Gaussian Process Regression for Pricing Variable Annuities with Stochastic Volatility and Interest Rate

Ludovic Goudenège*

Andrea Molent†

Antonino Zanette‡

Abstract

In this paper we develop an efficient approach based on a Machine Learning technique which allows one to quickly evaluate insurance products considering stochastic volatility and interest rate. Specifically, following De Spiegeleer et al. [10], we apply Gaussian Process Regression to compute the price and the Greeks of a GMWB Variable Annuity. Starting from observed prices previously computed by means of a Hybrid Tree PDE approach for some known combinations of model parameters, it is possible to approximate the whole target function on a bounded domain. The regression algorithm consists of two main steps: algorithm training and evaluation. In particular, the first step is the most time demanding, but it needs to be performed only once, while the prediction step is very fast and it requires to be performed only when evaluating the function. The developed method, as well as for the calculation of prices and Greeks, can also be employed to compute the no-arbitrage fee, which is a common practice in the Variable Annuities sector. We consider three increasing complexity models, namely the Black-Scholes, the Heston and the Heston Hull-White models, which extend the sources of randomness up to consider stochastic volatility and stochastic interest rate together. Numerical experiments show that the accuracy of the estimated values is high, while the computational cost is much lower than the one required by a direct calculation with standard approaches. Finally, we stress out that the analysis is carried out for a GMWB annuity but it could be generalized to other insurance products. Machine Learning seems to be a very promising and interesting tool for insurance risk management.

Keywords: Gaussian Process Regression, GMWB, Price, Greeks, No-arbitrage fee, Stochastic volatility, Stochastic interest rate.

*Féderation de Mathématiques de CentraleSupélec - CNRS FR3487, France - ludovic.goudenege@math.cnrs.fr
†Dipartimento di Scienze Economiche e Statistiche, Università degli Studi di Udine, Italy - andrea.molent@uniud.it
‡Dipartimento di Scienze Economiche e Statistiche, Università degli Studi di Udine, Italy - antonino.zanette@uniud.it

1 Introduction

Variable Annuities are tax deferred investment contracts with insurance coverage. The popularity of such products has been steadily increasing in the past years and the sales in the United States in 2017 amounted to over $116 billions, which represents almost half of the total annuity sales in 2017 (Insured Retirement Institute [19]).

In this paper, we focus on a particular Variable Annuity, which leads the living benefit riders (see [26]): the Guaranteed Minimum Withdrawal Benefit (GMWB). The contract is initiated by making a lump sum payment which is then invested in risky assets, usually a mutual fund. The benefit base, or guarantee account balance, is initially set to the amount of the premium. The holder of the policy (PH) is entitled to withdraw a fixed amount at contract anniversaries, even if the risky account has declined to zero. The PH may withdraw more than the guaranteed amount upon payment of a penalty. When the contract reaches maturity, the PH receives a final payoff depending on the residual risky account and on the final base benefit and the contract ends.

To manage these products, insurance companies resort to hedging techniques, which require the valuation and the calculation of the Greeks for a large number of policies. These products are fairly standardized and once a GMWB is launched it may remain on company's books for decades (see [26]): each company promotes a limited number of products, which can however vary according to a set of parameters established during the sale. Therefore, the companies do similar calculations many times, but varying the considered parameters. Hence, it is decisive that the time needed for all of these calculations is as small as possible.

Researchers have engaged to develop new numerical techniques that are increasingly efficient and less computationally expensive. For example, Bacinello et al. [1] and Bauer et al. [2] consider Monte Carlo methods to price and compute the Greeks of GMWB products. Chen and Forsyth [7] and Donnelly et al. [12] introduce partial derivative equations, while Luo and Shevchenko [21, 22] apply Gauss–Hermite integration quadrature. Yang, Dai and Liu [9] and Costabile [8] propose pricing methods based on the use of trees. Later, Ignatieva et al. [18] consider the Fourier Space Time-Stepping algorithm. More recently, Goudenège et al. apply the Hybrid PDE method to GLWB and GMWB pricing and Greeks calculation [14, 15].

However, as these techniques are more and more efficient, calculation times remain non-negligible, especially when the stochastic model considered to represent the market dynamics includes many random factors, such as stochastic volatility ans stochastic interest rate.

The computations that insurance companies face depend on model parameters and contract parameters which are only few and vary in a small range. This context is similar to what happens in the derivatives world where there are many repeated evaluations of standardized products. To overcome these difficulties, an innovative solution to speed up derivative pricing has been proposed by De Spiegeleer et al. [10]. They suggest to apply Machine Learning techniques to predict the price of the derivatives from a training set made of observed prices for particular combinations of model parameters. In particular, they consider Gaussian Process Regression (GPR) which is a Bayesian non-parametric technique. All the information needed to compute a function (the price or the Greeks of a derivative) can be summarized in a training set, then the algorithm learns the function and it can be applied to make predictions.

In the last few years, Machine Learning techniques have been employed in the insurance field. Specifically, Gan [13] studies the pricing of a large portfolio of Variable Annuities in the Black-Scholes model using clustering and GPR. Unlike what De Spiegeleer et al. do, Gan considers fixed values for the market parameters (interest rate and volatility), having to repeat the procedure at each change in market conditions. Other applications of Machine Learning techniques in the insurance field are due to Deprez et al. [11] who employ regression tree boosting machine to model mortality. More recently, Wüthrich [27] employs Neural Networks to extend Mack's chain-ladder method.

In this paper we show that Machine Learning techniques can be applied in the Variable Annuities framework and in particular in the scope of GMWB products as far as pricing and Greeks calculation is considered. In particular, we consider three complexity-increasing models as stochastic framework, namely the Black-Scholes model, the Heston model and the Heston Hull-White model. We show that, in each of these models, it is possible to develop a regression method capable of predicting the prices and the Greeks of each policy, for each combination of market and contract parameters within the training range. The training procedure needs to be performed only once and the developed regression can be applied under various market conditions and for different GMWB contracts. Moreover we also show that GPR can be applied effectively to compute the no-arbitrage fee, that is the cost that makes the contract fair under a risk neutral probability. Numerical tests show that the computational time can be reduced of several factors with a small loss of accuracy. Moreover, the use of the BCD method (see Grippo and Sciandrone [16]) allows us to manage a large amount of input data, thus improving the numerical results.

The reminder of the paper is organized as follows. In Section 2 we present the GMWB contract and we explain how to evaluate it. In Section 3 we briefly review the GPR method and we explain how to apply it in the GMWB scope. In Section 4 we report the results of some numerical tests. Finally, Section 5 draws the conclusions.

2 The GMWB Contract and its Evaluation

In this Section we present the GMWB contract and the evaluation method. For a comprehensive treatment we refer to Chen and Forsyth [7] and to Goudenège et al. [15].

2.1 The GMWB Contract

At contract inception the PH pays with a lump sum the premium P to the insurance company. Then, the PH is entitled to perform withdrawals during each contract anniversary, starting from $t = 1$. When the contract reaches maturity at $t = T$, the PH performs the last withdrawal, he receives a final payoff and the contract ends.

The contract evolution is described by two state parameters, the account value (A_t, $A_0 = P$) and the base benefit (B_t, $B_0 = P$), which change during the contract life according to the underlying fund value S_t and to the PH withdrawals. Specifically, during the time between two consecutive contract anniversaries, the base benefit B_t does not change, while the account value A_t follows the same dynamics of S_t, with the exception that some fees, determined by the parameter α, are be subtracted from A_t:

$$dA_t = \frac{A_t}{S_t} dS_t - \alpha A_t dt. \tag{2.1}$$

During the contract anniversaries, the PH is entitled to withdraw from his account a minimum amount G which is usually equal to P/T. Let us denote $A_{t_i^{(-)}}, B_{t_i^{(-)}}$ the contract state variables just before the i-th contract anniversary occurs at time $t_i = i$, and $A_{t_i^{(+)}}, B_{t_i^{(+)}}$ the same state variables just after t_i occurs. Moreover, let W_i represent the amount withdrawn at time t_i, which is required to be non-negative and smaller than the base benefit $B_{t_i^{(-)}}$.

