Some applications of Gaussian Processes Regression in Finance

Samy Mekkaoui

CMAP, Ecole Polytechnique
and LPSM

Master M2MO, SFA
Mastère spécialisé FGR, DS
ENSAE 3A

5 March 2025

Objectives :

- Present some basic applications of $\mathcal{GPR}$ in finance and insurance for pricing purposes and discuss about pros and cons of the method.
- Show how can $\mathcal{GPR}$ be combined to classic numerical methods to lead to new efficient algorithms.

Contents

Call Option and Greeks under $B - S$ model

Assuming a probability space $(\Omega, \mathcal{F})$ supporting a brownian motion $W = (W_t)_{t \geqslant 0}$. We consider a Black-Scholes model for the underlying S with dynamics given under the risk neutral probability Q by :

$$dS_t = S_t(rdt + \sigma dW_t).$$

Our aim is to learn thanks to $\mathcal{GPR}$ the price and delta of a call option given a set of model parameters.

Table: Parameters used in the numerical experiments in the $B - S$ setting

Parameters	r	σ	T	K
Value	0.03	0.3	1	100

For the numerical experiments, we used the *Radial Basis Kernel* which is given by :

$$k(x, x') = \sigma_f^2 e^{-\frac{1}{2l^2}\|x-x'\|^2}.$$

In the Black-Scholes model, and by denoting C_t the price of the call option at time t, it is known that we have $C = C(t, S_t)$ for $0 \leqslant t \leqslant T$ and for a measurable function C. Therefore, the idea of the $\mathcal{GPR}$ is to sample data of the form $(S^i, C^i)_{i=1,\dots,n}$ where n denotes the number of training samples and where :

- S^i denotes the price of the stock for the sample i.
- C^i denotes the model price of the call option with stock price S^i ($C^i = C(0, S^i)$)

From these training data, we fit our $\mathcal{GPR}$ thanks to the hyperparameter tuning. In our cases, it consists in finding the optimal pair (σ_f, l) which maximizes the loglikelihood of our data.
Once this is done, we can test our fitting model on a test set. For the numerical results, we set the following number of training and test data points :

Table: Number of training and test points for the numerical experiments

	Training set	Test Set
Number of Points	40	40

Call Option and Delta under $B - S$ model

Some numerical results

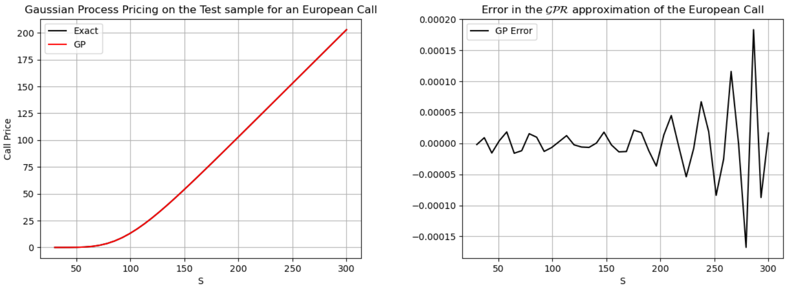


Figure: Call price learnt using $\mathcal{GPR}$ using 40 training points and 40 testing points and associated error

Once we have learn the $\mathcal{GPR}$ function associated to the call price function, we can calculate through analytic derivatives the greeks of the option. Indeed, it can be shown that we have the following formula :

$$\partial_{S_*}[f_*|S, C, S_*] = (\frac{1}{2}(S - S_*)K_{S*,S})[K_{S,S} + \sigma_n^2 I]^{-1} C \quad (1)$$

where S_* refers to the sample of the test set and σ_n^2 is an additive noise in the $\mathcal{GPR}$. Using this formula, we can therefore compare the delta obtained through $\mathcal{GPR}$ by an analytic formula with the true delta of the option.

