way, which makes it difficult to track how individual input components influence the final prediction. In other words, while the model may predict well, it does not naturally explain *why* it predicts in a certain way.

This weakness becomes particularly important in solvency applications. Although neural networks can improve accuracy in SCR proxy modelling, their lack of explainability limits their adoption in the highly regulated insurance sector. In an internal-model setting, one would ideally like not only an accurate approximation of the liability function, but also a proxy whose response to risk drivers can be interpreted and linked back to the insurer’s risk profile.

This motivates the next section, where the LocalGLMnet architecture is introduced as an attempt to retain the flexibility of deep learning while recovering a directly interpretable structure.

4.3 LocalGLMnet

4.3.1 General idea

LocalGLMnet was introduced by Richman and Wüthrich [5] as an interpretable deep learning model for tabular data. The motivation is straightforward. Standard deep neural networks often outperform classical regression models, but they do so by learning hidden representations in a way that is difficult to interpret directly. The LocalGLMnet architecture seeks to retain the predictive flexibility of neural networks while recovering an additive structure reminiscent of generalized linear models.

The key idea is to start from the GLM framework and make the regression coefficients feature dependent. In a classical GLM, the link-transformed response is expressed as a linear predictor with fixed coefficients. In LocalGLMnet, these

coefficients are no longer constant: they are produced by a neural network and may vary from one observation to another. The final prediction therefore remains additive in the original variables, but the weights assigned to those variables adapt locally to the input.

This gives the architecture its name. The model behaves like a *local* generalized linear model, because each observation is associated with its own set of regression coefficients. At the same time, these coefficients are learned by a neural network, which gives the model a flexibility that goes well beyond a standard GLM.

4.3.2 Model structure

Let $x = (x_1, \dots, x_q)$ denote the vector of input features. LocalGLMnet computes a vector of varying regression coefficients $\beta_W(x)$ through a neural network and then forms the prediction through a GLM-like additive structure of the form

$$y = g^{-1}(\beta^{(0)} + \beta_W(x)^\top x),$$

where $\beta^{(0)}$ is an intercept and g is the chosen link function. Thus, the nonlinear part of the model lies in the way the coefficients are generated, not in the final additive form itself.

This structure is central for interpretability. Because the prediction is written as a sum of terms of the form $\beta_W^{(k)}(x)x_k$, one can inspect the contribution of each risk driver to the final prediction. If the coefficient of a variable is approximately constant, the model behaves locally like a classical GLM for that variable. If the coefficient is close to zero, the variable contributes little. If the coefficient depends mainly on its own feature, the effect is essentially nonlinear but largely separable. If it depends strongly on other features, this indicates interactions. This makes the model much easier to interpret than a standard feed-forward network.

Richman and Wüthrich describe these varying regression coefficients as regression attention weights. Their role is analogous to an attention mechanism in the sense that the model learns how much weight each original feature should receive in a given context. Because the final prediction still uses the original features directly, the model maintains a transparent link between inputs and outputs.

4.3.3 Why LocalGLMnet is attractive for LSMC proxy modelling

The LocalGLMnet architecture is particularly attractive for the type of problem studied in this thesis. In proxy modelling for solvency applications, the target quantity is often driven by several risk drivers in a way that is nonlinear, heterogeneous across scenarios, and potentially interaction-rich. A classical polynomial proxy may struggle to represent this mapping parsimoniously. A standard deep neural network may approximate it more accurately, but offers limited interpretability. LocalGLMnet sits between these two extremes.

This point is made very explicitly in the recent explainable LSMC paper. Perla et al. [7] argue that the model continues to exploit the flexibility of neural networks to approximate the relationship between liabilities and risk drivers, while at the same time preserving an interpretable structure through the varying regression

coefficients. They also emphasize that inspecting these coefficients (which they call attention coefficients) provides useful insight into the basis of the predictions and into the role of each risk factor.

In addition, LocalGLMnet naturally fits the LSMC logic developed in Chapter 2. The input remains the vector of time-1 risk drivers, and the output remains the quantity to be proxied. Only the regression engine changes. This makes the model particularly suitable for the present thesis, where the objective is not to propose an entirely new capital framework, but rather to improve the proxy construction step within the existing LSMC methodology.

4.3.4 Interpretability and variable selection

Another important feature of LocalGLMnet is that it supports a richer interpretability analysis than a standard DNN. Richman and Wüthrich emphasize that the model allows not only the interpretation of predictions through an additive decomposition, but also variable selection and the study of interactions [5]. In later work, Richman and Wüthrich also show how regularization can be incorporated into the architecture to encourage sparsity in the attention coefficients, thereby improving variable selection and reducing unnecessary complexity [11].

This is especially relevant in an actuarial setting. For a solvency proxy, one often wants to know not only whether the model predicts well, but also which risk drivers materially influence the value of liabilities and how these influences vary across scenarios. LocalGLMnet offers a direct way to study such questions through the estimated coefficient functions themselves.

For the purpose of this chapter, the main point is therefore not yet to claim that LocalGLMnet is always superior to all alternatives. Rather, it provides a very appealing compromise: it preserves the flexibility of deep learning while keeping a structure that remains interpretable enough to be meaningful in a regulated insurance setting.

4.4 Explainable Boosting Machines

4.4.1 General idea

Explainable Boosting Machines (EBMs) belong to a different family of interpretable models. Rather than modifying a neural-network architecture to recover interpretability, EBMs start from a generalized additive model structure and enrich it with modern boosting techniques. In the InterpretML framework, EBM is presented as a glass-box model designed to combine competitive accuracy with high interpretability [4]. More recently, the insurance literature has also highlighted its relevance for actuarial applications, especially in pricing contexts where transparency and predictive performance must both be taken seriously [12].

The basic additive form of an EBM is

$$g(\mathbb{E}[Y]) = \beta_0 + \sum_j f_j(x_j),$$

possibly enriched with a selected number of pairwise interaction terms,

$$g(\mathbb{E}[Y]) = \beta_0 + \sum_j f_j(x_j) + \sum_{i < j} f_{ij}(x_i, x_j).$$

Here, the functions f_j and f_{ij} are not specified parametrically in advance. They are learned from the data, which gives the model much more flexibility than a standard GLM while preserving an additive and interpretable structure.

4.4.2 Cyclic boosting and shape functions

A distinctive feature of EBM is the way these additive functions are estimated. Instead of fitting all terms simultaneously, the model learns them through a cyclic gradient boosting procedure. Each feature is updated in turn, typically after being partitioned into bins, and the corresponding contribution function is refined gradually across boosting iterations. Because the algorithm cycles through the variables in a round-robin manner using a low learning rate, the final model is less dependent on the order in which features are processed and remains interpretable at the level of the individual shape functions [4, 12].

In practical terms, this means that each feature has an associated *shape function* that can be plotted and interpreted directly. These functions show how the model’s prediction changes as a given feature varies, all else being held fixed within the additive structure. This is one of the strongest advantages of EBM: the model is not only globally interpretable, but also visually interpretable.

The recent actuarial EBM paper by Krúpová, Rachdi and Guibert emphasizes exactly this point. They describe EBM as a glass-box model based on a GAM structure and a cyclic boosting algorithm, capable of accounting for both univariate and pairwise interaction effects while providing natural explanations for them [12]. Although their application concerns non-life pricing rather than solvency capital modelling, the methodological message is highly relevant here.

4.4.3 Bagging and interaction selection in InterpretML

In addition to the cyclic boosting procedure, the implementation of EBM in InterpretML uses bagging to make the learned shape functions more stable. Two parameters are particularly relevant: `inner_bags` and `outer_bags`. Although their names are similar, they act at different levels of the fitting procedure [4].

The *outer bags* correspond to repeated fits of the EBM on different random subsets of the training data. The resulting shape functions are then combined by the algorithm. This averaging reduces the dependence of the fitted model on one particular sample and generally produces smoother shape functions. The variation across outer bags can also be used by InterpretML to provide error bounds around the estimated effects. Increasing the number of outer bags can therefore improve stability, but it also increases the training time.

The *inner bags* operate at a finer level. During boosting, InterpretML can draw bootstrap subsamples when the individual shallow trees are grown. These additional resamples introduce another averaging step inside the estimation of the shape functions. In the InterpretML implementation, setting `inner_bags` equal to zero

switches off this additional resampling step. The usefulness of inner bagging depends on the application and is therefore treated as a tuning choice rather than as a fixed property of the EBM model.

The parameter `interactions` controls a different aspect of model complexity. EBM can automatically detect relevant pairwise effects between the explanatory variables, following the GA²M framework. If an integer K_{int} is supplied, InterpretML selects K_{int} pairwise interaction terms and adds them to the main effects [4, 12]. The model can then be written as

$$g(\mathbb{E}[Y]) = \beta_0 + \sum_{j=1}^p f_j(x_j) + \sum_{m=1}^{K_{\text{int}}} f_{a_m b_m}(x_{a_m}, x_{b_m}),$$

where (a_m, b_m) denotes one of the selected pairs. Hence, for example, `interactions=10` means that up to ten pairwise interaction functions are included; it does not mean that the model contains a ten-way interaction. With p explanatory variables there are at most $p(p-1)/2$ different pairs. A larger value of K_{int} gives the model more flexibility, but it also increases complexity and can make the final model harder to interpret. For this reason, the number of interactions and the bagging parameters are selected empirically in the numerical study using an independent validation set.

Finally, the terms *inner bags* and *outer bags* should not be confused with the inner and outer Monte Carlo simulations of the LSMC framework. They are re-sampling parameters internal to the EBM training algorithm and have no direct connection with the nested simulation structure used to generate the liability values.

4.4.4 Why EBM is relevant for this thesis

EBM is not originally a model developed for LSMC proxy modelling in life insurance. Its main recent insurance applications are in pricing and tariff analysis [12]. For this reason, it should not be presented as a standard actuarial solvency proxy model in the same way as polynomial LSMC or LocalGLMnet. However, it is still a very natural candidate for the present thesis.

The reason is that the structure of EBM is closely aligned with the needs of a proxy modelling exercise. The model is additive, interpretable, flexible, and capable of incorporating selected pairwise interactions. It can therefore approximate nonlinear relationships between risk drivers and liability values while keeping a transparent decomposition of the prediction. This makes it an appealing alternative to both rigid polynomial specifications and opaque black-box models.

Moreover, EBM provides an interesting contrast with LocalGLMnet. Both models aim at combining accuracy and interpretability, but they do so in very different ways. LocalGLMnet remains rooted in deep learning and constructs scenario-dependent GLM coefficients. EBM, by contrast, remains much closer to the GAM family and explains predictions through learned shape functions. This difference is precisely what makes the comparison valuable in the present thesis.

4.4.5 Strengths and caveats

The main strength of EBM is that its interpretability is intrinsic rather than added after the fact. Unlike post-hoc explanation methods, the model itself is transparent. This is one reason why EBM is described in the literature as a glass-box model [4, 12]. In regulated or decision-sensitive environments, this is a clear advantage.

At the same time, EBM also has its own limitations. Because it remains fundamentally additive, its ability to capture very complex interaction structures is more limited than that of a fully flexible neural network, unless enough pairwise terms are explicitly incorporated. In addition, while EBM is highly interpretable at moderate dimension, recent work points out that very large numbers of terms can reduce transparency and increase scoring time; this motivates sparsity-inducing post-processing methods in higher-dimensional settings [13].

For the purposes of this thesis, the important point is therefore not that EBM is universally dominant, but that it provides a strong interpretable baseline from a different modelling tradition than LocalGLMnet. This makes it a valuable model to include in the comparison.

4.5 Positioning the Models within the LSMC Framework

At this stage, it is useful to position the different proxy models presented so far within the broader LSMC framework developed in Chapter 2. The common point between them is that they all address the same approximation task: given a set of fitting scenarios generated from nested simulations, they seek to approximate the mapping from risk drivers to an economic quantity of interest. What differs is the structure of the regression engine and the trade-off it implies between flexibility and interpretability.

Classical polynomial LSMC represents the most transparent and traditional approach. Its additive least-squares structure is easy to explain and closely connected to classical actuarial modelling. However, as Chapter 2 showed, it can become difficult to calibrate in higher dimensions and may struggle to capture complex interactions parsimoniously.

Standard feed-forward neural networks sit at the opposite end of the spectrum. They are attractive because they can learn complex nonlinear relations without requiring the modeller to specify these relations in advance. In the explainable LSMC literature, this greater flexibility is explicitly identified as one of the reasons why neural networks may outperform polynomial regression in SCR proxy modelling [7]. Their weakness, however, is that they remain black-box models and offer limited direct insight into the structure of their predictions.

LocalGLMnet can be seen as a compromise between these two approaches. It keeps the flexibility of neural networks, but reorganizes the prediction in a GLM-like additive form with scenario-dependent coefficients. This makes it possible to study the direct contribution of each risk driver and to recover a meaningful local interpretation of the prediction [5, 7]. In the context of this thesis, LocalGLMnet is therefore best viewed as an explainable neural-network extension of the classical LSMC regression step.

EBM occupies a somewhat different position. It is not based on a neural-network

architecture, but on an additive GAM-like structure learned through cyclic boosting. Its interpretability comes from the explicit shape functions associated with each feature and with selected interactions [4, 12]. Within the LSMC framework, EBM can therefore be viewed as a glass-box nonlinear proxy model. It does not rely on the same logic as LocalGLMnet, but it addresses the same need: replacing rigid classical bases by a more flexible model that remains understandable.

This positioning is important for the rest of the thesis. The objective is not to compare unrelated machine learning tools in isolation, but to compare different ways of upgrading the regression step inside the same LSMC methodology. From this viewpoint, the benchmark remains classical polynomial LSMC; the feed-forward neural network represents the flexible but non-explainable alternative; LocalGLMnet represents an interpretable deep learning extension; and EBM represents an interpretable additive boosting alternative.

The next chapter illustrates these differences on a insurance portfolio of these three models to evaluate the solvency capital.

5 Numerical Implementation and Empirical Results

5.1 Introduction

The preceding chapters developed the Solvency II nested-simulation problem, the classical Least-Squares Monte Carlo (LSMC) approach and the two interpretable machine learning alternatives considered in this thesis. The purpose of the present chapter is to bring these elements together in a common numerical experiment. Polynomial LSMC, Explainable Boosting Machines (EBMs) and LocalGLMnet are applied to the same Universal Life portfolio, the same seven risk drivers and the same validation sets.

The numerical model is deliberately stylised. It is not intended to reproduce the complete internal model of a particular insurer. The objective is instead to obtain a liability function that contains the main mechanisms that make life-insurance valuation difficult: stochastic interest rates, equity risk, stochastic volatility, credit-spread movements, dynamic surrender behaviour, stochastic mortality, guarantees and future profit sharing. This makes it possible to study proxy performance in a setting that remains sufficiently transparent to trace how each state variable enters the asset and liability projections.

The empirical comparison is also designed to separate different notions of accuracy. A proxy may be accurate over the full validation domain but still contain a systematic error around the 99.5% loss quantile. For this reason, global prediction accuracy, the adverse region, a narrower capital region, the final SCR error and stability across numerical repetitions are analysed separately. This follows the validation logic used in the LSMC literature, where the region relevant for capital calculation must be examined in addition to ordinary out-of-sample fit [2, 3, 10].

The main experiment approximates the complete time-1 best estimate liability BEL_1 . A second experiment then approximates only the future discretionary benefit component FDB_1 , following the decomposition used in the recent LocalGLMnet SCR literature [7]. The two targets are compared using the same nested reference loss distribution.