If the amount withdrawn satisfies $W_i \leq G$, then there is no penalty imposed, whereas, if $W_i > G$, a proportional penalty charge $\kappa (W_i - G)$ is imposed, which reduces the amount actually received by the

PH. Therefore, the PH may not receive all the money he withdraws from the risky asset account: let $f_i(W_i) : \left[0, B_{t_i^{(-)}}\right] \to \mathbb{R}$ be the function of W_i denoting the cash flow received by the PH due to the withdrawal at time t_i, that is given by the following expression:

$$f_i(W_i) = \begin{cases} W_i & \text{if } W_i \le G \\ W_i - \kappa(W_i - G) & \text{if } W_i > G. \end{cases}$$

The contract state variables just after the withdrawal are given by:

$$A_{t_i^{(+)}} = \max\left(A_{t_i^{(-)}} - W_i, 0\right)$$

and

$$B_{t_i^{(+)}} = B_{t_i^{(-)}} - W_i$$

After the last withdrawal, the PH receives the final payoff, which is worth

$$FP = \max\left(A_T, (1-\kappa) B_T\right), \tag{2.2}$$

and the contract terminates.

As far as the determination of W_i is concerned, we assume that the PH selects W_i in order to maximize his total wealth, that is

$$W_i = \underset{w_i \in \left[0, B_{t_i^{(-)}}\right]}{\operatorname{argmax}} \; \mathcal{V}\left(\max\left(A_{t_i^{(-)}} - w_i, 0\right), B_{t_i^{(-)}} - w_i, v_{t_i}, r_{t_i}, t_i^+\right) + f_i(w_i), \tag{2.3}$$

where v_{t_i} is the volatility of the underlying fund and r_{t_i} is the interest rate, which may change or be constant depending on the considered model.

Thus, the value $\mathcal{V}$ of the GMWB contract at time t is completely settled by the account value A, the base benefit B and, in case of stochastic volatility, by the volatility level v_t.

2.2 The Stochastic Model for the Underlying and the GMWB Evaluation

For seek of comparison we consider three possible stochastic models for the underlying fund S_t, namely the Black-Scholes model, the Heston model (Heston [17]) which provides stochastic volatility and the Heston Hull-White model (see Kammeyer and Kienitz [20]) which provides both stochastic volatility and stochastic interest rate.

The dynamics of S_t under a risk neutral probability in the Black-Scholes model is given by

$$dS_t = rS_t dt + \sigma S_t dZ_t,$$

where Z is a Brownian motion. As opposed to the Black-Scholes model, the Heston model includes stochastic volatility. In particular, the dynamics of S_t and of the volatility process v_t are given by

$$\begin{cases} dS_t &= rS_t dt + \sqrt{v_t} S_t dZ_t^S \\ dv_t &= k_v(\theta_v - v_t)\, dt + \omega_v \sqrt{v_t} dZ_t^v, \end{cases} \tag{2.4}$$

where Z^S and Z^v are Brownian motions such that $d\left\langle Z_t^S, Z_t^v \right\rangle = \rho_v dt$.

As opposed to the Black-Scholes and to the Heston model, the Heston Hull-White model includes both stochastic volatility and stochastic interest rate. In particular, the dynamics of S_t, of the interest rate process r_t, and of the volatility process v_t are given by

$$
\begin{cases}
dS_t & = r_t S_t dt + \sqrt{v_t} S_t dZ_t^S \\
dr_t & = k_r \left(\theta_r \left(t\right) - r_t\right) dt + \omega_r dZ_t^r, \\
dv_t & = k_v \left(\theta_v - v_t\right) dt + \omega_v \sqrt{v_t} dZ_t^v,
\end{cases}
\tag{2.5}
$$

where Z^S, Z^v and Z^r are Brownian motions such that $d\left\langle Z_t^S, Z_t^v\right\rangle = \rho_v dt$, $d\left\langle Z_t^S, Z_t^r\right\rangle = \rho_r dt$ and $d\left\langle Z_t^v, Z_t^r\right\rangle = 0$. Here $\theta_r\left(t\right)$ is a deterministic function which is completely determined by the market values of the zero-coupon bonds by calibration (see Brigo and Mercurio [6]). For seek of simplicity, we assume that the market price of a zero-coupon bond at time t with maturity $\bar{t}$ is given by $P^M\left(t, \bar{t}\right) = e^{-r_0\left(\bar{t}-t\right)}$, the so-called *flat curve* case.

The parameters which identify the Black-Scholes model are the interest rate r and the volatility parameter σ. The parameters which identify the Heston model are the interest rate r, the initial volatility v_0, the rate of mean reversion k_v, the long run variance θ_v, the volatility of volatility ω_v and the correlation between the Brownian motions ρ_v. The parameters which identify the Heston Hull-White model are the initial interest rate r_0, the rate of mean reversion k_r, the volatility of the interest rate ω_r, the correlation ρ_r, the initial volatility v_0, the rate of mean reversion k_v, the long run variance θ_v, the volatility of volatility ω_v and the correlation ρ_v.

There are multiple approaches to evaluate the GMWB contract and to compute the Greeks. As far as the Black-Scholes model is considered, we apply the finite difference PDE method already employed by Chen and Forsyth [7] in the GMWB framework. As far as the Heston model is considered, we apply the Hybrid PDE method, introduced by Briani et al. [4] and already employed by Goudenège et al. [15] in the GMWB framework. In short, the Hybrid PDE method exploits a recombining tree to describe the possible evolution of the volatility process and solves a two-dimensional PDE at each node, assuming the value of the volatility as locally constant. With regard to the Heston Hull-White model, we consider a multidimensional version of the Hybrid PDE method, introduced by Briani et al. [5] never applied before in the Variable Annuities framework. In short, this extension uses a multidimensional tree to describe the volatility and interest rate processes together and solves a two-dimensional PDE at each node, assuming the values of the volatility and of the interest rate as locally constant.

3 Gaussian Process Regression for GMWB

3.1 Gaussian Process Regression

In this Section, we present a brief review of Gaussian Process Regression and for a comprehensive treatment we refer to Rasmussen and Williams [25].

Gaussian Process Regression (also known as GPR) is a class of non-parametric kernel-based probabilistic models which represents the input data as the random observations of a Gaussian stochastic process. The most important advantage of this approach in relation to other parametric regression techniques is that it is possible to effectively exploit a complex dataset which may consist of points sampled randomly in a multidimensional space.

In general, a Gaussian process $\mathcal{G}$ is a collection of random variables defined on a common probability space $(\Omega, \mathcal{F}, P)$, any finite number of which have consistent joint Gaussian distributions. We are interested in Gaussian processes for which the random variables in $\mathcal{G}$ are indexed by a point $\mathbf{x} \in \mathbb{R}^D$, $D \in \mathbb{N}$. Therefore, for all $\mathbf{x} \in \mathbb{R}^D$, $\mathcal{G}\left(\mathbf{x}, \cdot\right) : \Omega \to \mathbb{R}$ is a Gaussian random variable and if $X = \{\mathbf{x}_i | i = 1, \ldots, n\} \subset \mathbb{R}^D$ then

$(\mathcal{G}(\mathbf{x}_1), \ldots, \mathcal{G}(\mathbf{x}_n))^\top$ is a random Gaussian vector. Moreover, a Gaussian process is fully specified by its mean function $\mu(\mathbf{x}) : \mathbb{R}^D \to \mathbb{R}$ and by its covariance function $k(\mathbf{x}, \mathbf{x}') : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$.