Call Option and Delta under $B - S$ model

Some numerical results

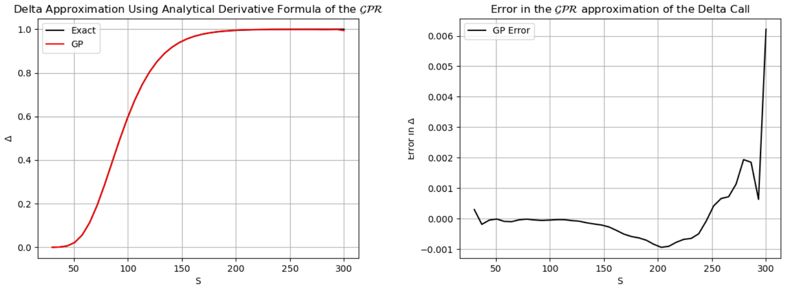


Figure: Delta of a Call Option obtained from analytic formulas using $\mathcal{GPR}$ and associated error with true Delta value

Remark

If we wanted to learn the vega of the option for a fixed S_0, we could have sampled $(\sigma^i, C^i)_{i=0,\ldots,n}$ and we could using the same formula (1). We could also get the gamma of the option by differentiating again the equation (1) with respect to S_*.

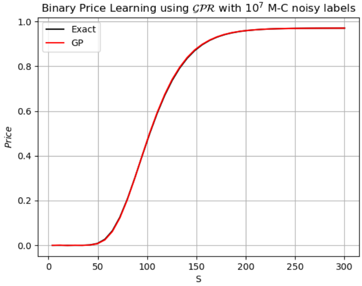
Training $\mathcal{GPR}$ with noisy data

From now, we show the capability of the $\mathcal{GPR}$ to learn efficient prices but when the $\mathcal{GPR}$ was fed with exact data in the sense that the prices $(C^i)_{i=1,\ldots,n}$ were computed through an exact formula. However, in most of the models, we don't have access to closed formulas and therefore we have to sample prices from a numerical method before using the same procedure as presented before. We will present result in the case of a binary option in the $B-S$ model with the following payoff :

$$\mathbb{1}_{S_T} \geqslant K$$

We will therefore sample price of a binary option using first 10^7 M-C simulations and after 10^5 to see the impact of the noisy data in the $\mathcal{GPR}$ performance.

A Binary Option with 10^7 simulations in data sampling



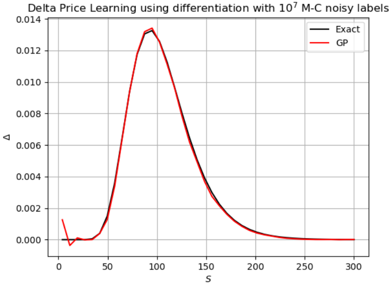


Figure: Price and Delta of a Binary Option using 10^7 simulations for the training of the $\mathcal{GPR}$

- When setting $M = 10^7$ simulations, we still have good results in the binary price learning and in the derivative through *GPR* but it still looks less efficient then in the case we provided exact option prices to train the model.

A Binary Option with 10^5 simulations in data sampling

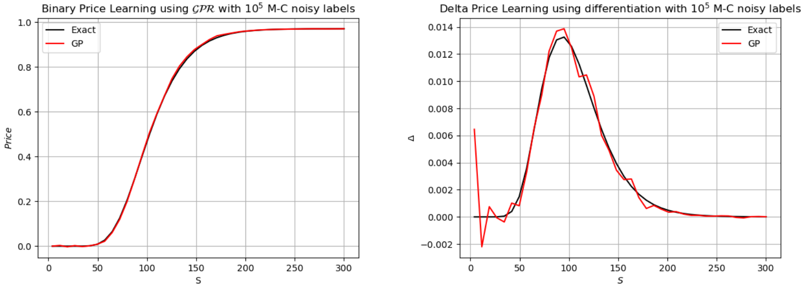


Figure: Price and Delta of a Binary Option using 10^5 simulations for the training of the $\mathcal{GPR}$

- The impact of the labels is therefore significant in the sense the GPR will have more difficulties to learn the true pricing surface and the derivatives can have significant differences. This feature is particularly important and it highlights the fact that the data which is provided to the GPR needs to be the most accurate as possible.

We present the case of a Guaranteed Minimum Maturity Benefit (*GMMB*) contract with payoff given by :

$$\mathbb{1}_{\tau > T} \max(S_T, K).$$

where :

- τ denotes the mortality date residual of the insured starting from 0 at age 50.
- S_T is the value of the underlying stock at time T with $S_0 \in \mathbb{R}_+^*$ and K is a minimum guarantee