The chapter is organised as follows. Section 5.2 describes the portfolio and the financial and actuarial projection model. Section 5.3 explains the nested simulation, the present-value calculation and the construction of the fitting and validation sets. Section 5.4 gives the final specifications of the three proxy models. Section 5.5 reports the results for BEL_1 , while Section 5.6 studies the numerical and economic sensitivity of this experiment. Section 5.7 considers the alternative FDB_1 target. Section 5.8 discusses the EBM explanations, and the last two sections present the main limitations and conclusions.

5.2 Universal Life Portfolio and ALM Projection Model

5.2.1 Portfolio representation

The liability side is represented by a stylised Universal Life portfolio with annual crediting, death and surrender cash flows, a minimum guaranteed rate and a profit-sharing mechanism. The portfolio contains 10,000 policies grouped into four model

points. Policies within a model point have the same initial age, initial mathematical reserve and guaranteed technical rate. This aggregation reduces the computational burden while preserving differences in age and reserve size.

Table 1: Model points of the Universal Life portfolio

Model point	Policies	Initial age	Reserve per policy	Initial guaranteed rate
1	4,000	40	15,000	1.5%
2	3,000	50	25,000	1.5%
3	2,000	60	40,000	1.5%
4	1,000	70	50,000	1.5%

The total initial mathematical reserve is

$$R_0 = \sum_{m=1}^4 N_{m,0} Res_{m,0} = 265,000,000. \quad (1)$$

The initial asset value is specified through a surplus ratio $s_0 = 20\%$,

$$A_0 = (1 + s_0)R_0 = 318,000,000. \quad (2)$$

The full numerical run uses a Monte Carlo estimate of the initial best estimate liability equal to 255.764 million with 1,000 inner simulations, giving an initial net asset value of approximately 62.236 million. Since BEL_0 is itself simulated, its numerical convergence is examined separately in Section 5.3.5 rather than treating this single estimate as an exact value.

The projection horizon is 40 years and the portfolio is run off without new business. The horizon is sufficiently long for the remaining policies and reserves to be substantially run off, while keeping the annual projection simple.

5.2.2 Asset portfolio and economic dynamics

The asset side is represented by three investment classes: equity, bonds and cash. This type of allocation is consistent with the stylised life-insurance balance sheets commonly used in the LSMC literature [1]. In the present experiment, the allocation is kept constant over the projection horizon and is given by

$$(w_E, w_B, w_C) = (20\%, 70\%, 10\%).$$

The exact weights are illustrative rather than calibrated to a particular insurer. They produce a bond-dominated asset portfolio, while retaining sufficient equity exposure for the profit-sharing mechanism to react to financial-market movements.

For each projection year, the portfolio return is constructed as

$$R_t^{\text{Port}} = w_E R_t^E + w_B R_t^B + w_C R_t^C, \quad (3)$$

where R_t^E , R_t^B and R_t^C denote respectively the annual equity, bond and cash returns. The three components are described below. This portfolio return is important on

both sides of the balance sheet: it determines the asset performance and also enters the profit-sharing rule of the Universal Life contract.

Interest-rate uncertainty is represented by two Gaussian mean-reverting factors, denoted by $x_t^{(1)}$ and $x_t^{(2)}$. Under the valuation dynamics, each factor follows an Ornstein–Uhlenbeck process,

$$dx_t^{(k)} = -a_k x_t^{(k)} dt + \sigma_k dW_t^{Q,k}, \quad k = 1, 2, \quad (4)$$

where $a_k > 0$ is the mean-reversion speed and $\sigma_k > 0$ is the volatility of factor k . A larger value of a_k means that shocks disappear more quickly, whereas a smaller value produces a more persistent factor.

Because the projection frequency is annual, the exact one-year transition used in the numerical engine is

$$x_t^{(k)} = e^{-a_k} x_{t-1}^{(k)} + \sqrt{v_k} Z_{k,t}^Q, \quad v_k = \frac{\sigma_k^2 (1 - e^{-2a_k})}{2a_k}, \quad (5)$$

with $Z_{k,t}^Q \sim \mathcal{N}(0, 1)$. The short rate is obtained by adding the two deviations to the initial rate level:

$$r_t = r_0 + x_t^{(1)} + x_t^{(2)}. \quad (6)$$

The numerical values are

$$r_0 = 2\%, \quad (a_1, \sigma_1) = (0.10, 1.0\%), \quad (a_2, \sigma_2) = (0.50, 0.8\%).$$

These values are illustrative rather than statistically calibrated. The first factor is deliberately more persistent and therefore mainly represents movements in the general level of interest rates. The second factor reverts more rapidly and introduces additional variation in the shape of the yield curve.

To connect the two factors with a longer maturity, the model constructs an eight-year yield,

$$y_t^{(8)} = r_0 + \frac{B_1(8)}{8} x_t^{(1)} + \frac{B_2(8)}{8} x_t^{(2)} + c_8, \quad (7)$$

where

$$B_k(\tau) = \frac{1 - e^{-a_k \tau}}{a_k}. \quad (8)$$

The functions $B_k(\tau)$ determine how strongly each rate factor affects a yield of maturity τ . The quantity

$$slope_t = y_t^{(8)} - r_t$$

is subsequently used as the yield-curve slope driver supplied to the proxy models.

The constant c_8 is an affine convexity adjustment. For each Gaussian factor, the implementation computes

$$A_k(\tau) = \frac{\sigma_k^2}{2a_k^2} (\tau - B_k(\tau)) - \frac{\sigma_k^2 B_k(\tau)^2}{4a_k}, \quad (9)$$

and sets

$$c_8 = -\frac{A_1(8) + A_2(8)}{8}. \quad (10)$$

This term is deterministic: it does not create an additional risk factor. Its purpose is simply to obtain a less trivial eight-year yield than a pure linear combination of the two rate factors. With the parameters used here the adjustment is approximately -7 basis points. Its numerical effect is therefore modest but it affects the level of the eight-year yield and consequently the slope variable and the future guaranteed-rate reset described below.

Equity returns are modelled in lognormal form. Conditional on the volatility state σ_t , the annual return is

$$R_t^E = \exp \left(m_t - \frac{1}{2} \sigma_t^2 + \sigma_t Z_{E,t} \right) - 1, \quad (11)$$

where the drift m_t depends on whether the simulation is performed under the real-world or the valuation dynamics. This distinction is discussed explicitly in Section 5.2.3.

The equity volatility is itself stochastic. Rather than keeping a constant volatility throughout the 40-year projection, it evolves around the long-run reference level

$$\bar{\sigma} = 18\%$$

according to

$$\sigma_t = \text{clip} [\bar{\sigma} + \phi_\sigma (\sigma_{t-1} - \bar{\sigma}) + \eta_\sigma Z_{\sigma,t}, \sigma_{\min}, \sigma_{\max}], \quad (12)$$

with

$$\phi_\sigma = 0.80, \quad \eta_\sigma = 4\%, \quad \sigma_{\min} = 5\%, \quad \sigma_{\max} = 60\%.$$

The value 18% should therefore not be interpreted as the volatility of every simulated year. It is the central level around which the stochastic volatility state moves. The persistence parameter $\phi_\sigma = 0.80$ implies that a high- or low-volatility environment tends to remain present for several years, while $\eta_\sigma = 4\%$ introduces annual innovations. The lower and upper bounds prevent negative or unrealistically extreme volatility values.

For the initial one-year real-world scenario, the state is sampled from the stationary distribution,

$$\sigma_1 = \text{clip} \left[\bar{\sigma} + \frac{\eta_\sigma}{\sqrt{1 - \phi_\sigma^2}} Z_{\sigma,1}, \sigma_{\min}, \sigma_{\max} \right]. \quad (13)$$

Hence the proxy variable vol_1 is the realised time-1 volatility state, not the fixed value of 18%.

The numerical values are not obtained from a statistical volatility calibration. They are illustrative assumptions chosen to create a persistent but non-degenerate equity-volatility risk factor. The inclusion of this risk factor is economically relevant for a participating life-insurance contract because changes in equity volatility affect the distribution of future investment returns and therefore the value of future profit sharing. Floryszczak et al. [1], for example, show that equity volatility can have a material impact on the NAV and SCR of a participating life insurer. Their analysis supports the inclusion of equity volatility as a risk factor.

The cash allocation does not contain an additional stochastic process. It earns the short-term risk-free rate. In the first real-world projection year, the rate known at time 0 is used,

$$R_1^C = r_0. \quad (14)$$

For the subsequent inner projection years, cash earns the simulated short rate of the corresponding projection year,

$$R_t^C = r_t, \quad t \geq 2. \quad (15)$$

Cash therefore enters the total portfolio return through the term $w_C R_t^C$ in Equation (3).

The bond allocation is exposed to both interest-rate and credit-spread risk. Interest-rate sensitivity is represented with a duration-convexity approximation. For projection years after time 1, define

$$\Delta y_t^{(8)} = y_t^{(8)} - y_{t-1}^{(8)}.$$

The bond return is approximated by

$$\begin{aligned} R_t^B = & y_{t-1}^{(8)} - D_B \Delta y_t^{(8)} + \frac{1}{2} D_B^2 \left(\Delta y_t^{(8)} \right)^2 \\ & + w_{cr} [c_{t-1} - D_{cr}(c_t - c_{t-1})], \end{aligned} \quad (16)$$

where $D_B = 8$ years.

The first term, $y_{t-1}^{(8)}$, approximates the return earned from holding the bond during the year if yields do not move. The second term is the first-order price effect of a change in yield: if yields increase, the bond price decreases. The third term introduces the usual second-order convexity correction to this price change.

The last term represents the credit-spread contribution. The variable c_t is the spread of the corporate bond component over the risk-free bond yield. It follows

$$c_t = \max \{ \bar{c} + \phi_c (c_{t-1} - \bar{c}) + \sigma_c Z_{c,t}, 0 \}, \quad (17)$$

with

$$\bar{c} = 1.2\%, \quad \phi_c = 0.85, \quad \sigma_c = 0.4\%.$$

The parameter $\bar{c}$ is the long-run spread level. The persistence $\phi_c = 0.85$ means that a widening or tightening of spreads tends to persist, whereas σ_c determines the size of annual spread innovations. The maximum with zero prevents negative credit spreads in this simplified model.

As for volatility, the time-1 credit driver is drawn from the stationary distribution,

$$c_1 = \max \left\{ \bar{c} + \frac{\sigma_c}{\sqrt{1 - \phi_c^2}} Z_{c,1}, 0 \right\}. \quad (18)$$

The proxy variable $credit_1$ is therefore the realised spread at the one-year risk horizon and becomes the initial credit state of the subsequent inner valuation.

The credit contribution to the bond return has two parts. The current spread c_{t-1} represents the additional return received for holding the corporate bond component. The term

$$-D_{\text{cr}}(c_t - c_{t-1})$$

captures the associated mark-to-market effect. For example, when the spread widens, $c_t - c_{t-1} > 0$, the price of the corporate bond decreases and this term reduces the bond return. When spreads tighten, the opposite effect occurs.

The spread duration is set to

$$D_{\text{cr}} = 5$$

years, while

$$w_{\text{cr}} = 40\%$$

of the bond allocation is assumed to be exposed to corporate credit spreads. Since bonds represent 70% of total assets, this corresponds to 28% of the total asset portfolio being exposed to this simplified spread component.

These credit parameters are illustrative rather than market calibrated. Their purpose is to introduce a persistent credit-risk channel that is large enough to affect the asset return without dominating the interest-rate component. The importance of explicitly representing corporate credit risk in an insurer's economic scenario generator is discussed by Krah [2]. The present model is deliberately simpler: it models spread movements and their effect on bond returns, but does not model rating migration or default events separately.

For the first projection year, the same logic is applied from the initial states $c_0 = \bar{c}$ and

$$y_0^{(8)} = r_0 + c_8.$$

The realised time-1 bond return is therefore

$$\begin{aligned} R_1^B = & r_0 - D_B \left(y_1^{(8)} - y_0^{(8)} \right) + \frac{1}{2} D_B^2 \left(y_1^{(8)} - y_0^{(8)} \right)^2 \\ & + w_{\text{cr}} [\bar{c} - D_{\text{cr}}(c_1 - \bar{c})]. \end{aligned} \quad (19)$$

Together with the first-year equity and cash returns, this produces R_1^{Port} , which is one of the seven proxy drivers.

5.2.3 Real-world and valuation dynamics

The one-year SCR requires two different probability views, as described in Chapter 1. The outer scenarios from $t = 0$ to $t = 1$ represent possible real-world developments and are generated under $\mathbb{P}$. Conditional on the resulting state X_1 , the remaining liability is valued using inner scenarios under the valuation dynamics. This is the standard nested structure used in LSMC applications [1, 2].

The distinction is explicit for the equity and interest-rate models. In the outer equity simulation,

$$R_1^E = \exp \left(\mu_E - \frac{1}{2} \sigma_1^2 + \sigma_1 Z_E^P \right) - 1, \quad \mu_E = 6\%, \quad (20)$$

whereas the inner projection uses the short rate in Equation (11). For the two rate factors, the one-year real-world transition is

$$x_1^{(k)} = \delta_k^P + \sqrt{v_k} Z_k^P, \quad \delta_k^P = \frac{\sigma_k \lambda_r}{a_k} (1 - e^{-a_k}), \quad (21)$$

while the inner valuation projection removes this risk-premium shift and follows Equation (5). The parameter $\lambda_r = 0.25$ is used only in the real-world rate step.

No separate market-price-of-risk adjustment is introduced for the volatility and credit-spread state processes. They are propagated with the same transition laws inside the conditional valuation. The same is true for the actuarial mortality and behavioural lapse states, which are projected according to their best-estimate stochastic laws. This is an explicit simplification of the stylised numerical model rather than a claim that all risk factors have identical real-world and pricing dynamics. The important point for the experiment is that the equity drift and interest-rate premium are not carried from the outer real-world step into the inner valuation step.

5.2.4 Liability cash flows, guarantees and profit sharing

For each model point m , let $N_{m,t}$ be the number of policies in force and $Res_{m,t}$ the reserve per policy. The initial guaranteed rate is $i_0^g = 1.5\%$. The annual profit-sharing rate is

$$PS_t = \max \left\{ \beta \left(R_t^{\text{Port}} - i_t^g - d \right), 0 \right\}, \quad (22)$$

with $\beta = 30\%$ and $d = 1.5\%$. The parameter d is described here simply as the additional margin in the profit-sharing formula. It implies that profit sharing becomes positive when the portfolio return exceeds the guaranteed rate plus this margin. The reserve of a surviving contract is then rolled forward according to

$$Res_{m,t} = Res_{m,t-1} (1 + i_t^g + PS_t). \quad (23)$$

The guaranteed rate is not fixed for the full 40-year horizon. It is reset at years 8, 16, 24, 32 and 40 according to

$$i_t^g = \max \left\{ \alpha y_t^{(8)} - m_g, i_{\min}^g \right\}, \quad (24)$$

where $\alpha = 90\%$, $m_g = 0.7\%$ and $i_{\min}^g = 0$. The future guaranteed rates are therefore not fixed at time 1 from a forward-rate curve. At each future reset date they are calculated from the simulated eight-year yield of that path. Between reset dates the most recently determined guaranteed rate remains in force.