Now, let us consider a training set $\mathcal{D}$ of n observations (the input data), $\mathcal{D} = \{(\mathbf{x}_i, y_i) | i = 1, \ldots, n\}$ where $X = \{\mathbf{x}_i | i = 1, \ldots, n\} \subset \mathbb{R}^D$ denotes the set of input vectors and $Y = \{y_i | i = 1, \ldots, n\} \subset \mathbb{R}$ denotes the set of scalar outputs. These observations are modeled as the realization of the sum of a Gaussian process and a noise source. Specifically,

$$y_i = f_i + \varepsilon_i \tag{3.1}$$

where $\{f_i = \mathcal{G}(\mathbf{x}_i) | i = 1, \ldots, n\}$ is a Gaussian process and $\{\varepsilon_i | i = 1, \ldots, n\}$ are i.i.d. random variables such that $\varepsilon_i \sim \mathcal{N}(0, \sigma_n^2)$. Moreover, the distribution of $\mathbf{f} = (f_1 \ldots f_n)$ is assumed to be given by

$$\mathbf{f} \sim \mathcal{N}(\mathbf{0}, K(X, X)), \tag{3.2}$$

where $K(X, X)$ is a $n \times n$ matrix with $K(X, X)_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$. Thus

$$\mathbf{y} \sim \mathcal{N}\left(\mathbf{0}, K(X, X) + \sigma_n^2 I_n\right), \tag{3.3}$$

where I_n is the $n \times n$ identity matrix.

Now, in addition, let us consider a test set $\tilde{X}$ of m points $\{\tilde{\mathbf{x}}_j | j = 1, \ldots, m\}$. The realizations $\tilde{f}_j = \mathcal{G}(\tilde{\mathbf{x}}_j)$ are not known but rather we want to estimate them considering the observed realizations of $\mathcal{G}$ in $\mathcal{D}$. The *a priori* joint distribution of $\mathbf{y}$ and $\tilde{\mathbf{f}} = \left(\tilde{f}_1 \ldots \tilde{f}_m\right)$ is given by

$$\begin{bmatrix} \mathbf{y} \\ \tilde{\mathbf{f}} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0}_n \\ \mathbf{0}_m \end{bmatrix}, \begin{bmatrix} K(X, X) + \sigma_n^2 I_n & K\left(X, \tilde{X}\right) \\ K\left(\tilde{X}, X\right) & K\left(\tilde{X}, \tilde{X}\right) \end{bmatrix}\right) \tag{3.4}$$

where $K\left(\tilde{X}, \tilde{X}\right)$ is a $m \times m$ matrix given by $K\left(\tilde{X}, \tilde{X}\right)_{i,j} = k(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j)$, $K\left(X, \tilde{X}\right)$ is a $n \times m$ matrix given by $K\left(X, \tilde{X}\right)_{i,j} = k(\mathbf{x}_i, \tilde{\mathbf{x}}_j)$ and $K\left(\tilde{X}, X\right)$ is a $m \times n$ matrix given by $K\left(\tilde{X}, X\right)_{i,j} = k(\tilde{\mathbf{x}}_i, \mathbf{x}_j)$.

Since we know the values for the training set, we can consider the conditional distribution of $\tilde{\mathbf{f}}$ given $\mathbf{y}$. It is possible to prove that $\tilde{\mathbf{f}} | \tilde{X}, \mathbf{y}, X$ follows a Gaussian distribution given by

$$\tilde{\mathbf{f}} | \tilde{X}, \mathbf{y}, X \sim \mathcal{N}\left(\mathbb{E}\left[\tilde{\mathbf{f}} | \tilde{X}, \mathbf{y}, X\right], Cov\left[\tilde{\mathbf{f}} | \tilde{X}, \mathbf{y}, X\right]\right), \tag{3.5}$$

where

$$\mathbb{E}\left[\tilde{\mathbf{f}} | \tilde{X}, \mathbf{y}, X\right] = K\left(\tilde{X}, X\right)\left[K(X, X) + \sigma_n^2 I_n\right]^{-1} \mathbf{y} \tag{3.6}$$

and

$$Cov\left[\tilde{\mathbf{f}} | \tilde{X}, \mathbf{y}, X\right] = K\left(\tilde{X}, \tilde{X}\right) - K\left(\tilde{X}, X\right)\left[K(X, X) + \sigma_n^2 I_n\right]^{-1} K\left(X, \tilde{X}\right). \tag{3.7}$$

Therefore, a natural choice consists in predicting the values $\tilde{\mathbf{f}}$ through $\mathbb{E}\left[\tilde{\mathbf{f}} | \tilde{X}, \mathbf{y}, X\right]$. Moreover, also confidence intervals for such a prediction can be computed.

The computation in (3.6) requires the knowledge of the covariance function K and of the noise variance σ_n^2. A typical used covariance function is the Automatic Relevance Determination Squared Exponential (ARD SE) kernel, which is given by

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2} \sum_{k=1}^{D} \frac{(\mathbf{x}_k - \mathbf{x}'_k)^2}{l_k^2}\right), \tag{3.8}$$

where σ_f^2 is called the signal variance and l_k is called the length-scale along the k direction. The parameters $\sigma_f^2, l_1, \ldots, l_D$ of the kernel function and σ_n^2 of the noise are called hyperparameters and need to be estimated. A common approach is to consider the maximum likelihood estimates which can be obtained by maximizing the log-likelihood function of the training data, that is by maximizing the following function:

$$-\frac{1}{2}\log\left(\det\left(K\left(X,X\right)+\sigma_n^2 I_n\right)\right)-\frac{1}{2}\mathbf{y}^\top\left[K\left(X,X\right)+\sigma_n^2 I_n\right]^{-1}\mathbf{y}. \tag{3.9}$$

Generally, one assumes the mean of the Gaussian process $\mathcal{G}$ to be zero, but this is not necessary. Conversely, it may be interesting to model the mean by using a deterministic function $\mu\left(X\right)$. In this case, the predictive value becomes

$$\mathbb{E}\left[\tilde{\mathbf{f}}|\tilde{X},\mathbf{y},X\right]=\mu\left(\tilde{X}\right)+K\left(\tilde{X},X\right)\left[K\left(X,X\right)+\sigma_n^2 I_n\right]^{-1}\left(\mathbf{y}-\mu\left(X\right)\right). \tag{3.10}$$

The function μ can be modeled using a set of basis functions, such as polynomials. In particular, when we consider a linear basis, the function μ is approximated via $\mathbf{h}\left(\mathbf{x}\right)\beta$, where $\mathbf{h}\left(\mathbf{x}\right)=\left(1,x_1,\ldots,x_D\right)$ and $\beta\in\mathbb{R}^{n+1}$ is a vector of coefficients estimated via the observed data. We stress out that in this paper we employ an ARD SE kernel and a linear basis function.

The development of the GPR model can be divided in the training step and the evaluation step (also called testing step). The training step only requires the knowledge of the training set $\mathcal{D}$ and it consists in estimating the β vector, the hyperparameters and the matrix product $A=\left[K\left(X,X\right)+\sigma_n^2 I_n\right]^{-1}\left(\mathbf{y}-\mu\left(X\right)\right)$. The evaluation step can be computed only after the training step has been accomplished and it consists in obtaining the predictions via the computation of $\mu\left(\tilde{X}\right)+K\left(\tilde{X},X\right)A$. We stress out that the training step is independent of the test set $\tilde{X}$. Thus one can store the values computed during the training step and perform the evaluation step many times with a small computational cost, which is $\mathcal{O}\left(n\cdot m\right)$.