Assuming the following dynamics for the underlying stock and the mortality rate λ :

$$\begin{aligned} dS_t &= S_t(rdt + \sigma dW_t^1), \\ d\lambda_t &= c\lambda_t dt + \xi\sqrt{\lambda_t} dW_t^2, \\ d < W^1, W^2 >_t &= \rho dt. \end{aligned}$$

where τ can be defined as $\tau = \inf\{t \geqslant 0 : \int_0^t \lambda_s ds \geqslant \nu\}$ where $\nu \sim \mathcal{E}(1) \perp\!\!\!\perp \mathrm{W}{=}(\mathrm{W}^1, W^2)$. The fair value of the *GMMB* contract is then defined as $t = 0$ by :

$$P_0^{GMMB}(S_0, \lambda_0) = \mathbb{E}^Q[e^{-rT} \mathbb{1}_{\tau > T} \max(S_T, K)]. \tag{2}$$

Gaussian Process Regression

$\mathcal{GPR}$ to learn a *GMMB* price surface

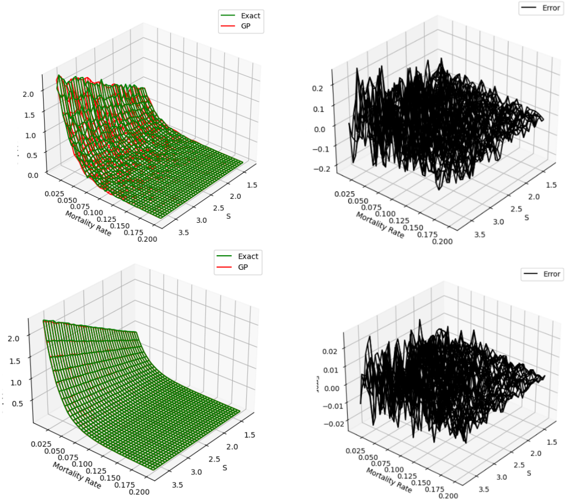


Figure: 1000 vs 100000 *MC* simulations to learn the price surface P_0^{GMMB} as a function of (λ_0, S_0) under the model (2) with the parameters : ($c = 7, 50.10^{-2}$, $\xi = 5, 97.10^{-4}$, $r = 0.02$, $\sigma = 0.2$, $\rho = -0.7$, $K = 1$ from [8])

Contents

Table: Different Types of XVA

XVA	valuation adjustment	Expected Cost of the Bank
CVA	Credit Valuation Adjustment	Client Default Losses
DVA	Debt Valuation Adjustment	Bank Default Losses
FVA	Funding Valuation Adjustment	Funding expenses for variation margin
MVA	Margin Valuation Adjustment	Funding expenses for initial margin
KVA	Capital Valuation Adjustment	Remuneration of Shareholder capital at risk

- *CVA* and *DVA* refer to credit valuation adjustments. When both quantities are computed, we use the term *BCVA* as *Bilateral Credit Valuation Adjustment*.
- *FVA* and *MVA* refer to funding valuation adjustments through the impact of collateralization in exchange markets.
- *KVA* refers to the capital valuation adjustment and highly depends in the institution's policy.

Assuming a probability space $(\Omega, \mathcal{F})$ with Q a risk-neutral probability measure associated to a numeraire $B = (B_t)_{t \geqslant 0}$ with dynamics $dB_t = B_t r_t dt$ with r_t the short rate, the *CVA* process can be computed at each time $t \geqslant 0$ as follows :

$$CVA_t = (1 - R^C)\mathbb{E}^Q[\mathbb{1}_{t \leqslant \tau^C \leqslant T}(V_{\tau^C})^+ \frac{B_t}{B_{\tau^C}}|\mathcal{G}_t] = (1 - R^C)\mathbb{E}^Q[\int_t^T \frac{B_t}{B_s}(V_s)^+ dH_s|\mathcal{G}_t]. \quad (3)$$

with :

- R^C the *recovery rate* for the counterparty C such as $LGD = 1 - R^C$.
- V_t the product/portfolio value at time t such that $(V_t)^+$ refers to counterparty *Exposure*.
- T the maturity of the product/portfolio.
- τ^C the time default of the counterparty C and $H_t = \mathbb{1}_{\tau^C \leqslant t}$.
- $\mathcal{F}_t$ the filtration associated with the market information preventing the information of the default time of the counterparty and $\mathcal{H}_t = \sigma((H_u)_{u \leqslant t})$.
- $\mathcal{G}_t$ defined as $\mathcal{G}_t = \mathcal{F}_t \vee \mathcal{H}_t$ the lowest filtration making τ^C a stopping time.