Mortality is based on a Gompertz–Makeham force

$$\mu(x) = A + Bc^x. \quad (25)$$

A stochastic mortality index κ_t modifies the baseline level,

$$q_{x,t} = 1 - \exp \left\{ -\mu(x) \exp(\kappa_t) \right\}, \quad \kappa_t = \kappa_{t-1} + \mu_\kappa + \sigma_\kappa Z_{\kappa,t}. \quad (26)$$

The construction is Lee–Carter-style in the sense that one common stochastic index moves mortality for all model points; it is not a statistical calibration of a Lee–Carter model to population data.

Surrender behaviour combines a base rate, a response to the difference between the short rate and the guaranteed rate, and a persistent behavioural state b_t :

$$b_t = \phi_b b_{t-1} + \sigma_b Z_{b,t}, \quad (27)$$

$$p_t^{surr} = \text{clip} [p_0 + \gamma \max(r_t - i_t^g, 0) + b_t, 0, 0.95]. \quad (28)$$

Thus the variable called $lapse_1$ in the proxy vector is the behavioural shift b_1 , not the total surrender probability. The total surrender probability is obtained only after combining b_1 with the base rate and the interest-rate response in Equation (28).

Given mortality and surrender rates, annual decrements are

$$\begin{aligned} N_{m,t}^d &= N_{m,t-1} q_{m,t}, \\ N_{m,t}^s &= (N_{m,t-1} - N_{m,t}^d) p_t^{surr}, \\ N_{m,t} &= N_{m,t-1} - N_{m,t}^d - N_{m,t}^s. \end{aligned} \quad (29)$$

Death benefits are paid at the reserve in force at the beginning of the year. A surrender receives $1 - \pi$ of this reserve, where $\pi = 5\%$ is the surrender penalty. The annual cash flow before maturity is therefore

$$CF_t = \sum_m Res_{m,t-1} [N_{m,t}^d + (1 - \pi) N_{m,t}^s]. \quad (30)$$

At $t = T$, the end-of-year reserve of the remaining policies is additionally paid as a maturity benefit. This ordering makes the path of each state variable explicit: the financial states determine the portfolio return and guarantee, the guarantee and behavioural state determine surrender, mortality and surrender determine the in-force counts, and the resulting cash flows are discounted in the inner valuation.

5.2.5 High-dimensional risk-driver vector

The proxy models receive the seven-dimensional time-1 state

$$X_1 = (r_1, slope_1, R_1^{\text{Port}}, vol_1, credit_1, lapse_1, \kappa_1). \quad (31)$$

All seven variables are produced by the outer real-world simulation and are sufficient to reconstruct the time-1 state used to start the inner valuation. In particular, R_1^{Port} determines the profit sharing already credited during the first year, $credit_1$ and vol_1 become the starting states of their persistent processes, and $lapse_1 = b_1$ enters the time-1 surrender probability.

Table 2: Risk-driver specifications and corresponding literature benchmarks

Driver	Specification in this thesis	Literature benchmark
r_1	Two-factor Gaussian OU short-rate model	G2N++ two-factor Gaussian model, Perla et al. (2025)
$slope_1$	$y_1^{(8)} - r_1$, derived from the rate factors	Multi-factor yield-curve representation, Koursaris (2011)
R_1^{Port}	Weighted equity–bond–cash return	Participating-life ALM model, Floryszczak et al. (2016)
vol_1	Bounded AR(1) equity-volatility process	Stochastic-variance model, Krah (2021)
$credit_1$	AR(1) credit spread with duration-based repricing	Credit-risk modelling framework, Krah (2021)
$lapse_1$	Rate-dependent lapse with behavioural AR(1) shift	Market-contingent surrender model, Floryszczak et al. (2016)

5.2.6 Parameter specification

The experiment does not claim a statistical or market calibration of the complete ALM model. The parameters have three different roles. Some values are contract or portfolio inputs of the stylised example, such as the model points, initial guaranteed rate and asset weights. Other values are illustrative economic parameters chosen to generate persistent but non-degenerate movements in the seven risk factors. Finally, the initial surplus and profit-sharing parameters were chosen jointly so that the initial balance sheet remains solvent while the profit-sharing option is neither almost always inactive nor almost always active. The sensitivity of the main liability parameters is examined in Section 5.6.5. More detailed parameter and hyperparameter experiments are reported in the Appendix.

Table 3: Main financial parameters used in the numerical experiment

Component	Parameter	Value	Role in the experiment
Initial rate	r_0	2.0%	Illustrative initial market level
Rate factor 1	a_1, σ_1	0.10, 1.0%	Level persistence and volatility
Rate factor 2	a_2, σ_2	0.50, 0.8%	Additional slope movement
Rate premium	λ_r	0.25	Outer $\mathbb{P}$ -drift adjustment only
Equity	$\mu_E, \bar{\sigma}$	6.0%, 18.0%	Illustrative return and volatility levels
Volatility state	ϕ_σ, η_σ	0.80, 4.0%	Persistent stochastic volatility
Volatility bounds	$\sigma_{\min}, \sigma_{\max}$	5%, 60%	Numerical and economic bounds
Bond	D_B	8 years	Duration approximation
Credit allocation	w_{cr}	40% of bonds	Corporate bond component
Credit state	$\bar{c}, \phi_c, \sigma_c$	1.2%, 0.85, 0.4%	Persistent spread process
Credit duration	D_{cr}	5 years	Spread mark-to-market sensitivity
Asset weights	w_E, w_B, w_C	20%, 70%, 10%	Stylised ALM allocation

Table 4: Main liability parameters used in the numerical experiment

Component	Parameter	Value	Role in the experiment
Initial surplus	s_0	20%	Positive initial net asset value
Initial guarantee	i_0^g	1.5%	Stylised contractual guarantee
Guarantee reset	period	8 years	Years 8,16,24,32,40
Guarantee rule	$\alpha, m_g, i_{\min}^g$	90%, 0.7%, 0	Illustrative pathwise reset rule
Profit sharing	β, d	30%, 1.5%	Non-trivial discretionary option
Base surrender	p_0	2.0%	Illustrative annual base rate
Rate sensitivity	γ	0.80	Dynamic surrender response
Behavioural lapse	ϕ_b, σ_b	0.80, 3.0%	Persistent lapse-state variation
Surrender penalty	π	5.0%	Reduction of surrender payment
Gompertz–Makeham	A, B, c	0.00022, $2.7 \cdot 10^{-6}$, 1.124	Age-dependent mortality pattern
Mortality index	$\mu_\kappa, \sigma_\kappa$	-1.0%, 5.0%	Systematic mortality movement
Projection horizon	T	40 years	Run-off horizon

5.3 Nested Simulation and Validation Design

5.3.1 One-year valuation and SCR definition

For each outer scenario i , the real-world simulation produces a state $X_1^{(i)}$, a first-year portfolio return and the first-year liability cash flow. Conditional on this state, the remaining liability is valued by inner simulations. The time-1 best estimate is

$$BEL_1(x) = \mathbb{E}^Q \left[\sum_{t=2}^T D(1, t) CF_t \middle| X_1 = x \right]. \quad (32)$$

The time-1 asset and net asset values are

$$\begin{aligned} A_1^{(i)} &= A_0(1 + R_1^{\text{Port},(i)}) - CF_1^{(i)}, \\ NAV_1^{(i)} &= A_1^{(i)} - BEL_1^{(i)}. \end{aligned} \quad (33)$$

Following the loss-based convention introduced in Chapter 1, the numerical implementation defines the risk-free adjusted one-year loss, expressed in time-1 monetary units, as

$$L_1^{(i)} = NAV_0(1 + r_0) - NAV_1^{(i)} \quad (34)$$

and calculates

$$SCR^{\text{ref}} = \text{VaR}_{0.995}(L_1^{\text{ref}}). \quad (35)$$

The same loss definition and the same outer scenarios are used for every proxy. The reference is therefore a numerical benchmark for the proxy comparison, not an analytical value.

5.3.2 Present value of an inner path

For outer scenario i and inner path j , the present value used in the nested valuation is

$$PV_{i,j} = \sum_{t=2}^T D_{i,j}(1, t) [CF_{i,j,t}^d + CF_{i,j,t}^s + \mathbf{1}_{\{t=T\}} CF_{i,j,T}^{\text{mat}}], \quad (36)$$

where

$$D_{i,j}(1, t) = \exp \left(- \sum_{u=2}^t r_{i,j,u} \right) \quad (37)$$

is the discount factor from time t back to time 1. The three cash-flow components are

$$\begin{aligned} CF_{i,j,t}^d &= \sum_m Res_{m,t-1}^{(i,j)} N_{m,t}^{d,(i,j)}, \\ CF_{i,j,t}^s &= (1 - \pi) \sum_m Res_{m,t-1}^{(i,j)} N_{m,t}^{s,(i,j)}, \\ CF_{i,j,T}^{mat} &= \sum_m Res_{m,T}^{(i,j)} N_{m,T}^{(i,j)}. \end{aligned} \quad (38)$$

This definition makes explicit what is averaged in the nested simulation. For a point x_i , an inner Monte Carlo estimate based on M simulations is

$$\widehat{BEL}_{1,M}(x_i) = \frac{1}{M} \sum_{j=1}^M PV_{i,j}. \quad (39)$$

The annual simulation order is the same on every inner path: the financial states are advanced, the guarantee is reset when the year is a reset date, the profit-sharing rate and reserve growth are determined, mortality and surrender decrements are calculated, the cash flows are paid, and the surviving reserve is rolled forward. At maturity, the remaining reserve is paid to surviving policies.

5.3.3 Fitting scenarios

The proxy is fitted to the map

$$x \mapsto BEL_1(x)$$

or, in the extension of Section 5.7, to $x \mapsto FDB_1(x)$. The fitting domain is first constructed from 20,000 real-world scenarios. For each of the seven drivers, the observed minimum and maximum are widened by 25%. A scrambled Sobol design is then used to generate 100,000 space-filling points inside the resulting seven-dimensional box. This follows the principle that fitting scenarios should cover the relevant risk-factor domain rather than reproduce only the central part of the physical distribution [2, 3].

At each fitting point, only five inner simulations are used:

$$\tilde{Y}_i = \frac{1}{5} \sum_{j=1}^5 PV_{i,j}, \quad (40)$$

where Y denotes the chosen target. These values are referred to below as Monte Carlo fitting targets. The small number of inner simulations introduces inner-simulation Monte Carlo error, which the regression is expected to smooth across a large number of outer fitting points. The sensitivity of this choice is examined in Section 5.6.2.

5.3.4 Validation and capital calculation

Global validation uses 256 independent scrambled Sobol points generated inside the same fitting box but with a different Sobol seed. The use of $256 = 2^8$ keeps the low-discrepancy construction complete. Each point is valued with 1,000 inner simulations. These values are not used to fit or select the final model after the tuning stage; they are used to measure out-of-sample prediction error.

For each driver, a one-dimensional validation sweep is constructed by evaluating 60 equally spaced points across its fitting interval while holding the other six drivers at the midpoint of their respective intervals. The choice of 60 points is a numerical resolution choice intended to provide a sufficiently detailed view of the proxy shape; these sweeps are used as graphical diagnostics and do not enter model fitting or hyperparameter selection. One driver is varied through its fitting interval while the six other drivers are fixed at the midpoint of their respective intervals. These graphs do not replace the seven-dimensional validation cloud. Their role is to check that the complete proxy response remains reasonable when one risk factor is moved across the fitting domain.

The reference SCR is based on 100,000 real-world outer scenarios and 1,000 inner simulations for every outer scenario. A separate random seed is used for this final reference calculation. The proxy SCR used to measure approximation error is evaluated on exactly the same 100,000 outer states, with the same first-year asset returns and cash flows. Consequently, the difference between the reference SCR and a common-scenario proxy SCR cannot be attributed to the use of a different number of outer scenarios.

Two subsets of the reference loss distribution are then used for more focused validation. The adverse region contains all observations satisfying

$$L_i^{\text{ref}} \geq q_{0.99}(L^{\text{ref}}). \quad (41)$$

With 100,000 outer scenarios this gives 1,000 adverse observations. The capital region is the narrower band

$$q_{0.993}(L^{\text{ref}}) \leq L_i^{\text{ref}} \leq q_{0.997}(L^{\text{ref}}), \quad (42)$$

centred on the 99.5% SCR quantile. Because empirical quantiles are observed order statistics, 401 scenarios satisfy the inclusive bounds in the final sample. The target values of both subsets are recomputed with 2,000 inner simulations.

5.3.5 Numerical budget and BEL_0 convergence

Table 5 summarises the final numerical design. The nested reference is processed in batches of 100 outer states for memory reasons; batching changes only the implementation and not the estimator.

Table 5: Final numerical simulation and validation budget

Quantity	Final value
Outer fitting scenarios	100,000
Inner simulations per fitting point	5
Fitting-range buffer	25%
Independent global validation points	256
Inner simulations per global validation point	1,000
One-dimensional sweep points per driver	60
Common real-world scenarios for reference and proxy SCR	100,000
Inner simulations per reference outer scenario	1,000
Adverse-region scenarios	1,000
Capital-region scenarios	401
Inner simulations for adverse/capital-region revaluation	2,000

A separate convergence experiment was performed for BEL_0 in order to assess the inner simulation budget. For each number of inner simulations, 20 independent repetitions were compared with a 50,000-path numerical benchmark. Table 6 reports the values most relevant for the choice of inner paths.

Table 6: BEL_0 inner Monte Carlo convergence

Inner paths	Mean BEL_0	Standard deviation	Mean abs. rel. error	95% abs. rel. error
200	252.389 M	3.878 M	1.281%	2.657%
1,000	253.676 M	1.334 M	0.433%	0.933%
5,000	253.448 M	0.728 M	0.227%	0.474%

The 50,000-path benchmark is 253.407 million. The move from 200 to 1,000 inner simulations materially reduces both the standard deviation and the relative error. At 1,000 inner simulations, the 95th percentile of the absolute relative error is below 1%. For this reason, 1,000 inner simulations are retained for the global validation and the final nested reference. The benchmark is used only as a numerical reference and is not interpreted as the exact analytical value of the liability.

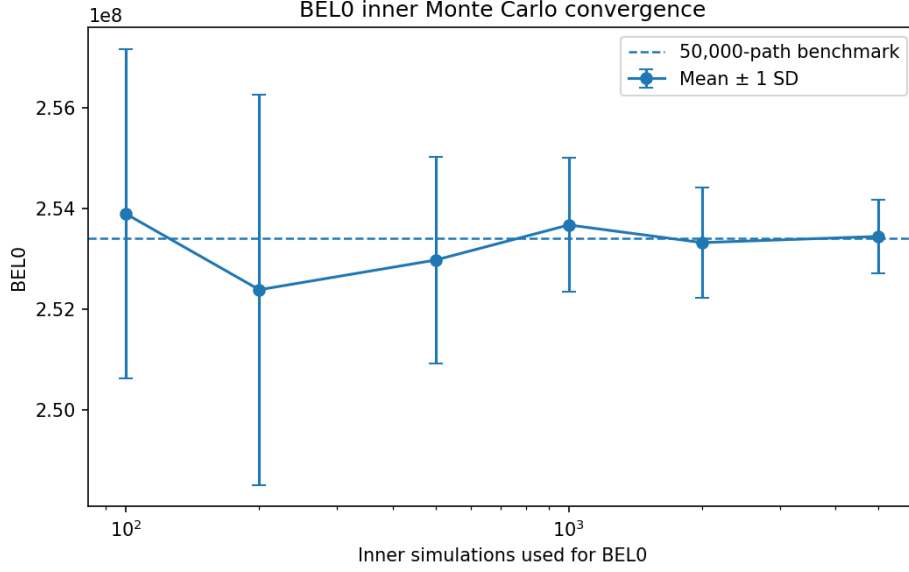


Figure 1: BEL_0 convergence as the number of inner simulations increases. The points show the mean over 20 repetitions and the vertical bars show one standard deviation.