Finally, we observe that the computation of the product matrix A (exact method) can be quite expensive since it involves the computation and the storage of a $n\times n$ matrix. Thus, for a large value of n (say $n>10000$), using the exact method for parameter estimation and making predictions can lead to technical difficulties. One of the approximation methods to overcome the storage of a large matrix is the Block Coordinate Descent (BCD) method (Bo and Sminchisescu [3]). The BCD method consists in computing A as the solution of the following optimization problem

$$A=\operatorname{argmin}_B\frac{1}{2}B^\top\left[K\left(X,X\right)+\sigma_n^2 I_n\right]B-B^\top\left(\mathbf{y}-\mu\left(X\right)\right), \tag{3.11}$$

which can be solved by means of the block coordinate descent algorithm without the storage of any large matrix (see Grippo and Sciandrone[16]).

3.2 Applying the GPR to the GMWB Contract

We aim to to apply the GPR method to GMWB products to speed up the computation of the price and of the Greeks. The modeling process starts by computing a training set $\mathcal{D}$. The predictor set X consists of n combinations of the product, market and model parameters. Specifically, as far as the Black-Scholes model is considered, the predictors are r, σ, α and κ. When the Heston model is considered, the predictors are r, v_0, k_v, θ_v, ω_v, ρ_v, α and κ. When the Heston Hull-White model is considered, the predictors are r, k_r, ω_r, ρ_r v_0, k_v, θ_v, ω_v, ρ_v, α and κ. We stress out that we can avoid considering the premium P as a predictor since the price $\mathcal{V}$ of the GMWB contract is directly proportional to P, that is $\mathcal{V}/P$ does not depend on P. Similarly Greeks can be obtained from the Greeks computed for a particular value of P: for example, one can prove that Δ does not depend on the considered value of P. Moreover we do not consider the contract maturity among the predictors since this is a discrete parameter which usually varies in a small set (usually $T = 5, 10$ or 20) and one can compute an independent GPR for each value of T.

Pseudo-random combinations are sampled uniformly over a fixed range for each parameter. Specifically, n combinations of the parameters are sampled through the Faure sequence (Niederreiter [24]) which permits to cover efficiently all the domain of the parameters and yields better results than a random sample.

For each parameter combination, we compute the price and then, observed data are passed to the GPR algorithm. Once the training step is finished, the model is ready to estimate prices. In order to asses the performances of the algorithm, we compare the exact price with the GPR predicted price for both the training set and a testing set consisting m random combinations of the parameters. The same procedure is performed for the Greeks.

In the scope of Variable Annuities, calculating the no-arbitrage fee is a common practice. In practice, this means computing the particular value of α which makes the initial value of the policy equal to the premium and thus it is also known as the fair cost of the policy. This value is usually computed by employing the secant method, seeking to equate the initial premium and the initial policy value. Once we have a GPR estimator, the search for this value can be carried out very efficiently using the estimator to evaluate the price of the function during the various steps of the secant algorithm.

4 Numerical Results

In this Section we report some numerical results. Specifically, in the first test we show the accuracy of the PDE and of the Hybrid PDE algorithms in pricing and in computing Delta and the no-arbitrage fees in the three stochastic models. Subsequently, we try to replicate the results of De Spiegeleer et al. [10], applying the method in the case of a put option. Then, in the following two tests, we apply the GPR method to predict the price and the Greek Δ of a GMWB contract respectively. Finally, in the last test, we apply the GPR method to predict the no-arbitrage fee. The PDE and Hybrid PDE algorithms have been implemented in C++, whereas the regression algorithm has been implemented using MATLAB and computations have been performed on a PC equipped with 8 GB of RAM and a 2.5GHz i5-7200u processor.

Let $\mathcal{Y}$ be the target function which we want to predict (price, Greek or no-arbitrage fee). The performance of the GPR algorithm is measured in terms of the following indicators which depend on the maximum and average absolute and relative error. Specifically, we compute the RMSE (Root Mean Squared Error) $RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(\mathcal{Y}\left(\mathbf{x}_i\right) - \mathcal{Y}_{GPR}\left(\mathbf{x}_i\right)\right)^2}$, the RMSRE (Root Mean Squared Relative Error) $RMSRE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(\frac{\mathcal{Y}(\mathbf{x}_i)-\mathcal{Y}_{GPR}(\mathbf{x}_i)}{\mathcal{Y}(\mathbf{x}_i)}\right)^2}$, the MaxAE (Maximum Absolute Error) $MaxAE = \max_i \left|\mathcal{Y}\left(\mathbf{x}_i\right) - \mathcal{Y}_{GPR}\left(\mathbf{x}_i\right)\right|$ and the MaxRE (Maximum Relative Error) $MaxRE = \max_i \left|\frac{\mathcal{Y}(\mathbf{x}_i)-\mathcal{Y}_{GPR}(\mathbf{x}_i)}{\mathcal{Y}(\mathbf{x}_i)}\right|$.

4.1 Testing PDE and HPDE methods

We show the accuracy of the considered methods, which are involved in the computation of the data employed in both the training and testing steps. We stress out that we compute the results by means of the finite difference PDE method in the case of the Black-Scholes model and by means of the Hybrid PDE method in the reminder models. Input parameters are reported in Tables 1, 2 and 3 while results are available in Table 4. In particular we compute the price, the Delta and the no-arbitrage fee considering several configurations with increasing number of time and space steps (see Goudenège et al. [14, 15] for further details). Numerical results suggested that a few steps is enough to obtain accurate values for all the three stochastic models. The computational times are very small for the Black-Scholes model, they increase in the Heston model and

Name	Symbol	Value
Premium	P	100
Maturity	T	10 years
Interest rate	r	0.02
Initial volatility	σ	0.245
Fees	α	0.05
Penalty	κ	0.05

Table 1: The parameters employed when considering the Black-Scholes model.

Name	Symbol	Value
Interest rate	r	0.02
Initial volatility	v_0	0.06
Rate of mean reversion	k_v	2.00
Long run variance	θ_v	0.06
Volatility of volatility	ω_v	0.60
Correlation	ρ_v	-0.55

Table 2: The parameters employed when considering the Heston model. Contract parameters are the same as in Table 1.

they become considerable in the Heston Hull-White model. For this reason we propose to use the GPR technique.

4.2 Put options

Let us consider the case of a Put option, just as done by De Spiegeleer et al. [10]. In particular, we focus on the Heston and on the Heston Hull-White models. The fixed values and the ranges for the input parameters are reported in Tables 5 and 6. Results are shown in Table 7.

4.3 Pricing Results

We test the ability of the GPR model to predict the price of a GMWB contract, that is the vector $\mathbf{y}$ which represents the observed prices of a GMWB contract.

Tables 8, 9 and 10 report the ranges for the input parameters. The GPR algorithm has been trained considering $n = 1250$, 2500, 5000, 10000, or 20000 combinations of parameters. The training method is the exact one for $n < 20000$ and otherwise is the BCD method. The regression model has been tested on both input data and $m = 20000$ additional parameters combinations sampled randomly. Numerical results are reported in Table 11 while the scatter plot of the estimation errors is shown in Figure 4.2.

4.4 Greeks Results

We test the ability of the GPR model to predict the Greeks of a GMWB contract. In particular, we focus on the first derivative Δ, which is crucial in hedging. Thus, in this numerical experiment, the vector $\mathbf{y}$ represents

Name	Symbol	Value
Initial interest rate	r_0	0.02
Rate of mean reversion	k_r	0.50
Volatility of interest rate	ω_r	0.05
Correlation	ρ_r	0.20
Initial volatility	v_0	0.06
Rate of mean reversion	k_v	2.00
Long run variance	θ_v	0.06
Volatility of volatility	ω_v	0.60
Correlation	ρ_v	−0.55

Table 3: The parameters employed when considering the Heston Hull-White model. Contract parameters are the same as in Table 1.