Remark

The computation of CVA involves the computation of the portfolio value at any time which in the most common case needs to be performed using a numerical method like a Monte − Carlo procedure resulting in a nested Monte-Carlo.

By noting $G(t) = Q(\tau^C > t)$ and by supposing that τ^C admits a density probability function under Q, we can rewrite CVA_0 as follows :

$$CVA_0 = -(1 - R^C) \int_0^T \mathbb{E}^Q[\frac{(V_t)^+}{B_t}|\tau = t]dG(t). \quad (4)$$

Under independance between exposure value of the portfolio and default time, equation (4) can be rewritten over a timegrid $0 = t_0 < t_1 < \ldots < t_N = T$ by :

$$CVA_0 \approx -(1 - R^C) \sum_{i=0}^{N-1} \mathbb{E}^Q[\frac{(V_{t_i})^+}{B_{t_i}}](G(t_{i+1}) - G(t_i)). \quad (5)$$

- $\mathbb{E}^Q[\frac{(V_t)^+}{B_t}]$ is called *Expected Positive Exposure* and is noted $EPE(t)$.
- $\mathbb{E}^Q[\frac{(V_t)^-}{B_t}]$ is called *Expected Negative Exposure* and is noted $ENE(t)$.

Remark

We recover the 3 components of the credit risk in the CVA_0 expression with the the Loss Given Default (LGD) , the Probability of Default (PD) and the Exposure at Default (EAD).

Using M samples of Monte-Carlo, CVA_0 from equation (5) can be approximated as :

$$CVA_0 \approx -\frac{(1-R^C)}{M}\sum_{j=1}^{M}\sum_{i=0}^{N-1}\frac{V(t_i, X^j_{t_i})^+}{B^j_{t_i}}(G(t_{i+1})-G(t_i)) \tag{6}$$

In a standard nested Monte-Carlo framework, the quantity $V(t_i, X_{t_i^j})^+$ should be itself calculated using a MC procedure. The goal of the $\mathcal{GPR}$ will be to learn price surfaces at different dates t_i and evaluate efficiently the quantity $V(t_i, X_{t_i^j})^+$ to save one level of the nested Monte-Carlo. Our $\mathcal{GPR}-\mathcal{MC}$ estimator can therefore be defined as :

$$C\hat{V}A_0 = -\frac{(1-R^C)}{M}\sum_{j=1}^{M}\sum_{i=0}^{N-1}\frac{(\mathbb{E}[V_*|X,Y,x^*=X^j_{t_i}])^+}{B^j_{t_i}}(G(t_{i+1})-G(t_i)) \tag{7}$$

Remark

The calculation of $\mathbb{E}[V_*|X,Y,x^*=X_{t_i^j}]$ at each time-date $(t_i)_{i\in[\![0;N]\!]}$ is performed using $\mathcal{GPR}$. Therefore, we will have to train as much $\mathcal{GPR}$ as number of timesteps in the discretization of $[0,T]$. As we combined 2 numerical methods, we can take advantage of each of them. $\mathcal{GPR}$ will provide an error on EPE profile and MC an error on CVA_0.

EE computation for European Options

For the numerical results, we will consider European options under the $B-S$ model as we can have explicit formulas for EPE. First, we notice that for European options, we have :

$$\begin{aligned} EPE(t) = \mathbb{E}^Q[e^{-rt}(V_t)^+] &= \mathbb{E}^Q[e^{-rt}V_t] \\ &= \mathbb{E}^Q[e^{-rt}\mathbb{E}^Q[e^{-r(T-t)}g(S_T)|\mathcal{F}_t]] \\ &= V_0 \quad \forall t \in [0, T] \end{aligned}$$

Therefore, we see that for long positions in options, the function $t \mapsto EPE(t)$ is a constant function of t equals to the initial price of the option. Similarly, we easily see that $ENE(t) = 0$ and we expect that the $\mathcal{GPR}$ at each discretization time t_i should be able to capture this pattern.

In the following, we assume that G is given by $G(t) = e^{-\gamma t}$ where $\gamma = 0.01$ represents the default intensity of the counterparty.