5.3.6 Performance measures

For accurate values y_i and proxy predictions $\hat{y}_i$, the global root mean squared error is

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2}. \quad (43)$$

The range-normalised RMSE is

$$NRMSE = \frac{RMSE}{\max_i y_i - \min_i y_i}. \quad (44)$$

The signed average bias is

$$Bias = \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i). \quad (45)$$

The same measures are calculated separately on the adverse and capital regions.

For the capital requirement, the signed relative error is

$$RE_{SCR}^{\text{signed}} = \frac{\widehat{SCR} - SCR^{\text{ref}}}{SCR^{\text{ref}}}. \quad (46)$$

A negative value means that the proxy underestimates the reference SCR, while a positive value means that it overestimates it. The sign is retained because the direction of the capital error is important.

5.4 Implementation of the Proxy Models

5.4.1 Polynomial LSMC benchmark

The LSMC benchmark is a total-degree Legendre polynomial regression,

$$\hat{Y}^{LSMC}(x) = \sum_{m=1}^M \alpha_m \phi_m(x). \quad (47)$$

Legendre polynomials are used on the bounded fitting domain because the orthogonal construction is numerically more stable than an unscaled power basis [3]. The selected degree is four. With seven risk drivers, the complete total-degree basis contains

$$M = \binom{7+4}{4} = 330 \quad (48)$$

terms, including the intercept. No AIC stepwise selection is applied in the final comparison.

5.4.2 Explainable Boosting Machine

The EBM takes the form

$$\hat{Y}^{EBM}(x) = \beta_0 + \sum_{j=1}^7 f_j(x_j) + \sum_{(j,k) \in \mathcal{I}} f_{jk}(x_j, x_k), \quad (49)$$

where the f_j are learned main-effect functions and the f_{jk} are selected pairwise interaction functions [4]. With seven risk drivers, at most $\binom{7}{2} = 21$ distinct pairs are available. Therefore, an EBM specification with ten interactions means that up to ten two-variable interaction terms are selected; it does not mean a ten-way interaction.

The InterpretML implementation also contains inner and outer bagging parameters. Outer bags correspond to repeated bagged EBM fits whose contributions are averaged, which improves stability. Inner bags are resamples used inside the boosting procedure when estimating individual terms. The final BEL_1 EBM uses 10 pairwise interactions, 0 inner bags and 25 outer bags. For the FDB_1 target, 15 pairwise interactions are selected while the bagging parameters remain unchanged. Setting the number of inner bags to zero disables this additional internal resampling. These parameters belong to the EBM fitting algorithm and are unrelated to the outer and inner simulations of the nested Monte Carlo framework.

5.4.3 LocalGLMnet

LocalGLMnet retains the additive local-coefficient form developed in Chapter 3,

$$\hat{Y}^{LGN}(x) = \beta_0 + \sum_{k=1}^7 \beta_k(x) x_k. \quad (50)$$

The coefficients $\beta_k(x)$ are generated by a feed-forward neural network, allowing the local effects to vary with the complete state vector [5, 7]. The final architecture

contains two hidden layers with 32 neurons each and hyperbolic tangent activation. Training uses Adam, a learning rate of 3×10^{-4} , a batch size of 512, a maximum of 400 epochs and early stopping with patience 30. No ElasticNet penalty is used in the experiment.

5.4.4 Hyperparameter selection

Hyperparameters are selected using out-of-sample validation NRMSE, not the SCR error. This separation is important because selecting a model directly on the final capital quantile would use the quantity that is subsequently reported as a performance measure. The fitting sample contains 100,000 Sobol points with five inner simulations, while the independent validation cloud contains 256 points valued with 1,000 inner simulations. The full tuning grids and repeated neural-network runs are reported in the Appendix.

The BEL_1 and FDB_1 targets are tuned separately because they are different functions. Table 7 gives the final specifications.

Table 7: Final proxy specifications

Model	BEL_1 target	FDB_1 target
LSMC	Legendre degree 4	Legendre degree 4
EBM	10 interactions, 0/25 inner/outer bags	15 interactions, 0/25 inner/outer bags
LocalGLMnet	$(32, 32)$, tanh, $lr = 3 \cdot 10^{-4}$, no penalty	$(32, 32)$, tanh, $lr = 3 \cdot 10^{-4}$, no penalty

5.5 Main Experiment: Full Time-1 Best Estimate Liability

5.5.1 Global validation accuracy

The first experiment uses

$$Y(X_1) = BEL_1(X_1)$$

as the proxy target. The final common-scenario nested reference SCR is

$$SCR^{\text{ref}} = 26.6034 \text{ M.} \quad (51)$$

Table 8 reports the results on the independent 256-point validation cloud. All three models produce a high R^2 . LocalGLMnet has the lowest RMSE and NRMSE, followed by polynomial LSMC. EBM is less accurate globally, although its R^2 remains above 0.994.

Table 8: Global validation results for the BEL_1 target

Model	RMSE	NRMSE	R^2	Bias
LSMC–Legendre degree 4	3.777 M	0.780%	0.99807	-0.103 M
EBM	6.143 M	1.269%	0.99489	0.054 M
LocalGLMnet	3.130 M	0.646%	0.99867	0.103 M

The result confirms that the more flexible LocalGLMnet architecture can improve the global approximation of the seven-dimensional liability. At the same

time, degree-four LSMC remains a strong benchmark despite its fixed polynomial structure.

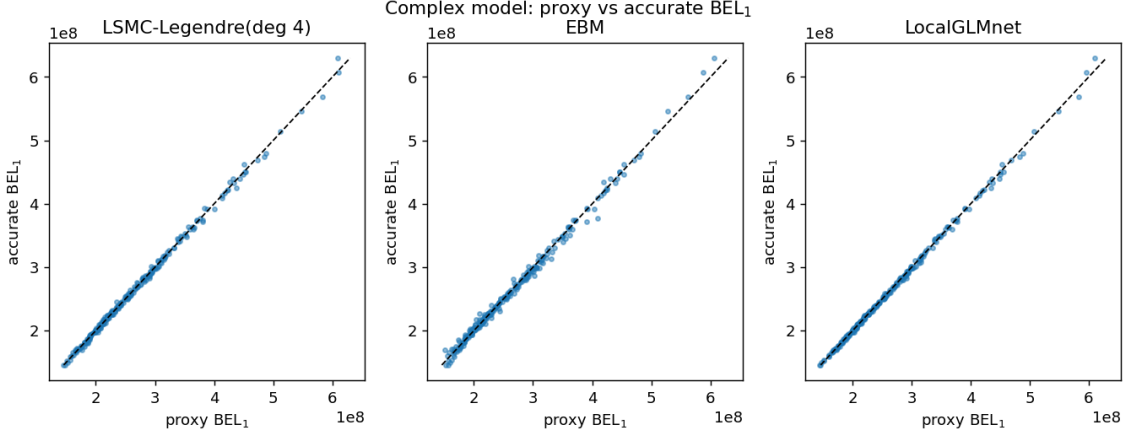


Figure 2: Predicted versus accurate BEL_1 values on the independent validation cloud.

5.5.2 One-dimensional validation sweeps

The one-dimensional sweeps provide a direct view of how the total fitted liability changes when each driver is varied. The rate level and curve slope produce strong decreasing relationships because higher rates increase discounting and reduce the value of existing guarantees. The portfolio-return curve is nonlinear because profit sharing becomes more important as investment performance improves. The lapse relationship is strongly decreasing, since a higher behavioural lapse state reduces the number of policies that remain in force.

The volatility and credit effects are smaller in scale but remain visible, while the mortality effect is comparatively weak in the one-dimensional slice. These plots should not be interpreted as causal marginal effects because the complete seven-dimensional function contains interactions. Their role is to compare the total proxy response with accurate valuations along controlled slices of the fitting domain.

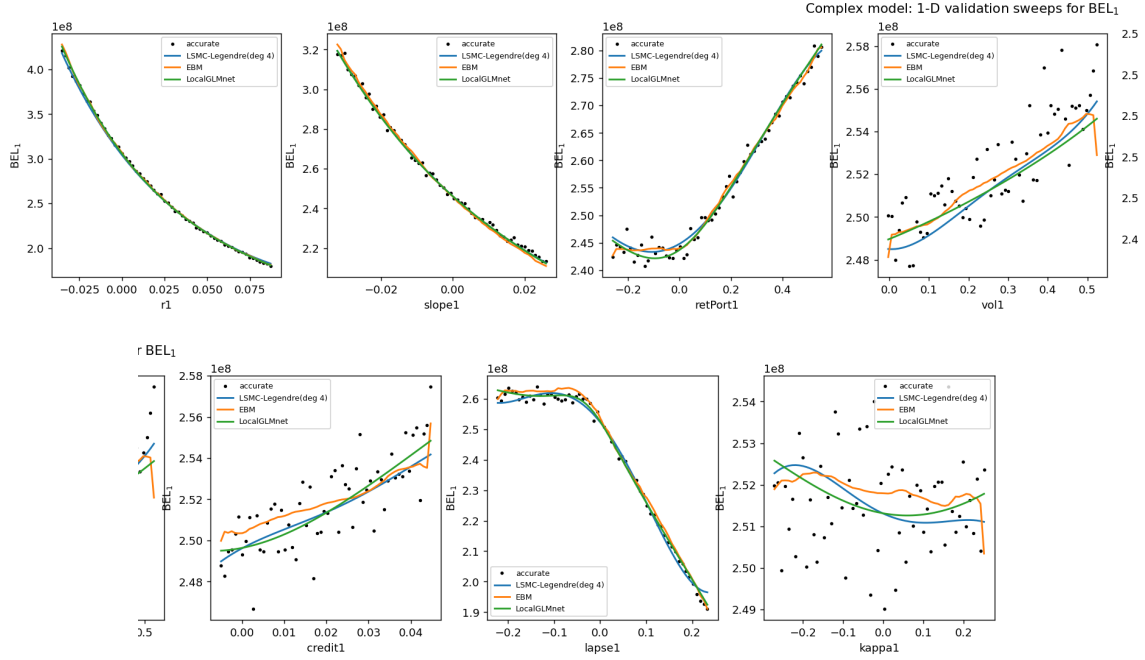


Figure 3: One-dimensional validation sweeps for the seven BEL_1 risk drivers. Black points are valuations obtained with 1,000 inner simulations.

5.5.3 Adverse and capital-region validation

Table 9 reports the two focused validation regions. In the final reference sample, LocalGLMnet gives the lowest NRMSE in both the full 99% adverse region and the narrower capital region. EBM is slightly more accurate than LSMC in the adverse region and very close to LSMC in the capital band.

Table 9: Adverse- and capital-region validation for the BEL_1 target

Model	Adverse NRMSE	Adverse bias	Capital NRMSE	Capital bias
LSMC-Legendre degree 4	1.630%	-0.079 M	1.644%	0.005 M
EBM	1.534%	0.783 M	1.628%	0.801 M
LocalGLMnet	1.457%	-0.851 M	1.443%	-0.827 M

The signed bias is important for interpreting the subsequent SCR. LSMC has almost zero average bias in the capital band, whereas EBM has a positive bias and LocalGLMnet a negative bias of similar magnitude. The model with the lowest capital-region NRMSE does not therefore necessarily have the smallest error in the empirical 99.5% quantile.

5.5.4 SCR accuracy

Table 10 compares the proxy SCRs with the common nested reference. LSMC produces the closest final SCR, with an error of +0.78%. LocalGLMnet underestimates the reference by 2.67%, while EBM overestimates it by 3.65%.

Table 10: Common-scenario SCR comparison for the BEL_1 target

Model	Proxy SCR	Reference SCR	Signed relative error
LSMC–Legendre degree 4	26.810 M	26.603 M	+0.78%
EBM	27.575 M	26.603 M	+3.65%
LocalGLMnet	25.892 M	26.603 M	-2.67%

This ranking differs from the global and capital-region NRMSE ranking. LocalGLMnet is the most accurate pointwise model in the final validation sets, but the nearly zero capital-region bias of LSMC gives the closest empirical quantile in this particular reference sample. This illustrates why SCR validation cannot be replaced by a single global prediction metric.

5.6 Robustness and Sensitivity Analysis

5.6.1 EBM interactions and bagging

The EBM tuning confirms that pairwise interactions are important in the seven-dimensional liability. For the BEL_1 target, validation performance improves strongly when interactions are added and reaches a plateau around ten selected pairs. For FDB_1 , the best tested interaction count is fifteen(15). Additional interactions do not improve the validation NRMSE sufficiently to justify a larger specification.

The bagging refinement shows that outer bagging improves stability, while additional inner bagging is not necessary in the final setting. This leads to 25 outer bags and zero inner bags for both targets. The complete tuning tables are reported in the Appendix. The choice is made from validation NRMSE rather than SCR performance.

5.6.2 Sensitivity to the number of inner simulations

The number of inner simulations used to construct the Monte Carlo fitting targets is varied over

$$n'_Q \in \{1, 2, 5, 10, 20, 50\}.$$

The experiment keeps the outer fitting size fixed and re-estimates each proxy. Table 11 reports the global and capital-region NRMSE for three representative budgets.

Table 11: Sensitivity to the number of inner simulations used for fitting

Model	1 inner		5 inner		50 inner	
	Global	Capital	Global	Capital	Global	Capital
LSMC	1.072%	2.486%	0.675%	2.399%	0.635%	2.300%
EBM	1.372%	2.513%	1.296%	2.067%	1.238%	1.454%
LocalGLMnet	0.933%	4.297%	0.615%	3.090%	0.418%	1.567%

Five inner simulations provide a strong cost–accuracy compromise for LSMC and EBM. LocalGLMnet continues to improve more materially when the Monte Carlo fitting targets become more accurate. The choice of five inner simulations should therefore not be interpreted as full numerical convergence for LocalGLMnet; it is a

common low-inner budget used to compare the three proxy architectures under the same simulation cost.

The SCR error is not monotonic in the number of inner simulations. Improving the average conditional valuation does not guarantee that the small number of scenarios determining the empirical quantile is improved in the same direction.