Black-Scholes model				
Time and space steps	125×125	250×250	500×500	1000×1000
Price	100.58 (3.1e−2)	100.59 (5.6e−2)	100.59 (2.7e−1)	100.59 (4.5e−1)
Delta	0.3125 (3.1e−2)	0.3124 (5.6e−2)	0.3125 (2.7e−1)	0.3125 (4.5e−1)
No-arbitrage fee	541.50 (8.1e−2)	542.69 (1.8e−1)	543.02 (7.0e−1)	543.19 (1.7e+0)

Heston model				
Time and space steps	125×125	250×250	500×500	1000×1000
Price	99.37 (9.2e−2)	99.43 (2.6−1)	99.44 (1.3e+0)	99.45 (7.2e+0)
Delta	0.3274 (9.2e−2)	0.3298 (2.6−1)	0.3306 (1.3e+0)	0.3310 (7.2e+0)
No-arbitrage fee	456.28 (3.5−1)	459.60 (1.5e+0)	460.65 (7.2e+0)	461.11 (4.9e+1)

Heston Hull-White model				
Time and space steps	125×125	250×250	500×500	1000×1000
Price	102.31 (1.2e+0)	102.33 (1.6e+1)	102.34 (2.0e+2)	102.33 (3.0e+3)
Delta	0.3225 (1.2e+0)	0.3245 (1.6e+1)	0.3252 (2.0e+2)	0.3255 (3.0e+3)
No-arbitrage fee	701.73 (1.1e+1)	703.59 (9.8e+2)	703.68 (1.2e+3)	703.80 (1.8e+4)

Table 4: PDE and Hybrid PDE computation results. The values in brackets are the computational times measured in seconds. The no-arbitrage fee is expressed in bps.

Name	Symbol	Range
Spot	S_0	100
Dividend	q	$[0.00, 0.03]$
Interest rate	r	$[0.01, 0.02]$
Initial volatility	V_0	$[0.02, 0.10]$
Rate of mean reversion	k_v	$[1.4, 2.6]$
Long run variance	θ_v	$[0.02, 0.10]$
Volatility of volatility	ω_v	$[0.45, 0.75]$
Correlation	ρ_v	$[-0.7, -0.4]$
Maturity	T	$\left[\frac{11}{12}, 1\right]$
Strike	K	$[70, 130]$

Table 5: The range of the parameters for Put options when considering the Heston model.

Name	Symbol	Range
Spot	S_0	100
Dividend	q	$[0.00, 0.03]$
Initial interest rate	r_0	$[1.4, 2.6]$
Rate of mean reversion	k_r	$[0.02, 0.10]$
Volatility of interest rate	ω_r	$[0.45, 0.75]$
Correlation	ρ_r	$[-0.7, -0.4]$
Initial volatility	V_0	$[0.02, 0.10]$
Rate of mean reversion	k_v	$[1.4, 2.6]$
Long run variance	θ_v	$[0.02, 0.10]$
Volatility of volatility	ω_v	$[0.45, 0.75]$
Correlation	ρ_v	$[-0.7, -0.4]$
Maturity	T	$\left[\frac{11}{12}, 1\right]$
Strike	K	$[70, 130]$

Table 6: The range of the parameters for Put options when considering the Heston Hull-White model.

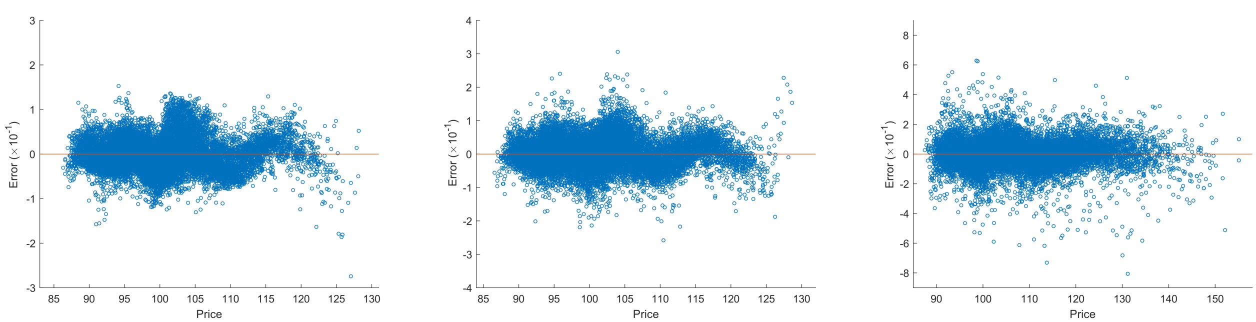

Figure 4.1: Out-of-sample price predictions with a GPR model trained on 10000 points (Black-Scholes model on the left, Heston model in the middle and Heston Hull-White on the right).

Put option pricing (Heston model)							
		European			American		
Size of training set		5000	10000	20000	5000	10000	20000
In-sample prediction	RMSE	1.99e-02	1.73e-02	1.55e-02	2.47e-02	2.12e-02	2.02e-02
	RMSRE	3.91e-03	3.11e-03	2.78e-03	3.88e-03	2.89e-03	2.82e-03
	MaxAE	1.87e-01	2.16e-01	2.47e-01	2.45e-01	3.08e-01	2.88e-01
	MaxRE	9.47e-02	5.60e-02	4.77e-02	7.56e-02	3.41e-02	3.34e-02
Out-of-sample prediction	RMSE	2.06e-02	1.69e-02	1.50e-02	2.57e-02	2.10e-02	1.94e-02
	RMSRE	4.68e-03	3.48e-03	2.88e-03	4.46e-03	3.10e-03	2.84e-03
	MaxAE	2.72e-01	2.39e-01	2.12e-01	5.64e-01	4.59e-01	3.91e-01
	MaxRE	1.71e-01	9.42e-02	6.22e-02	1.39e-01	7.09e-02	4.89e-02

Put option pricing (Heston Hull-White model)							
		European			American		
Size of training set		5000	10000	20000	5000	10000	20000
In-sample prediction	RMSE	2.13e-02	1.97e-02	1.84e-02	2.56e-02	2.42e-02	2.17e-02
	RMSRE	4.47e-03	4.24e-03	3.81e-03	4.16e-03	3.61e-03	3.18e-03
	MaxAE	1.87e-01	2.40e-01	2.14e-01	2.44e-01	3.66e-01	2.96e-01
	MaxRE	7.03e-02	8.16e-02	6.03e-02	5.30e-02	5.51e-02	5.49e-02
Out-of-sample prediction	RMSE	2.27e-02	2.02e-02	1.85e-02	2.96e-02	2.56e-02	2.26e-02
	RMSRE	5.13e-03	4.23e-03	3.73e-03	5.07e-03	3.81e-03	3.26e-03
	MaxAE	3.05e-01	2.54e-01	2.08e-01	4.40e-01	4.07e-01	3.40e-01
	MaxRE	1.76e-01	8.34e-02	6.48e-02	1.70e-01	8.31e-02	4.28e-02

Table 7: Performances of the GPR method in computing the price of a Put option.

Name	Symbol	Range
Premium	P	100
Maturity	T	10 years
Interest rate	r	$[0.01, 0.03]$
Initial volatility	σ	$[0.10, 0.30]$
Fees	α	$[0.00, 0.10]$
Penalty	κ	$[0.00, 0.20]$

Table 8: The range of the parameters when considering the Black-Scholes model.

Name	Symbol	Range
Premium	P	100
Maturity	T	10 years
Interest rate	r	$[0.01, 0.03]$
Initial volatility	v_0	$[0.02, 0.10]$
Rate of mean reversion	k_v	$[1.40, 2.60]$
Long run variance	θ_v	$[0.02, 0.10]$
Volatility of volatility	ω_v	$[0.45, 0.75]$
Correlation	ρ_v	$[-0.70, -0.40]$
Fees	α	$[0.00, 0.10]$
Penalty	κ	$[0.00, 0.20]$

Table 9: The range of the parameters when considering the Heston model.