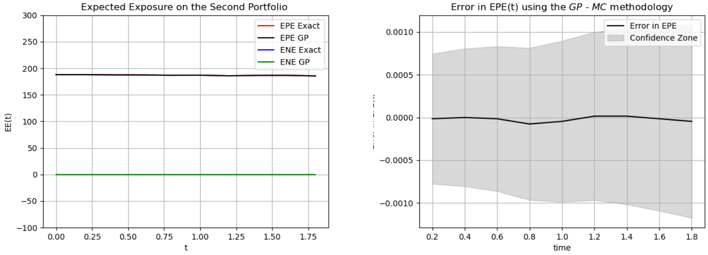


Figure: Expected Exposure Profile on a Portfolio of 10 long positions in European Call and 5 long positions in European Put using the $GP - MC$ methodology with 10 timesteps discretization for the $\mathcal{GPR}$

Table: CVA_0 using the $GP - MC$ methodology on the Portfolio with $M = 10000$ simulations

	True Value	$GP - MC$ estimation	Upper Bound	Lower Bound
CVA_0	2.2333603	2.2333624	2.2654195	2.2013054

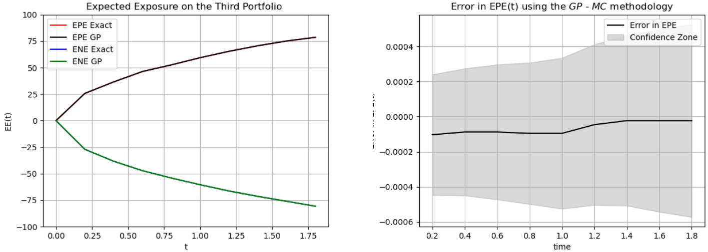


Figure: Expected Exposure Profile on a portfolio of 5 long positions in calls and 5 short positions in puts using the $GP - MC$ methodology with 10 timesteps discretization for the $\mathcal{GPR}$

Table: CVA_0 using the $GP - MC$ methodology on the the portfolio with $M = 10000$ simulations

	True Value	$GP - MC$ estimation	Upper Bound	Lower Bound
CVA_0	0.6092085	0.6092076	0.61602855	0.6023867

Pros :

- Require a really low number of training samples $(X_i, Y_i)_{i \in \mathbb{N}*}$ to learn the price surface as a function of the Markov state X.
- Provide a really accurate estimation of the EE profile with a confidence interval.
- The error in the CVA_0 computation is almost fully based on the simple Monte-Carlo loop and not in the $\mathcal{GPR}$ algorithm.
- Benefits from a simple implementation through Python packages like. See the Python notebook.

Cons :

- The learning process can be difficult when the output labels $(Y_i)_{i \in \mathbb{N}*}$ are noisy which can lead to an inefficient learning algorithm.

Sum up of the presentation :

- Study of pros and cons of the $\mathcal{GPR}$ in their ability to reproduce pricing surfaces efficiently.
- Review of the mathematical framework for CVA and the computational challenged associated to its computation.
- Study of the **GPR-MC** methodology for the fast computation of EE profile and CVA_0 computation to avoid the nested Monte-Carlo.

To go further on GPRs:

- What about pricing Options in High Dimension and of American Type ? See [5] where they combine $\mathcal{GPR}$ with Monte-Carlo and PDE methods to build efficient pricing algorithms.
- Other applications of $\mathcal{GPR}$: Calibration like in [6] or portfolio optimization like in [7]

References

- S.Crépey, M.F.Dixon, 2019, *Gaussian Process Regression for Derivative Portfolio modelling and Application to CVA Computations*, arXiv : 1901.11081.
- M.F.Dixon, I.Halperin, P.Bilokon, 2020, Machine Learning in Finance: From Theory to Practice, Springer
- C.E. Rasmussen and C.K.I. Williams, “Gaussian Processes for Machine Learning”, MIT Press 2006
- J. Gonzalvez, E. Lezmi, T. Roncalli, J. Xu : “Financial applications of Gaussian processes and Bayesian optimization"
- L.Goudenège, A.Molent, A. Zanette, "Machine Learning for Pricing American Options in High-Dimensional Markovian and non-Markovian models", 2019
- S.Gümbel, Thorstein Schmidt, "Machine learning for multiple yield curve markets: fast calibration in the Gaussian affine framework", 2020
- J.Gonzalvez, E.Lezmi, T.Roncalli, J.Xu, "“Financial applications of Gaussian processes and Bayesian optimization", 2019
- K.Barigou, D.Linders, F.Yang, ""Actuarial-consistency and two-step actuarial valuations: a new paradigm to insurance valuation", 2022