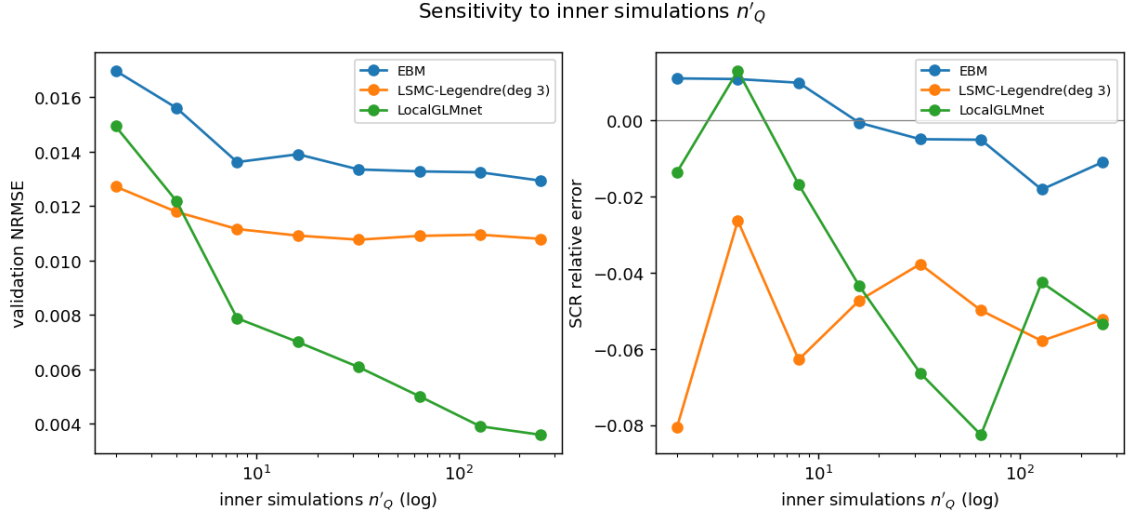


Figure 4: Sensitivity to the number of inner simulations used for the Monte Carlo fitting targets.

5.6.3 Sensitivity to the number of fitting scenarios

The outer fitting size is varied from 5,000 to 100,000 while the inner budget is kept fixed. Global NRMSE decreases for all three models as the space-filling design becomes denser.

Table 12: Global NRMSE as the number of fitting scenarios increases

Outer fitting points	LSMC	EBM	LocalGLMnet
5,000	1.495%	1.732%	1.407%
10,000	1.101%	1.523%	1.137%
20,000	0.983%	1.463%	1.010%
50,000	0.757%	1.313%	0.758%
100,000	0.675%	1.275%	0.670%

This experiment supports the use of 100,000 fitting points in the final comparison. However, the SCR error again varies non-monotonically. A smaller fitting sample can occasionally produce a close SCR by chance even when its global approximation is worse. The final fitting size is therefore selected from the general improvement in validation accuracy, not from a single favourable SCR value.

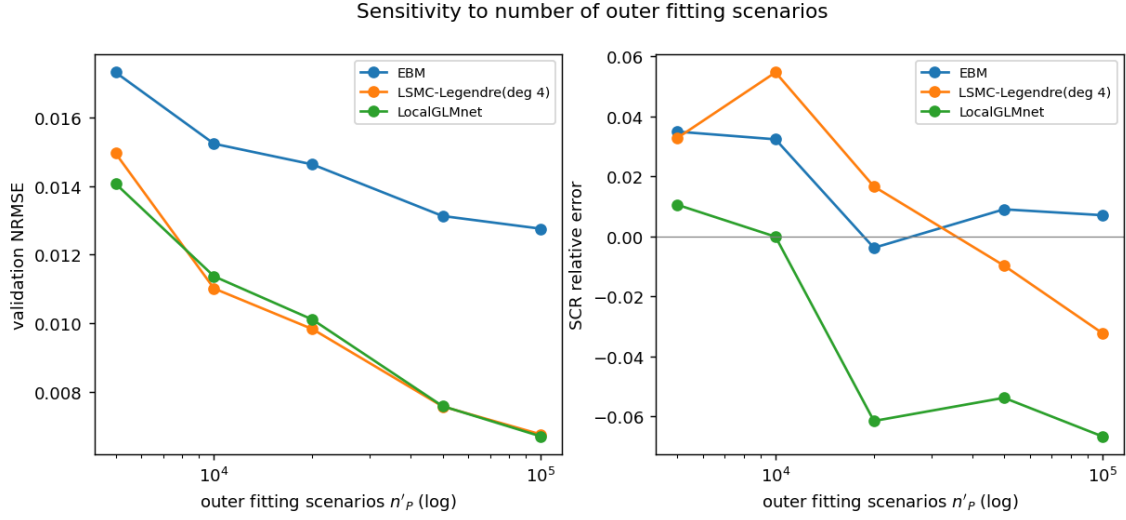


Figure 5: Sensitivity to the number of outer fitting scenarios.

5.6.4 Stability across random repetitions

The complete fitting procedure is repeated ten times with different random seeds. The sensitivity experiment uses a reduced but still accurate nested benchmark of 20,000 outer scenarios and 1,000 inner simulations. Its absolute SCR level is therefore not used as a replacement for the final 100,000-by-1,000 reference. The purpose of the repetitions is to compare the variability of the proxies under repeated fitting.

Table 13: Stability over ten fitting repetitions

Model	Mean global NRMSE	Mean capital NRMSE	Mean SCR error	SCR coeff var
LSMC-degree 4	0.702%	2.437%	-3.35%	1.66%
EBM	1.300%	1.918%	-0.39%	1.12%
LocalGLMnet	0.639%	2.377%	-5.52%	2.24%

The repetitions confirm that LocalGLMnet has the strongest average global accuracy and that LSMC is also very stable globally. EBM, however, has the lowest mean capital-region NRMSE, the smallest average signed SCR error and the lowest SCR coefficient of variation. This result is complementary to the single final holdout, where LSMC happens to produce the closest SCR. EBM appears less dependent on a favourable local cancellation of errors around one particular empirical quantile.

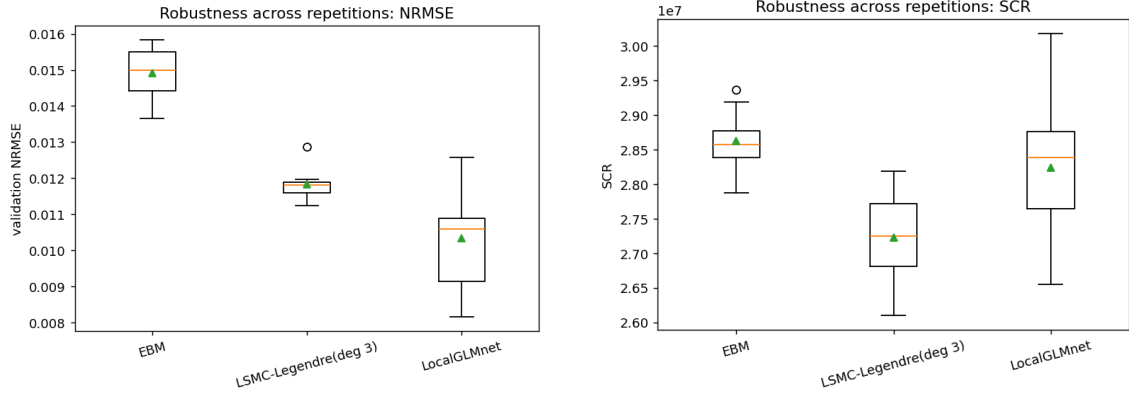


Figure 6: Global NRMSE and SCR estimates over ten fitting repetitions.

5.6.5 Economic parameter sensitivity

A one-at-a-time sensitivity analysis is also performed on three liability assumptions: the profit-sharing participation coefficient β , the margin d in the profit-sharing formula and the dynamic surrender sensitivity γ . The proxy hyperparameters are kept fixed, but the proxies are re-fitted under each alternative liability specification.

Table 14: Alternative liability assumptions used in the sensitivity analysis

Scenario	β	d	γ
Base	0.30	1.5%	0.80
Lower profit-sharing participation	0.20	1.5%	0.80
Higher profit-sharing participation	0.40	1.5%	0.80
Lower profit-sharing margin	0.30	1.0%	0.80
Higher profit-sharing margin	0.30	2.0%	0.80
Lower surrender sensitivity	0.30	1.5%	0.40
Higher surrender sensitivity	0.30	1.5%	1.20

The reference BEL_0 changes materially with the profit-sharing assumptions, ranging from approximately 247.0 million to 264.9 million. By contrast, the reference SCR remains between approximately 26.76 million and 27.05 million across the tested specifications. LocalGLMnet retains the lowest global NRMSE in most of the scenarios, with LSMC very close. EBM has the lowest capital-region NRMSE in every tested specification, between 1.77% and 2.01%, and its mean absolute SCR error across the seven scenarios is approximately 0.74%. The corresponding values are approximately 2.61% for LSMC and 5.77% for LocalGLMnet.

These results do not imply that EBM is universally superior for capital estimation. They show that, under the alternative liability assumptions considered here, its capital-region error is comparatively stable. The sensitivity analysis therefore strengthens the need to report both pointwise accuracy and capital accuracy instead of deriving model selection from one metric.

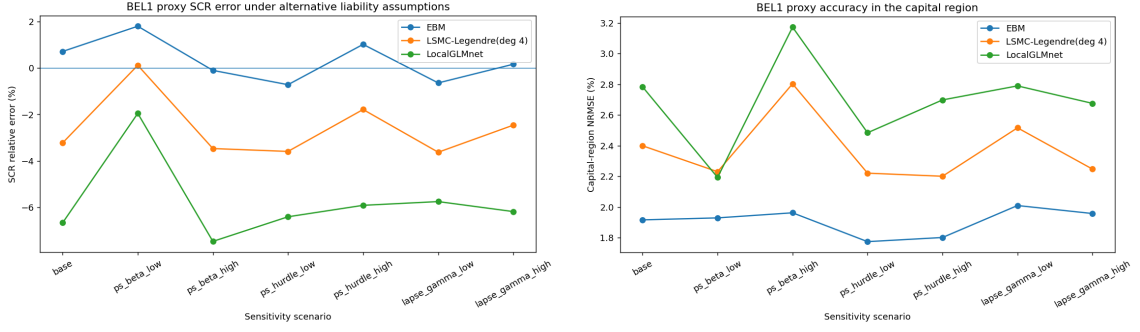


Figure 7: SCR error and capital-region NRMSE under alternative liability assumptions.

5.7 Extension: Future Discretionary Benefits

5.7.1 Actuarial motivation

Solvency II requires future discretionary benefits to be identified separately within technical provisions. In the life-insurance decomposition discussed by Perla et al. [7], the liability can be separated into a guaranteed component(GB) and an option-like discretionary component(FDB). This motivates a second proxy target:

$$FDB_1(x) = BEL_1(x) - GB_1(x). \quad (52)$$

The purpose of the experiment is to test whether proxying this smaller option-like component and reconstructing the full liability can improve the capital estimate.

5.7.2 Construction of GB_1 and FDB_1

In the present implementation, GB_1 is not defined as a deterministic cash-flow component. This distinction is important because the contract still contains stochastic future guaranteed-rate resets, mortality, surrender behaviour, discount rates, volatility and credit spreads.

For each time-1 state and each inner scenario, two reserve tracks are projected with the same random numbers. The full reserve follows

$$Res_{m,t}^{full} = Res_{m,t-1}^{full}(1 + i_t^g + PS_t), \quad (53)$$

whereas the parallel GB reserve follows

$$Res_{m,t}^{GB} = Res_{m,t-1}^{GB}(1 + i_t^g). \quad (54)$$

All other simulated quantities are shared between the two books. In particular, the future guaranteed rate is still reset from the simulated eight-year yield, mortality and surrender remain stochastic, and the same discount factors are used. The only mechanism removed from the GB projection is future profit sharing.

Both reserve tracks start from the same time-1 reserve. Consequently, any profit sharing already credited during the first projection year is vested at time 1 and belongs to both starting states. The decomposition therefore isolates discretionary

benefits to be credited after the risk horizon, rather than rewriting the year-1 balance sheet.

For M common inner simulations,

$$\begin{aligned} GB_1(x_i) &= \frac{1}{M} \sum_{j=1}^M PV_{i,j}^{GB}, \\ BEL_1(x_i) &= \frac{1}{M} \sum_{j=1}^M PV_{i,j}^{full}, \\ FDB_1(x_i) &= \frac{1}{M} \sum_{j=1}^M \left(PV_{i,j}^{full} - PV_{i,j}^{GB} \right). \end{aligned} \quad (55)$$

Using common random numbers reduces the simulation error of the difference.

For the final SCR comparison, GB_1^{ref} is calculated once on the common nested reference scenarios. If a proxy predicts $\widehat{FDB}_1^{(m)}$, the corresponding full liability is reconstructed as

$$\widehat{BEL}_1^{(m)} = GB_1^{ref} + \widehat{FDB}_1^{(m)}. \quad (56)$$

The same GB_1^{ref} is used for LSMC, EBM and LocalGLMnet. Therefore,

$$BEL_1^{ref} - \widehat{BEL}_1^{(m)} = FDB_1^{ref} - \widehat{FDB}_1^{(m)},$$

so the proxy error in this controlled experiment comes only from the approximation of the discretionary component.

The adverse and capital regions are not redefined from the distribution of FDB_1 . They are the same scenarios selected from the full reference one-year loss distribution in Equations (41) and (42). This keeps the validation linked to the scenarios that determine the insurer's SCR.

5.7.3 Global and capital-region results

The direct approximation of FDB_1 is more difficult globally than the approximation of the complete BEL_1 . Table 15 shows that LocalGLMnet again gives the lowest global NRMSE, 1.423%. LSMC and EBM are close, at 1.624% and 1.681%, respectively.

Table 15: Global validation results for the FDB_1 target

Model	RMSE	NRMSE	R^2	Bias
LSMC–Legendre degree 4	0.585 M	1.624%	0.99597	-0.020 M
EBM	0.605 M	1.681%	0.99568	-0.013 M
LocalGLMnet	0.512 M	1.423%	0.99690	-0.072 M

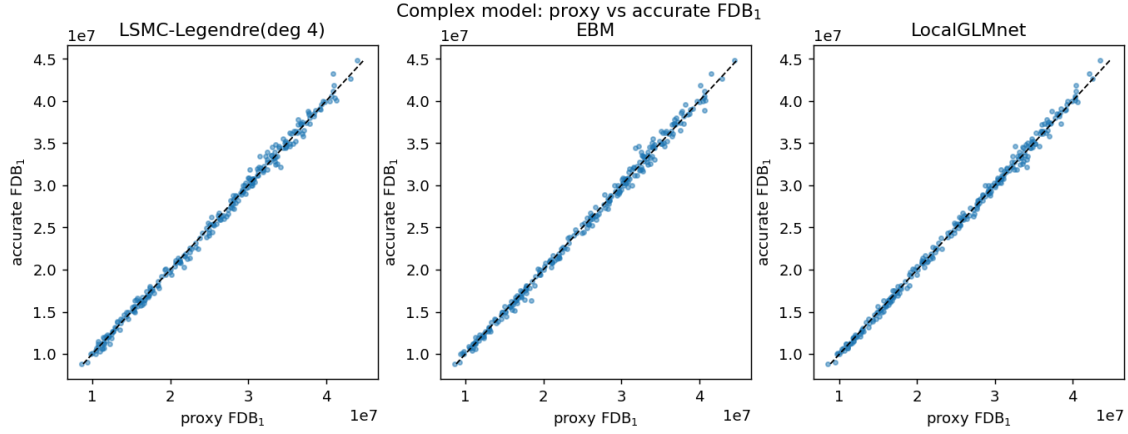


Figure 8: Predicted versus accurate FDB_1 values on the independent validation cloud.

The ranking changes in the adverse and capital regions. EBM gives the lowest NRMSE in both regions and also the smallest absolute capital-region bias.

Table 16: Adverse- and capital-region validation for the FDB_1 target

Model	Adverse NRMSE	Adverse bias	Capital NRMSE	Capital bias
LSMC-Legendre degree 4	1.922%	-0.152 M	1.894%	-0.122 M
EBM	1.475%	-0.074 M	1.551%	-0.044 M
LocalGLMnet	2.155%	-0.348 M	2.140%	-0.331 M

The reconstructed SCRs are reported in Table 17. All three FDB-based proxies underestimate the reference, but the errors are small. EBM is the closest at -0.33% , followed by LSMC at -0.51% and LocalGLMnet at -1.29% .