Name	Symbol	Range
Premium	P	100
Maturity	T	10 years
Initial interest rate	r_0	$[0.01, 0.03]$
Rate of mean reversion	k_r	$[0.05, 0.75]$
Volatility of interest rate	ω_r	$[0.005, 0.10]$
Correlation	ρ_r	$[0.05, 0.35]$
Initial volatility	v_0	$[0.02, 0.10]$
Rate of mean reversion	k_v	$[1.40, 2.60]$
Long run variance	θ_v	$[0.02, 0.10]$
Volatility of volatility	ω_v	$[0.45, 0.75]$
Correlation	ρ_v	$[-0.70, -0.40]$
Fees	α	$[0.00, 0.10]$
Penalty	κ	$[0.00, 0.20]$

Table 10: The range of the parameters when considering the Heston Hull-White model.

Pricing (Black-Scholes model)						
Size of training set		1250	2500	5000	10000	20000
In-sample prediction	RMSE	3.90e-02	3.46e-02	3.36e-02	3.23e-02	2.83e-02
	RMSRE	3.87e-04	3.44e-04	3.33e-04	3.21e-04	2.81e-04
	MaxAE	1.49e-01	1.31e-01	1.64e-01	2.09e-01	1.53e-01
	MaxRE	1.43e-03	1.33e-03	1.52e-03	1.69e-03	1.42e-03
Out-of-sample prediction	RMSE	4.77e-02	3.82e-02	3.44e-02	3.27e-02	2.83e-02
	RMSRE	4.69e-04	3.77e-04	3.41e-04	3.24e-04	2.81e-04
	MaxAE	4.84e-01	4.59e-01	3.24e-01	2.74e-01	1.60e-01
	MaxRE	3.93e-03	3.62e-03	2.55e-03	2.16e-03	1.41e-03
Speed-up		×25000	×13000	×6300	×3100	×1700

Pricing (Heston model)						
Size of training set		1250	2500	5000	10000	20000
In-sample prediction	RMSE	4.17e-02	4.08e-02	4.02e-02	4.00e-02	3.69e-02
	RMSRE	4.12e-04	4.03e-04	3.97e-04	3.94e-04	3.64e-04
	MaxAE	1.79e-01	1.84e-01	2.34e-01	2.68e-01	2.26e-01
	MaxRE	1.78e-03	1.85e-03	2.30e-03	2.61e-03	2.20e-03
Out-of-sample prediction	RMSE	7.49e-02	5.91e-02	4.67e-02	4.31e-02	3.78e-02
	RMSRE	7.33e-04	5.79e-04	4.63e-04	4.23e-04	3.73e-04
	MaxAE	8.10e-01	6.28e-01	4.68e-01	3.06e-01	2.28e-01
	MaxRE	7.69e-03	5.96e-03	4.51e-03	2.94e-03	2.23e-03
Speed-up		×95000	×48000	×24000	×12000	×6000

Pricing (Heston Hull-White model)						
Size of training set		1250	2500	5000	10000	20000
In-sample prediction	RMSE	6.60e-02	6.72e-02	7.00e-02	6.96e-02	6.88e-02
	RMSRE	6.21e-04	6.35e-04	6.61e-04	6.58e-04	6.49e-04
	MaxAE	2.79e-01	3.16e-01	3.87e-01	5.23e-01	5.59e-01
	MaxRE	2.80e-03	3.40e-03	4.16e-03	5.71e-03	6.17e-03
Out-of-sample prediction	RMSE	7.49e-02	5.91e-02	4.67e-02	4.31e-02	3.78e-02
	RMSRE	7.33e-04	5.79e-04	4.63e-04	4.23e-04	3.73e-04
	MaxAE	8.10e-01	6.28e-01	4.68e-01	3.06e-01	2.28e-01
	MaxRE	7.69e-03	5.96e-03	4.51e-03	2.94e-03	2.23e-03
Speed-up		×420000	×210000	×110000	×52000	×26000

Table 11: Performances of the GPR method in computing the price.

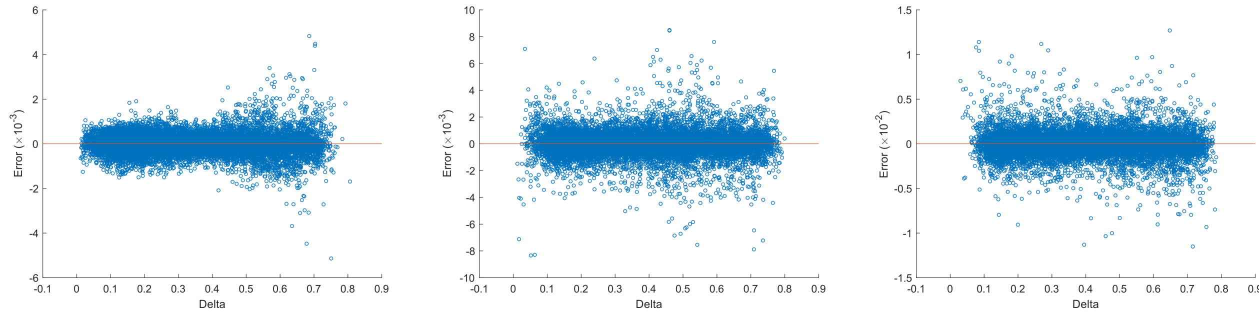

Figure 4.2: Out-of-sample Delta predictions with a GPR model trained on 10000 points (Black-Scholes model on the left, Heston model in the middle and Heston Hull-White on the right).

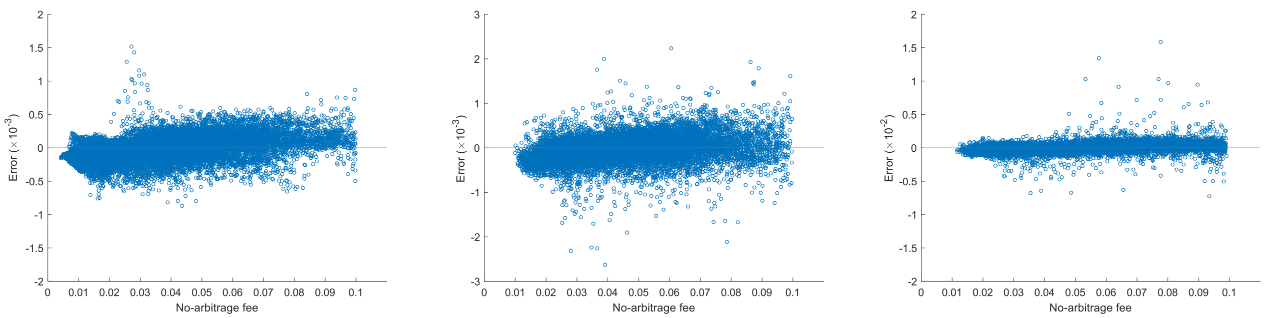

Figure 4.3: Out-of-sample no-arbitrage fee predictions with a GPR model trained on 10000 points (Black-Scholes model on the left, Heston model in the middle and Heston Hull-White on the right).

the observed Δ of a GMWB contract. Numerical results are reported in Table 12 while the scatter plot of the estimation errors is shown in Figure 4.1.

4.5 No-arbitrage Fee Results

We test the ability of the GPR model to compute the no-arbitrage fee of a GMWB contract. In this numerical experiment, is the vector $\mathbf{y}$ represent the observed price of a GMWB contract. We employ the secant method to compute the fair value of α for $m = 20000$ randomly generated parameters combinations and we compare the predicted value with the exact one. Numerical results are reported in Table 13 while the scatter plot of the estimation errors is shown in Figure 4.3.

4.6 Numerical Remarks

Tables 11, 12 and 13 show the performance of the developed approach. Errors are very small in all the considered cases, indicating that the predictions are very accurate. The most interesting aspect is the gain in terms of computation time: it is reduced by thousands of times, in particular for the Heston Hull-White model.

Moreover, Table 14 shows the computational time for the training step. The cost of the training step depends on both the size of the training step, the considered model and the target function (price or Delta). We observe that the required time is high when the size of the training set is $n = 20000$ because in this case

Delta Computation (Black-Scholes model)						
Size of training set		1250	2500	5000	10000	20000
In-sample prediction	RMSE	4.25e-04	3.79e-04	3.65e-04	3.76e-04	3.71e-04
	RMSRE	1.95e-03	1.84e-03	1.79e-03	1.81e-03	1.89-03
	MaxAE	1.99e-03	2.28e-03	2.35e-03	3.26e-03	3.14e-03
	MaxRE	1.41e-02	2.00e-02	1.55e-02	2.82e-02	1.99e-02
Out-of-sample prediction	RMSE	7.67e-04	5.73e-04	4.52e-04	4.10e-04	3.84e-04
	RMSRE	4.05e-03	2.88e-03	2.43e-03	2.06e-03	1.96e-03
	MaxAE	1.35e-02	9.79e-03	7.12e-03	5.14e-03	3.65e-03
	MaxRE	1.62e-01	9.28e-02	8.36e-02	4.18e-02	3.10e-02
Speed-up		$\times$25000	$\times$13000	$\times$6300	$\times$3100	$\times$1700

Delta Computation (Heston model)						
Size of training set		1250	2500	5000	10000	20000
In-sample prediction	RMSE	4.26e-04	4.40e-04	4.99e-04	5.37e-04	5.72e-04
	RMSRE	1.97e-03	1.92e-03	2.23e-03	3.34e-03	3.22e-03
	MaxAE	2.70e-03	2.61e-03	3.47e-03	4.70e-03	6.37e-03
	MaxRE	3.09e-02	2.48e-02	3.93e-02	2.10e-02	1.86e-02
Out-of-sample prediction	RMSE	1.81e-03	1.25e-03	9.43e-04	8.11e-04	6.64e-04
	RMSRE	1.15e-02	7.94e-03	6.58e-03	4.75e-03	4.20e-03
	MaxAE	1.78e-02	1.27e-02	1.26e-02	8.49e-03	8.02e-03
	MaxRE	6.82e-01	3.38e-01	3.11e-01	2.07e-01	1.70e-01
Speed-up		$\times$90000	$\times$44000	$\times$23000	$\times$12000	$\times$6000

Delta Computation (Heston Hull-White model)						
Size of training set		1250	2500	5000	10000	20000
In-sample prediction	RMSE	2.98e-04	5.67e-04	6.92e-04	8.25e-04	8.17e-04
	RMSRE	1.19e-03	2.14e-03	3.53e-03	4.25e-03	3.62e-03
	MaxAE	1.49e-03	3.29e-03	4.57e-03	7.32e-03	8.49e-03
	MaxRE	1.73e-02	3.12e-02	1.58e-01	2.73e-01	2.48e-01
Out-of-sample prediction	RMSE	2.38e-03	1.86e-03	1.47e-03	1.19e-03	1.04e-03
	RMSRE	1.08e-02	7.86e-03	7.02e-03	5.63e-03	4.17e-03
	MaxAE	2.72e-02	1.83e-02	1.92e-02	1.27e-02	1.15e-02
	MaxRE	4.13e-01	3.41e-01	3.22e-01	2.33e-01	2.15e-01
Speed-up		$\times$410000	$\times$200000	$\times$100000	$\times$52000	$\times$26000

Table 12: Performances of the GPR method in computing Delta.

No-arbitrage fee Computation (Black-Scholes model)					
Size of training set	1250	2500	5000	10000	20000
RMSE	2.79e-04	2.24e-04	1.99e-04	1.83e-04	1.61e-04
RMSRE	1.14e-02	9.80e-03	8.73e-03	8.49e-03	7.46e-03
MaxAE	2.54e-03	1.91e-03	1.29e-03	1.52e-03	1.11e-03
MaxRE	8.87e-02	7.45e-02	6.08e-02	4.83e-02	4.09e-02
Speed-up	×25000	×13000	×6300	×3100	×1700

No-arbitrage fee Computation (Heston model)					
Size of training set	1250	2500	5000	10000	20000
RMSE	4.83e-04	3.71e-04	3.04e-04	2.64e-04	2.26e-04
RMSRE	1.33e-02	1.08e-02	8.91e-03	7.88e-03	7.05e-03
MaxAE	5.14e-03	3.38e-03	2.68e-03	2.63e-03	2.21e-03
MaxRE	1.41e-01	9.26e-02	8.27e-02	7.95e-02	7.30e-02
Speed-up	×96000	×48000	×24000	×12000	×6100

No-arbitrage fee Computation (Heston Hull-White model)					
Size of training set	1250	2500	5000	10000	20000
RMSE	1.02e-03	8.33e-04	7.19e-04	6.15e-04	5.62e-04
RMSRE	2.32e-02	1.91e-02	1.69e-02	1.47e-02	1.42e-02
MaxAE	1.52e-02	1.64e-02	1.51e-02	1.58e-02	1.12e-02
MaxRE	3.31e-01	2.72e-01	2.39e-01	2.33e-01	1.84e-01
Speed-up	×410000	×200000	×100000	×52000	×26000

Table 13: Performances of the GPR method in computing the no-arbitrage fee.

Training time (Black-Scholes model)					
Size of training set	1250	2500	5000	10000	20000
Price	15	45	48	48	1602
Delta	22	76	83	92	1616

Training time (Heston model)					
Size of training set	1250	2500	5000	10000	20000
Price	35	103	117	123	2417
Delta	23	108	115	121	2583

Training time (Heston Hull-White model)					
Size of training set	1250	2500	5000	10000	20000
Price	49	154	185	188	3219
Delta	32	110	128	142	3144

Table 14: The computational time (in seconds) to train the GPR algorithm.

the BCD algorithm is used in place of the exact computation.

Finally, Table 15 shows the predicted price, Delta and no-arbitrage fee for the same GMWB policy considered in Table 4. We observe that the estimates are very accurate and the gain in terms of computational time is considerable, especially for the Heston Hull-White model.

5 Conclusions

In this paper we have presented how Gaussian Process Regression can be applied in the insurance field to address the problem of pricing and hedging a GMWB Variable Annuity with stochastic volatility and stochastic interest rate. The computational time is considerably reduced as the greater computational effort is carried out during the training phase, which must be performed only once before the actual use of the model while computing a single prediction and it has a linear cost in the number of input observation. The use of a quasi-Monte Carlo methods and of a general kernel allow us to obtain good results. Moreover, the use of the Block Coordinate Descent method allows one to manage a large amount of observed data. The resulting model can be applied to compute the prices of the policies, for hedging purposes, or to compute the no-arbitrage cost of a policy.

We conclude observing that the same approach may be applied to all types of Variable Annuities contracts.