Table 17: SCR estimates obtained from the FDB_1 proxy target

Model	Reconstructed proxy SCR	Reference SCR	Signed relative error
LSMC-Legendre degree 4	26.468 M	26.603 M	-0.51%
EBM	26.515 M	26.603 M	-0.33%
LocalGLMnet	26.260 M	26.603 M	-1.29%

5.7.4 Comparison of the BEL_1 and FDB_1 proxy targets

The FDB target has a higher global NRMSE for each model because it isolates a smaller, more nonlinear component. Nevertheless, the reconstructed capital estimate is closer to the reference for all three proxies.

Table 18: Comparison of SCR errors under the two proxy targets

Model	Direct BEL_1 target	FDB_1 target with BEL_1 reconstruction
LSMC-Legendre degree 4	+0.78%	-0.51%
EBM	+3.65%	-0.33%
LocalGLMnet	-2.67%	-1.29%

This result does not mean that the FDB function is statistically easier to learn. The global results show the opposite. Instead, it shows that separating the liability target can change how approximation error is distributed around the capital quantile. The guaranteed projection is supplied from the common nested reference, while the proxy focuses only on the incremental effect of future profit sharing. In this experiment, this leads to a smaller SCR error for all three model classes.

5.8 Interpretability and Actuarial Validation

The EBM is particularly useful for examining how the fitted liability is decomposed across the seven drivers. The plots below are internal model contributions rather than causal effects. They should therefore be read together with the out-of-sample validation results.

5.8.1 EBM main effects for the full liability

Figure 9 shows the main-effect functions of the final BEL_1 EBM. The short-rate level and yield-curve slope have large decreasing contributions. Higher rates increase discounting and reduce the value of guarantees. The portfolio-return effect is non-linear and increases when the return becomes sufficiently favourable for future profit sharing. The volatility contribution also increases over most of the range.

The behavioural lapse factor has a large decreasing contribution. This is consistent with higher surrender activity reducing the number of policies that remain exposed to future guaranteed and discretionary payments. The credit-spread contribution is smaller than the main rate and lapse effects but remains visible because the spread affects the corporate bond return. The mortality-index effect is comparatively small and less regular.

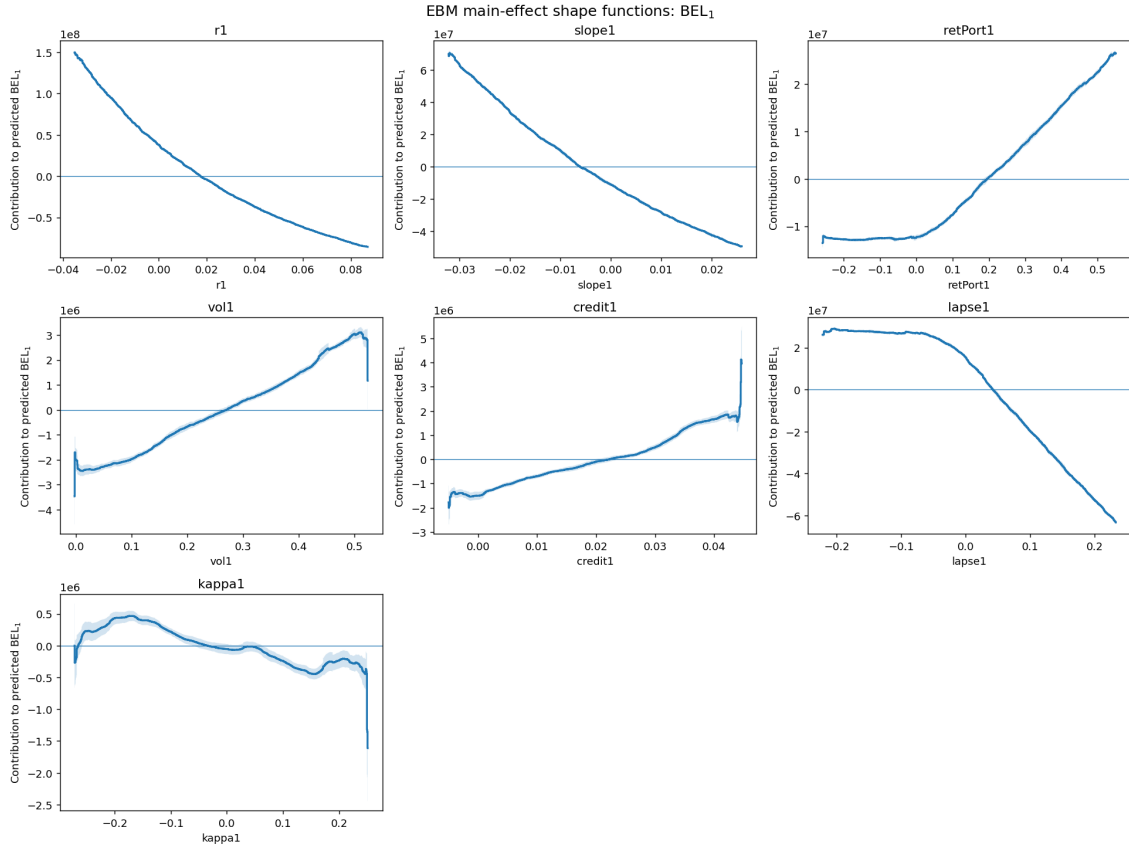


Figure 9: EBM main-effect functions for the BEL_1 target.

5.8.2 Term importance and selected interactions

The term-importance ranking in Figure 10 is dominated by r_1 , followed by $slope_1$ and $lapse_1$. The interaction $r_1 \times slope_1$ is the most important pairwise term, while interactions between interest rates and lapse also appear among the largest components. This is consistent with the model structure: discounting, guarantee reset and surrender behaviour are not independent channels.

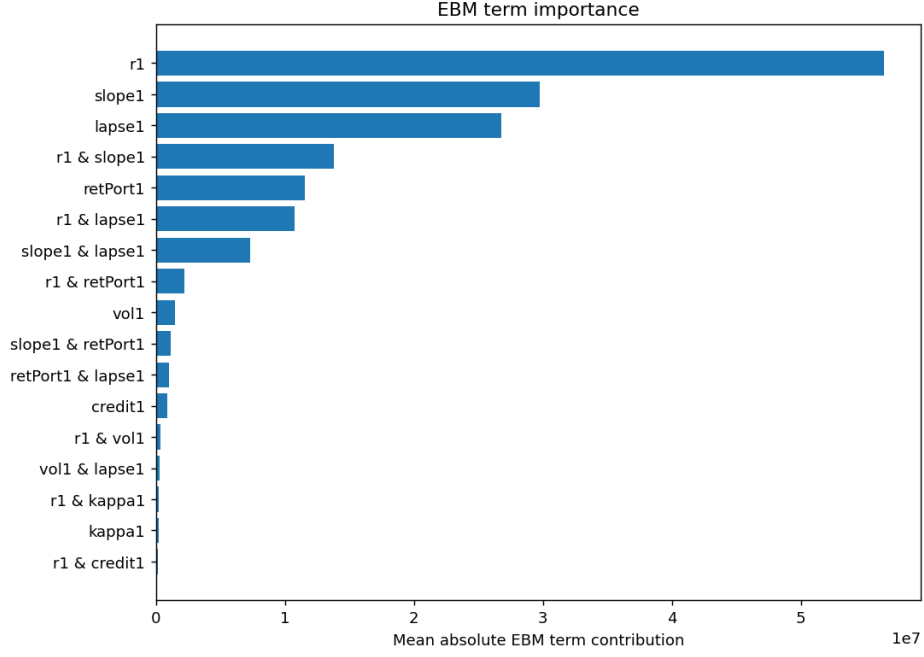


Figure 10: Importance of EBM main effects and selected pairwise interactions for the BEL_1 target.

The three leading pairwise interaction surfaces are shown in Figure 11. They provide a direct explanation for why an EBM without interactions is too restrictive for this liability. The surfaces are centred components of the additive prediction and should not be read as independent stress tests.

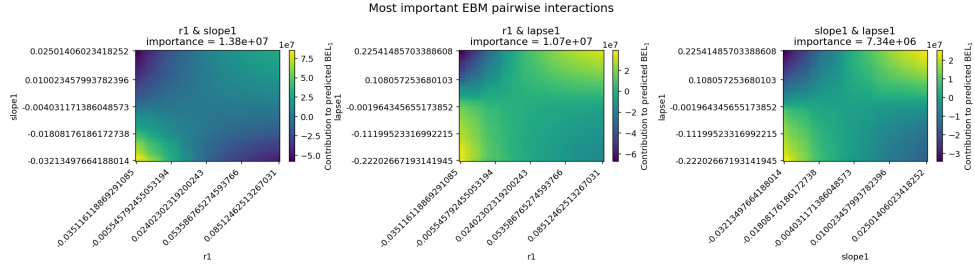


Figure 11: Three leading EBM pairwise interaction surfaces for the BEL_1 target.

5.8.3 Interpretability of the FDB target

The importance ranking changes materially when the target is FDB_1 . Figure 12 shows that the behavioural lapse state becomes the dominant term. Volatility is second, followed by the first-year portfolio return, credit spread and the short-rate level. The leading interactions include $r_1 \times lapse_1$, $r_1 \times vol_1$ and $vol_1 \times lapse_1$.

This change is consistent with the meaning of the target. Future discretionary benefits depend strongly on whether policies remain in force long enough to receive future crediting and on the investment conditions that determine whether profit sharing becomes positive. The ranking should be interpreted as a property of this numerical portfolio and target, not as a universal ordering of insurance risk factors.

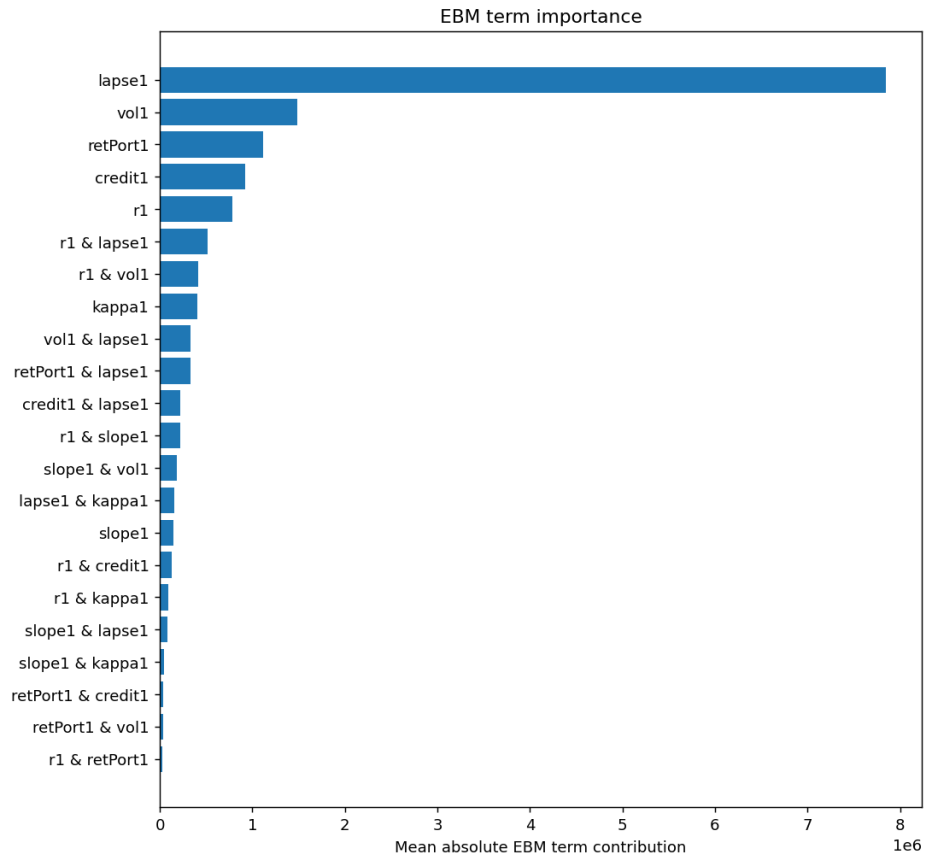


Figure 12: Importance of EBM main effects and selected pairwise interactions for the FDB_1 target.

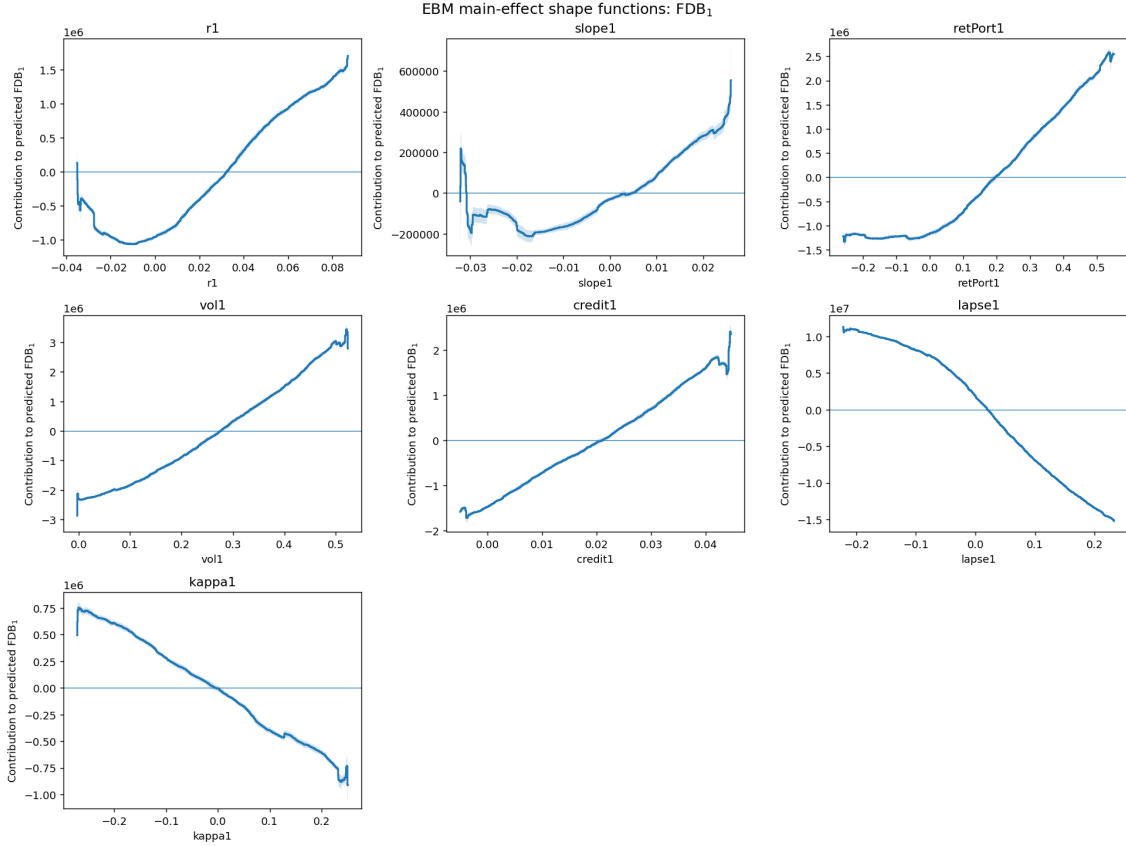


Figure 13: EBM main-effect functions for the FDB_1 target.