GPR prediction (Black-Scholes model)					
Size of training set	1250	2500	5000	10000	20000
Price	100.62 (4.2e−5)	100.62 (1.3e−4)	100.61 (1.7e−4)	100.62 (3.5e−4)	100.61 (6.2−4)
Delta	0.3127 (4.5e−5)	0.3126 (9.6e−5)	0.3126 (1.9e−4)	0.3125 (3.2e−4)	0.3125 (6.6e−4)
No-arbitrage fee	544.23 (1.9e−2)	543.75 (1.9e−2)	543.74 (2.7e−2)	543.73 (2.8e−2)	543.91 (3.0e−2)

GPR prediction (Heston model)					
Size of training set	1250	2500	5000	10000	20000
Price	99.44 (7.4e−5)	99.44 (1.7e−4)	99.44 (3.0e−4)	99.44 (5.7e−4)	99.44 (1.1e−3)
Delta	0.3318 (7.7e−5)	0.3310 (1.6e−4)	0.3312 (3.2e−4)	0.3310 (5.9e−4)	0.3310 (1.1e−3)
No-arbitrage fee	461.13 (1.4e−2)	461.36 (2.0e−2)	461.20 (2.5e−2)	461.42 (2.8e−2)	461.32 (3.1e−2)

GPR prediction (Heston Hull-White model)					
Size of training set	1250	2500	5000	10000	20000
Price	102.34 (8.5e−5)	102.33 (2.2e−4)	102.33 (4.6e−4)	102.33 (6.8e−4)	102.33 (1.3e−3)
Delta	0.3242 (8.4e−5)	0.3245 (1.9e−4)	0.3245 (4.2e−4)	0.3245 (7.4e−4)	0.3246 (1.4e−3)
No-arbitrage fee	702.34 (1.8e−2)	704.15 (1.9e−2)	704.05 (2.5e−2)	704.29 (2.8e−2)	704.71 (3.4e−2)

Table 15: GPR computation results. The values in brackets are the computational times measured in seconds. The no-arbitrage fee is expressed in bps.

Appendices

A MATLAB code

In this Appendix we give some details about the MATLAB code used to implement the GPR algorithm. In particular, we have used the R2018b version of the software.

Once input data have been loaded as column vectors, predictors are stored in a Table. For example, aiming to predict the price of a GMWB policy in the Black-Scholes model, we write the following code:

```
predictors_in = table(Interest_rate_in, Volatility_in, Fee_in, Penalty_in);
response_in = Price_in;
```

Then, we train the model by calling the function fitrgp as follows:

```
my_GPR_model=fitrgp(predictors_in, response_in, 'BasisFunction', 'linear', ...
              'KernelFunction', 'ardsquaredexponential', 'Standardize', true);
```

The fitgrp function has several options and thus, for further details we refer the interest reader to the webpage [23]. Finally, once the model has been trained, one can obtain predictions:

```
predictors_out = table(Interest_rate_out, Volatility_out, Fee_out, Penalty_out);
response_out = my_GPR_model.predictFcn(predictors_out);
```

References

[1] A. R. Bacinello, P. Millossovich, A. Olivieri, and E. Pitacco. Variable Annuities: A unifying valuation approach. *Insurance: Mathematics and Economics*, 49(3):285–297, 2011.

[2] D. Bauer, A. Kling, and J. Russ. A universal pricing framework for guaranteed minimum benefits in Variable Annuities. *ASTIN Bulletin: The Journal of the IAA*, 38(2):621–651, 2008.

[3] L. Bo and C. Sminchisescu. Greedy block coordinate descent for large scale Gaussian process regression. *arXiv preprint arXiv:1206.3238*, 2012.

[4] M. Briani, L. Caramellino, and A. Zanette. A hybrid approach for the implementation of the Heston model. *IMA Journal of Management Mathematics*, 28(4):467–500, 2017.

[5] M. Briani, L. Caramellino, and A. Zanette. A hybrid tree/finite-difference approach for Heston–Hull–White-type models. *Journal of Computational Finance*, 21(3), 2017.

[6] D. Brigo and F. Mercurio. *Interest Rate Models - Theory and Practice: with Smile, Inflation and Credit*. Springer Science & Business Media, 2007.

[7] Z. Chen and P. A. Forsyth. A numerical scheme for the impulse control formulation for pricing Variable Annuities with a Guaranteed Minimum Withdrawal Benefit (GMWB). *Numerische Mathematik*, 109(4):535–569, 2008.

[8] M. Costabile. A lattice-based model to evaluate Variable Annuities with Guaranteed Minimum Withdrawal Benefits under a regime-switching model. *Scandinavian Actuarial Journal*, 2017(3):231–244, 2017.

[9] T.-S. Dai, S. S. Yang, and L.-C. Liu. Pricing guaranteed minimum/lifetime withdrawal benefits with various provisions under investment, interest rate and mortality risks. *Insurance: Mathematics and Economics*, 64:364–379, 2015.

[10] J. De Spiegeleer, D. B. Madan, S. Reyners, and W. Schoutens. Machine learning for quantitative finance: fast derivative pricing, hedging and fitting. *Quantitative Finance*, 18(10):1635–1643, 2018.

[11] P. Deprez, P. V. Shevchenko, and M. V. Wüthrich. Machine Learning techniques for mortality modeling. *European Actuarial Journal*, 7(2):337–352, 2017.

[12] R. F. Donnelly, S. Jaimungal, and D. Rubisov. Valuing guaranteed withdrawal benefits with stochastic interest rates and volatility. *Quantitative Finance*, 14(2):369–382, 2014.

[13] G. Gan. Application of Data Clustering and Machine Learning in Variable Annuity valuation. *Insurance: Mathematics and Economics*, 53(3):795–801, 2013.

[14] L. Goudenège, A. Molent, and A. Zanette. Pricing and hedging GLWB in the Heston and in the Black–Scholes with stochastic interest rate models. *Insurance: Mathematics and Economics*, 70:38–57, 2016.

[15] L. Goudenège, A. Molent, and A. Zanette. Pricing and hedging GMWB in the Heston and in the Black–Scholes with stochastic interest rate models. *Computational Management Science*, 16(1):217–248, 2019.

[16] L. Grippo and M. Sciandrone. On the convergence of the block nonlinear Gauss–Seidel method under convex constraints. *Operations Research Letters*, 26(3):127–136, 2000.

[17] S. L. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The Review of Financial Studies*, 6(2):327–343, 1993.

[18] K. Ignatieva, A. Song, and J. Ziveyi. Fourier space time-stepping algorithm for valuing Guaranteed Minimum Withdrawal Benefits in Variable Annuities under regime-switching and stochastic mortality. *ASTIN Bulletin: The Journal of the IAA*, pages 1–31, 2017.

[19] I. R. Institute. IRI issues second-quarter 2018 annuity sales report. `https://www.myirionline.org/newsroom/newsroom-detail-view/iri-issues-fourth-quarter-2017-annuity-sales-report`. Accessed: 26 Nov 2018.

[20] H. Kammeyer and J. Kienitz. The Heston–Hull–White Model Part I: Finance and Analytics. *Wilmott*, 2012(57):46–53, 2012.

[21] X. Luo and P. V. Shevchenko. Valuation of Variable Annuities with guaranteed minimum withdrawal and death benefits via stochastic control optimization. *Insurance: Mathematics and Economics*, 62:5–15, 2015.

[22] X. Luo and P. V. Shevchenko. Variable Annuity with GMWB: surrender or not, that is the question. Available at `https://arxiv.org/abs/1507.08738`, 2015.

[23] Mathworks. Fit a Gaussian Process Regression (GPR) model. https://it.mathworks.com/help/stats/fitrgp.html. Accessed: 08 Mar 2019.

[24] H. Niederreiter. *Random number generation and quasi-Monte Carlo methods*, volume 63. Siam, 1992.

[25] C. E. Rasmussen and C. K. I. Williams. Gaussian processes for machine learning. *the MIT Press*, 2(3):4, 2006.

[26] W. T. Watson. Variable Annuities: Market pressures push the case for model sophistication. `https://www.willistowerswatson.com/en/insights/2018/12/variable-annuities-market-pressures-push-the-case-for-model-sophistication`. Accessed: 18 Feb 2019.

[27] M. V. Wüthrich. Neural networks applied to chain–ladder reserving. *European Actuarial Journal*, 8(2):407–436, 2018.