5.8.4 Interpretability as a model-risk control

Interpretability and predictive validation answer different questions. EBM shape functions show how the fitted model decomposes its own prediction, while the Sobol validation cloud and one-dimensional sweeps compare the complete prediction with the nested valuation. Neither replaces the capital-region checks. A smooth and economically reasonable fitted shape can still lead to a biased empirical quantile, and a close SCR in one sample can result from offsetting errors. For this reason, the interpretation plots are considered together with global NRMSE, regional NRMSE, signed bias and repeated SCR results.

5.9 Discussion and Limitations

The numerical results lead to several conclusions.

First, polynomial LSMC remains a strong benchmark. With a degree-four Legendre basis, the seven-dimensional model contains 330 terms and reaches a global NRMSE of 0.780% in the final holdout. It also gives the closest direct- BEL_1 SCR in the final common reference sample. This shows that a well-specified polynomial proxy can remain competitive even when the number of candidate interactions has increased substantially.

Second, LocalGLMnet provides the strongest pointwise accuracy for the complete liability. It has the lowest global, adverse and capital-region NRMSE in the

final BEL_1 experiment. Its performance also improves materially as the inner fitting targets become more accurate. However, its capital bias is negative in the final hold-out and across the repeated sensitivity experiment. The best global approximation therefore does not automatically imply the best empirical 99.5% quantile.

Third, EBM shows a different strength. It is less accurate globally than the other two models in the final holdout, but it is the most stable capital proxy in the repeated and economic sensitivity experiments. Its pairwise terms also give a direct view of the interactions responsible for this behaviour. The final model therefore illustrates the value of separating average predictive accuracy from stability in the region relevant for capital.

Fourth, the FDB experiment changes the comparison. FDB_1 is harder to approximate globally, but reconstructing BEL_1 from a common GB_1 and the FDB proxy reduces the absolute SCR error for all three models in the final experiment. EBM gives the closest FDB-based SCR and the lowest FDB capital-region NRMSE, while LocalGLMnet remains best globally. The proxy target is therefore itself an important modelling choice.

Several limitations remain. The portfolio, asset allocation and management rules are stylised and the economic parameters are illustrative rather than calibrated to a specific market date or insurance portfolio. The credit and stochastic-volatility processes use the same transition laws in outer and inner projections, since no separate risk premia are introduced for these states. The mortality process is Lee–Carter-style but is not estimated from historical mortality data. The dynamic surrender rule is also illustrative. These choices are acceptable for the controlled model-comparison objective of the thesis, but they limit direct transfer of the numerical levels to a production internal model.

The reference SCR is a large nested Monte Carlo benchmark rather than an exact value. The BEL_0 convergence study and the use of 100,000 common outer scenarios with 1,000 inner simulations reduce this numerical uncertainty, but do not remove it completely. The one-dimensional sweeps also cannot show all seven-dimensional interactions. Finally, the hyperparameter search is intentionally limited to a manageable set of specifications. The detailed tuning and parameter-sensitivity tables are therefore reported in the Appendix rather than interpreted as exhaustive optimisation.

Within these limits, the common experiment gives a controlled comparison of how the three interpretable proxy architectures allocate their approximation error across the full scenario space and the capital region.

5.10 Conclusion

This chapter implemented polynomial LSMC, EBM and LocalGLMnet for a stylised Universal Life portfolio exposed to seven financial and actuarial risk drivers. The financial model contains two rate factors, stochastic equity volatility and a credit-spread process. The liability model contains path-dependent guaranteed-rate resets, profit sharing, dynamic surrenders and stochastic mortality. The outer one-year scenario is generated under the real-world dynamics, while the remaining liability is valued conditionally with the valuation dynamics.

The final BEL_1 experiment does not produce one model that dominates every measure. LocalGLMnet gives the lowest global, adverse and capital-region NRMSE. Degree-four LSMC gives the closest SCR in the final common reference sample. EBM gives the most stable capital behaviour across repeated fittings and alternative liability assumptions, while also providing direct shape and interaction explanations.

The FDB extension gives an additional result. The discretionary component is more difficult to approximate globally, but the reconstructed SCR is closer to the reference for all three models. In that experiment, EBM gives the smallest capital-region error and the closest SCR, while LocalGLMnet remains the most accurate global FDB proxy.

The main conclusion is therefore not that one interpretable machine learning method universally replaces classical LSMC. The three models have different strengths, and these strengths become visible only when validation includes both the complete scenario space and the region that determines the capital requirement. For a Solvency II proxy model, global fit, signed bias, capital-region accuracy, stability and interpretability should therefore be considered together.

6 Conclusion

This thesis investigated whether interpretable machine learning models could provide useful extension to the polynomial Least-Squares Monte Carlo for the estimation of a Solvency II capital requirement. The motivation comes from the computational difficulty of nested simulation in life insurance. LSMC reduces this cost by replacing repeated market-consistent valuations with a regression proxy, but the usual polynomial specification may become difficult to adapt when the liability depends on several nonlinear and interacting risk factors.

Three proxy models were therefore compared: a Legendre polynomial LSMC model,

an Explainable Boosting Machine and LocalGLMnet. The comparison was carried out on a stylised Universal Life portfolio represented by seven financial and actuarial risk drivers. All models were trained and evaluated using common simulation designs, and their predictions were compared with the same nested Monte Carlo reference.

The first question was if the more flexible interpretable models could improve the approximation of the liability. For the direct approximation of the full time-1 best estimate liability, LocalGLMnet achieved the best pointwise predictive performance. Its global NRMSE was 0.646 %, compared with 0.780 % for polynomial LSMC and 1.269 % for EBM. LocalGLMnet also obtained the lowest errors in the adverse and capital regions. This confirms that the additional flexibility of the model can be useful when the liability depends on several nonlinear risk factors.

The second question was if better predictive accuracy automatically produces a better SCR approximation. To begin with, the nested reference SCR was approximately EUR 26.60 million. In the direct liability experiment, LSMC produced the closest approximation of the reference SCR with a relative error of approximately +0.78 %, compared with -2.67 % for LocalGLMnet and +3.65 % for EBM. Nevertheless, LocalGLMnet had the lowest capital-region RMSE and LSMC had almost no average bias around the capital quantile. This illustrates that the location and direction of proxy errors matter for an extreme quantile and not only their average magnitude. Thus, predictive accuracy does not automatically produce a better SCR approximation.

The stability evaluations provide an additional perspective. Across repeated fittings, LocalGLMnet remained the strongest model in terms of global prediction accuracy, whereas EBM showed the strongest stability of the capital estimate. EBM also performed well in the capital region when selected profit-sharing and surrender assumptions were modified. These results suggest that global accuracy, capital accuracy and stability should be considered as

separate validation criteria.

The last question concerned the choice of proxy target. In the second experiment, future discretionary benefits were approximated separately and the full liability was reconstructed using the corresponding guaranteed component. The discretionary-benefit target was more difficult to approximate globally, but the reconstructed SCR was closer to the reference for all three models. The relative SCR errors became approximately -0.51 % for LSMC, -0.33 % for EBM and -1.29 % for LocalGLMnet. In

this experiment, EBM also produced the best capital-region results. The comparison therefore shows that the choice of what is approximated can be as important as the choice of regression method itself.

No single model can consequently be identified as the best proxy according to every criterion in this capital settings. Polynomial LSMC acts as a strong benchmark and produced a very accurate capital estimate despite its simpler functional form. LocalGLMnet offered the best overall predictive accuracy, while EBM provided a useful combination of nonlinear flexibility, direct interpretability as well as capital stability. The EBM shape functions also made it possible to examine how individual risk factors and selected interactions affected the estimated liability.

The numerical study nevertheless has several limitations. The Universal Life portfolio and a number of financial, behavioural and mortality parameters are stylised rather than calibrated to a specific insurer. The mortality model is a simplified stochastic extension of a Gompertz–Makeham baseline, and the credit model represents spread movements without modelling defaults or rating transitions separately. The conclusions should therefore be interpreted as a comparison of proxy methods within a controlled numerical framework rather than as a calibration study for a particular insurance portfolio.

To go deeper, it could be useful to apply the same comparison to an insurer-specific data and calibrated economic and actuarial assumptions, study a higher-dimensional capital settings where the limitations of polynomial bases may become even more pronounced and to investigate simulation designs that allocate additional valuation effort specifically around the capital region.

The conclusion is then that the quality of a solvency proxy cannot be judged from a single average prediction error. What matters for evaluating solvency capital is the behaviour of the proxy around the capital quantile, stability, the choice of the liability quantity being approximated, and its interpretability. Machine Learning models which are interpretable can therefore be a valuable extension of the classical LSMC framework for the regression step.

References

- [1] Anthony Floryszczak, Olivier Le Courtois, and Mohamed Majri. 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, 2016.
- [2] Anne-Sophie Krah. *Least-Squares Monte Carlo Methods in the Life Insurance Sector*. Dissertation, Technische Universit"at Kaiserslautern, 2021.
- [3] Adam Koursaris. A least squares monte carlo approach to liability proxy modelling and capital calculation. Technical report, Barrie & Hibbert, September 2011.
- [4] Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. Interpretml: A unified framework for machine learning interpretability, September 2019. Preprint.
- [5] Ronald Richman and Mario V. W"uthrich. Localglmnet: Interpretable deep learning for tabular data. *Scandinavian Actuarial Journal*, 2023(1):71–95, 2023. doi: 10.1080/03461238.2022.2081816.
- [6] Directive 2009/138/ec of the european parliament and of the council of 25 november 2009 on the taking-up and pursuit of the business of insurance and reinsurance (solvency ii) (recast). Official Journal of the European Union, L 335, 17 December 2009, pp. 1–155, 2009. CELEX: 32009L0138.
- [7] Francesca Perla, Salvatore Scognamiglio, Andrea Spadaro, and Paolo Zanetti. Explainable least square monte carlo for solvency capital requirement evaluation, February 2025. Preprint, version of February 22, 2025.
- [8] Francis A. Longstaff and Eduardo S. Schwartz. Valuing american options by simulation: A simple least-squares approach. *The Review of Financial Studies*, 14(1):113–147, 2001. doi: 10.1093/rfs/14.1.113.
- [9] Anthony Floryszczak, Olivier Le Courtois, and Mohamed Majri. Inside the solvency 2 black box: Net asset values and solvency capital requirements with a least-squares monte-carlo approach. In *ASTIN, AFIR/ERM and IACA Colloquia*, Sydney, Australia, August 2015. Conference paper presented to the Actuaries Institute.
- [10] Anne-Sophie Krah, Zoran Nikolić, and Ralf Korn. Machine learning in least-squares monte carlo proxy modeling of life insurance companies. *Risks*, 8(1): 21, 2020. doi: 10.3390/risks8010021.
- [11] Ronald Richman and Mario V. W"uthrich. Lasso regularisation within the localglmnet architecture. In *Actuarial Society of South Africa's 2022 Convention*, Cape Town, South Africa, October 2022. Conference paper.

- [12] Marketa Krupova, Nabil Rachdi, and Quentin Guibert. Explainable boosting machine for predicting claim severity and frequency in car insurance. *arXiv preprint arXiv:2503.21321*, 2025.
- [13] Brandon M. Greenwell, Annika Dahlmann, and Saurabh Dhoble. Explainable boosting machines with sparsity: Maintaining explainability in high-dimensional settings, November 2023. Preprint.

A Appendix

A.1 Numerical parameterisation

A.1.1 Purpose of the parameterisation

The numerical experiment of Chapter 4 is designed as a controlled model-comparison exercise rather than as the replication of the internal model of a specific insurer. Consequently, the parameters should not all be interpreted in the same way. Three categories are distinguished.

First, some parameters define the stylised portfolio itself, such as the model points, the initial guaranteed rate, the projection horizon and the strategic asset allocation. Second, the stochastic-process parameters are illustrative values chosen to generate economically meaningful and sufficiently persistent variations in the seven risk drivers. Third, a small number of parameters, in particular the initial surplus ratio and the profit-sharing specification, were selected jointly to keep the initial balance sheet solvent while preserving a non-negligible discretionary option. No statistical calibration to a historical data set or to a specific market date is claimed.

This distinction is important for the interpretation of the results. The thesis compares the approximation properties of LSMC, EBM and LocalGLMnet conditional on one common ALM model. The numerical level of the SCR is therefore specific to this parameterisation, whereas the comparison of the proxy methods is the main object of interest.

A.1.2 Profit-sharing moneyness and initial solvency

The profit-sharing rule is

$$PS_t = \max \left\{ \beta \left(R_t^{\text{Port}} - i_t^g - d \right), 0 \right\}.$$

At inception, $i_0^g = 1.5\%$ and $d = 1.5\%$. Profit sharing therefore becomes positive when the annual portfolio return exceeds approximately

$$i_0^g + d = 3.0\%.$$

The parameter $\beta = 30\%$ controls the amount credited once this threshold is exceeded. The purpose of the selected pair (β, d) is not to reproduce a particular management rule observed in data. It is to keep the discretionary component active in a meaningful part of the simulated state space without making it dominate the guaranteed liability.

The initial surplus ratio is set to 20%, giving

$$A_0 = 318.0 \text{ M}$$

for an initial mathematical reserve of 265.0 M. Together with the profit-sharing parameters, this produces a positive initial net asset value in the numerical model. The economic sensitivity experiment in Appendix A.5 varies β , d and the surrender sensitivity γ around their base values, so the main proxy conclusions are not based only on one liability specification.

A.2 Random seeds

The main experiment uses the common numerical budget reported in Chapter 4. For reproducibility, the main fitting seed is

20240601,

while the final nested reference uses the separate seed

20260809.

A.3 Proxy Hyperparameter Tuning

A.3.1 Tuning protocol

All proxy hyperparameters are selected before the final SCR comparison. The candidate models are fitted on the same 100,000 Sobol fitting points and evaluated on the same independent 256-point Sobol validation cloud. The fitting values use five inner simulations per point, whereas the validation values use 1,000 inner simulations. Model selection is based on validation NRMSE and not on the final SCR error.

This separation prevents the final capital benchmark from being used as a tuning criterion. It also follows the general approach in the proxy-modelling literature, where model architecture and regression specification are selected on independent validation data. For the neural-network model, several training seeds are used because the fitted network depends on random initialisation and stochastic optimisation.

The BEL_1 and FDB_1 targets are tuned separately because they are different conditional valuation functions.

A.3.2 Polynomial LSMC

Table 19: LSMC degree tuning for the BEL_1 target

Degree	Terms	Validation NRMSE	R^2	Bias	Fit time (s)
1	8	6.859%	0.85061	+0.029 M	0.1
2	36	2.424%	0.98134	+0.038 M	0.3
3	120	1.029%	0.99664	-0.019 M	1.5
4	330	0.780%	0.99807	-0.103 M	5.3

Table 20: LSMC degree tuning for the FDB_1 target

Degree	Terms	Validation NRMSE	R^2	Bias	Fit time (s)
1	8	5.751%	0.94946	-0.015 M	0.1
2	36	3.713%	0.97893	-0.014 M	0.4
3	120	1.816%	0.99496	-0.022 M	1.5
4	330	1.624%	0.99597	-0.020 M	4.9

A.3.3 Explainable Boosting Machine

The EBM tuning is carried out in two stages. The first stage controls the number of pairwise interactions. With seven risk factors there are 21 possible pairs. The second stage adjusts the bagging configuration after the interaction count has been chosen. As explained in Chapter 3, these inner and outer bags are parameters of the EBM algorithm and are unrelated to the inner and outer scenarios of the nested Monte Carlo framework..

Table 21: EBM interaction-count tuning for the BEL_1 target

Requested pairs	Fitted pairs	Inner bags	Outer bags	NRMSE	R^2	Time (s)
0	0	8	8	5.652%	0.89856	32.4
5	5	8	8	1.331%	0.99437	229.5
10	10	8	8	1.317%	0.99450	296.0
15	15	8	8	1.338%	0.99432	364.2
21	21	8	8	1.325%	0.99442	496.1

Table 22: EBM bagging refinement for the BEL_1 target with ten interactions

Inner bags	Outer bags	NRMSE	R^2	Bias	Time (s)
0	8	1.291%	0.99471	+0.040 M	120.7
0	14	1.277%	0.99482	+0.057 M	213.2
0	25	1.269%	0.99489	+0.054 M	446.2

Table 23: EBM interaction-count tuning for the FDB_1 target

Requested pairs	Fitted pairs	Inner bags	Outer bags	NRMSE	R^2	Time (s)
0	0	0	25	3.726%	0.97878	43.2
5	5	0	25	1.976%	0.99403	154.0
10	10	0	25	1.741%	0.99537	232.1
15	15	0	25	1.681%	0.99568	274.4
21	21	0	25	1.706%	0.99555	320.2

Table 24: EBM bagging comparison for the FDB_1 target with fifteen interactions

Inner bags	Outer bags	NRMSE	R^2	Bias	Time (s)
0	25	1.681%	0.99568	-0.013 M	274.4
0	8	1.685%	0.99566	-0.014 M	88.7
8	8	1.715%	0.99551	-0.016 M	433.1
8	14	1.684%	0.99567	-0.015 M	609.0
20	14	1.683%	0.99567	-0.016 M	1430.5
20	25	1.683%	0.99567	-0.015 M	2608.0

A.3.4 LocalGLMnet

LocalGLMnet is first tuned over a compact set of architectures and activation functions. Each architecture is trained three times on the same fitting and validation

data. The reported standard deviation therefore reflects the network-training randomness rather than a change in the data set.

Table 25: LocalGLMnet architecture screen for the BEL_1 target

Hidden layers	Activation	Mean NRMSE	NRMSE s.d.	Mean R^2	Mean time (s)
$32 \times 32 \times 32 \times 32$	tanh	0.740%	0.058%	0.99825	61.8
32×32	tanh	0.745%	0.012%	0.99824	42.5
$64 \times 64 \times 64$	tanh	0.803%	0.061%	0.99794	74.7
$32 \times 32 \times 32 \times 32$	sigmoid	0.847%	0.063%	0.99772	163.9
$32 \times 32 \times 32 \times 32$	relu	0.884%	0.130%	0.99748	50.9

The four-layer 32-unit tanh network has the lowest mean NRMSE in the first screen, but the difference from the two-layer 32-unit tanh network is very small. The two-layer model has a substantially smaller NRMSE standard deviation and a lower mean fit time. It is therefore retained.

The second stage fixes the architecture to $(32, 32)$ with tanh activation and varies the learning rate and a small ElasticNet penalty. The penalty is written

$$\eta \left[(1 - \alpha) \|\beta(x)\|_2^2 + \alpha \|\beta(x)\|_1 \right].$$

When $\eta = 0$, no regularisation is applied.

Table 26: LocalGLMnet learning-rate and penalty refinement for the BEL_1 target

Learning rate	Penalty	Mean NRMSE	NRMSE s.d.	Mean R^2	Mean time (s)
0.0003	$\eta = 1e - 05, \alpha = 0.5$	0.710%	0.031%	0.99840	94.7
0.0003	$\eta = 1e - 06, \alpha = 0.5$	0.710%	0.031%	0.99840	105.2
0.0003	none	0.710%	0.031%	0.99840	72.6
0.001	none	0.809%	0.051%	0.99792	62.2
0.001	$\eta = 1e - 06, \alpha = 0.5$	0.809%	0.051%	0.99792	74.8
0.001	$\eta = 1e - 05, \alpha = 0.5$	0.809%	0.050%	0.99792	70.3

Table 27: LocalGLMnet architecture screen for the FDB_1 target

Hidden layers	Activation	Mean NRMSE	NRMSE s.d.	Mean R^2	Mean time (s)
32×32	tanh	1.384%	0.061%	0.99707	30.3
$64 \times 64 \times 64$	tanh	1.431%	0.018%	0.99687	49.3
$32 \times 32 \times 32 \times 32$	tanh	1.432%	0.046%	0.99686	56.9
$32 \times 32 \times 32 \times 32$	relu	1.491%	0.059%	0.99660	34.4
$32 \times 32 \times 32 \times 32$	sigmoid	1.621%	0.160%	0.99596	225.2

A.4 Final selected hyperparameters

Table 28: Final proxy hyperparameters used in Chapter 4

Model	BEL_1 target	FDB_1 target
LSMC	Legendre degree 4, 330 terms	Legendre degree 4, 330 terms
EBM	10 pairs, 0 inner bags, 25 outer bags	15 pairs, 0 inner bags, 25 outer bags
LocalGLMnet	$(32, 32)$, tanh, $lr = 3 \times 10^{-4}$, $\eta = 0$	$(32, 32)$, tanh, $lr = 3 \times 10^{-4}$, $\eta = 0$

A.5 Additional Numerical Sensitivity Results

A.5.1 BEL_0 convergence

The convergence analysis reported in Chapter 4 is reproduced here with the complete set of simulation budgets. For every budget, 20 independent estimates are compared with a 50,000-path numerical benchmark of 253.407 M. The benchmark is a high-precision Monte Carlo value and is not treated as an analytical truth.

Table 29: Complete BEL_0 inner-simulation convergence results

Inner paths	Mean BEL_0	S.d.	Mean rel. error	Mean abs. rel. error	95% abs. rel. error
100	253.896 M	3.270 M	0.193%	1.114%	1.975%
200	252.389 M	3.878 M	-0.402%	1.281%	2.657%
500	252.979 M	2.057 M	-0.169%	0.665%	1.442%
1,000	253.676 M	1.334 M	0.106%	0.433%	0.933%
2,000	253.325 M	1.092 M	-0.032%	0.330%	0.887%
5,000	253.448 M	0.728 M	0.016%	0.226%	0.474%

A.5.2 Number of inner simulations used for fitting

The complete inner-simulation sensitivity is shown in Table 30. In this experiment, the nested sensitivity benchmark uses 20,000 reference outer scenarios and 1,000 inner simulations. Its reference SCR is approximately 26.975 M and should not be confused with the final 100,000-by-1,000 reference SCR of 26.603 M.

Table 30: Complete sensitivity to the number of inner simulations used for fitting

n'_Q	LSMC			EBM			LocalGLMnet		
	Global NRMSE	Capital NRMSE	SCR error	Global NRMSE	Capital NRMSE	SCR error	Global NRMSE	Capital NRMSE	SCR error
1	1.072%	2.486%	-3.50%	1.372%	2.513%	-3.79%	0.933%	4.297%	-10.38%
2	0.864%	2.380%	-2.93%	1.354%	2.556%	+1.58%	0.782%	3.053%	-7.02%
5	0.675%	2.399%	-3.23%	1.296%	2.067%	-1.20%	0.614%	3.090%	-7.89%
10	0.661%	2.336%	-3.53%	1.282%	1.876%	-0.48%	0.544%	2.137%	-5.18%
20	0.651%	2.614%	-5.47%	1.247%	1.773%	-2.24%	0.473%	2.684%	-7.31%
50	0.635%	2.300%	-3.57%	1.238%	1.454%	-1.36%	0.418%	1.567%	-2.79%

A.5.3 Number of outer fitting scenarios

Table 31: Complete sensitivity to the number of outer fitting scenarios

n'_p	LSMC			EBM			LocalGLMnet		
	Global NRMSE	Capital NRMSE	SCR error	Global NRMSE	Capital NRMSE	SCR error	Global NRMSE	Capital NRMSE	SCR error
5,000	1.495%	3.031%	+3.28%	1.731%	2.873%	+3.50%	1.407%	2.579%	+1.06%
10,000	1.101%	3.134%	+5.47%	1.523%	2.697%	+3.24%	1.137%	2.690%	-0.01%
20,000	0.983%	2.405%	+1.67%	1.463%	2.075%	-0.38%	1.010%	3.243%	-6.15%
50,000	0.757%	2.398%	-0.97%	1.313%	2.022%	+0.91%	0.758%	2.970%	-5.38%
100,000	0.675%	2.399%	-3.23%	1.275%	1.916%	+0.71%	0.669%	2.782%	-6.67%

A.5.4 Stability across fitting repetitions

Table 32: Summary of the ten-seed stability experiment

Model	Mean global NRMSE	NRMSE s.d.	Mean adverse NRMSE	Mean capital NRMSE	Mean capital bias	Mean SCR	SCR s.d.	Mean SCR error	SCR CV
LSMC	0.702%	0.029%	2.007%	2.437%	-0.752 M	26.072 M	0.434 M	-3.35%	1.66%
EBM	1.300%	0.037%	1.706%	1.918%	+0.152 M	26.870 M	0.300 M	-0.39%	1.12%
LocalGLMnet	0.639%	0.048%	1.942%	2.377%	-1.249 M	25.486 M	0.571 M	-5.52%	2.24%

Table 33: Detailed results of the ten fitting repetitions

Rep.	Model	Global NRMSE	Capital NRMSE	Capital bias	SCR	SCR error
1	LSMC	0.675%	2.399%	-0.724 M	26.105 M	-3.23%
1	EBM	1.296%	2.067%	-0.173 M	26.652 M	-1.20%
1	LocalGLMnet	0.614%	3.090%	-1.891 M	24.845 M	-7.89%
2	LSMC	0.738%	2.279%	-0.413 M	26.612 M	-1.35%
2	EBM	1.252%	1.844%	+0.089 M	26.812 M	-0.60%
2	LocalGLMnet	0.624%	2.058%	-1.089 M	25.724 M	-4.64%
3	LSMC	0.694%	2.549%	-0.988 M	25.750 M	-4.54%
3	EBM	1.295%	2.130%	+0.348 M	27.196 M	+0.82%
3	LocalGLMnet	0.599%	2.725%	-1.676 M	25.076 M	-7.04%
4	LSMC	0.674%	3.003%	-1.691 M	25.189 M	-6.62%
4	EBM	1.247%	2.131%	-0.540 M	26.252 M	-2.68%
4	LocalGLMnet	0.690%	2.190%	-0.936 M	25.789 M	-4.40%
5	LSMC	0.754%	2.155%	-0.387 M	26.262 M	-2.64%
5	EBM	1.317%	1.926%	+0.015 M	26.604 M	-1.37%
5	LocalGLMnet	0.701%	2.262%	-1.337 M	25.305 M	-6.19%
6	LSMC	0.674%	2.575%	-0.799 M	26.066 M	-3.37%
6	EBM	1.305%	1.791%	+0.372 M	27.041 M	+0.25%
6	LocalGLMnet	0.676%	3.196%	-2.068 M	24.663 M	-8.57%
7	LSMC	0.687%	2.134%	-0.203 M	26.705 M	-1.00%
7	EBM	1.327%	1.773%	+0.514 M	27.143 M	+0.62%
7	LocalGLMnet	0.688%	1.741%	-0.353 M	26.346 M	-2.33%
8	LSMC	0.727%	2.303%	-0.554 M	26.161 M	-3.02%
8	EBM	1.318%	2.108%	+0.333 M	26.831 M	-0.53%
8	LocalGLMnet	0.606%	2.823%	-1.749 M	25.153 M	-6.75%
9	LSMC	0.709%	2.479%	-0.797 M	26.066 M	-3.37%
9	EBM	1.370%	1.698%	+0.329 M	27.121 M	+0.54%
9	LocalGLMnet	0.557%	1.768%	-0.448 M	26.268 M	-2.62%
10	LSMC	0.691%	2.492%	-0.966 M	25.804 M	-4.34%
10	EBM	1.274%	1.711%	+0.232 M	27.051 M	+0.28%
10	LocalGLMnet	0.633%	1.920%	-0.949 M	25.686 M	-4.78%

A.5.5 Economic parameter sensitivity

Table 34: Detailed economic-parameter sensitivity results

Scenario	Model	β	d	γ	BEL_0	Global NRMSE	Capital NRMSE	Reference SCR	SCR error
Base	LSMC	0.30	1.5%	0.80	255.764 M	0.675%	2.399%	26.975 M	-3.23%
Base	EBM	0.30	1.5%	0.80	255.764 M	1.275%	1.916%	26.975 M	+0.71%
Base	LocalGLMnet	0.30	1.5%	0.80	255.764 M	0.669%	2.782%	26.975 M	-6.67%
Lower participation	LSMC	0.20	1.5%	0.80	247.047 M	0.668%	2.230%	26.760 M	+0.11%
Lower participation	EBM	0.20	1.5%	0.80	247.047 M	1.294%	1.929%	26.760 M	+1.80%
Lower participation	LocalGLMnet	0.20	1.5%	0.80	247.047 M	0.681%	2.192%	26.760 M	-1.96%
Higher participation	LSMC	0.40	1.5%	0.80	264.921 M	0.682%	2.803%	26.938 M	-3.47%
Higher participation	EBM	0.40	1.5%	0.80	264.921 M	1.243%	1.962%	26.938 M	-0.11%
Higher participation	LocalGLMnet	0.40	1.5%	0.80	264.921 M	0.652%	3.172%	26.938 M	-7.47%
Lower profit-sharing margin	LSMC	0.30	1.0%	0.80	258.958 M	0.677%	2.220%	26.857 M	-3.60%
Lower profit-sharing margin	EBM	0.30	1.0%	0.80	258.958 M	1.267%	1.774%	26.857 M	-0.72%
Lower profit-sharing margin	LocalGLMnet	0.30	1.0%	0.80	258.958 M	0.667%	2.483%	26.857 M	-6.41%
Higher profit-sharing margin	LSMC	0.30	2.0%	0.80	252.870 M	0.674%	2.200%	27.050 M	-1.79%
Higher profit-sharing margin	EBM	0.30	2.0%	0.80	252.870 M	1.277%	1.801%	27.050 M	+1.01%
Higher profit-sharing margin	LocalGLMnet	0.30	2.0%	0.80	252.870 M	0.672%	2.697%	27.050 M	-5.92%
Lower surrender sensitivity	LSMC	0.30	1.5%	0.40	255.454 M	0.667%	2.516%	26.999 M	-3.63%
Lower surrender sensitivity	EBM	0.30	1.5%	0.40	255.454 M	1.283%	2.009%	26.999 M	-0.65%
Lower surrender sensitivity	LocalGLMnet	0.30	1.5%	0.40	255.454 M	0.670%	2.789%	26.999 M	-5.76%
Higher surrender sensitivity	LSMC	0.30	1.5%	1.20	256.081 M	0.690%	2.247%	26.893 M	-2.46%
Higher surrender sensitivity	EBM	0.30	1.5%	1.20	256.081 M	1.275%	1.957%	26.893 M	+0.16%
Higher surrender sensitivity	LocalGLMnet	0.30	1.5%	1.20	256.081 M	0.666%	2.675%	26.893 M	-6.19%

