Two Machine Learning Schemes for a Class of Anticipated BSDEs

Stéphane Crépey (Université Paris Cité/LPSM),
Based on two papers involving Lokman Abbas-Turki, Bouazza Saadeddine, Shiqi Song, and Wissal Sabbagh.

Related material on https://perso.lpsm.paris/∼crepey/

The 9th International Colloquium on BSDEs and Mean Field Systems
Annecy, June 28, 2022

June 28, 2022

Outline

ABSDEs

- Anticipated backward stochastic differential equations, whose coefficient depends (in an adapted fashion) on the solution in the future
- First introduced in Peng and Yang (2009)
- Agarwal, Marco, Gobet, López-Salas, Noubiagain, and Zhou (2019) already consider an ABSDE involving a conditional expected shortfall anticipated term as we do
  - by contrast with a conditional expectation in the previous ABSDE literature.

- Exploiting the short horizon of the anticipation in their equation (one week in their case versus one year in ours) allows them devising approximations by standard BSDEs, thus avoiding the numerical problem posed by the anticipated term[1].
- Jumps that we introduce in the form of a Markov chain add a lot (of variance) to this problem

[1] cf. the beginning of Section 3.2 in Agarwal et al. (2019).

(Market and Default) Model

$\mathcal{X} = (X, J)$, for

- an $\mathbb{R}^p$ valued diffusion

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

- a $\{0,1\}^q$ valued "Markov chain like" component

$$dJ_t = \sum_{k \in \{0,1\}^q} (k - J_{t-})d\nu_t^k, \tag{2}$$

where ν_t^k counts the number of transitions of J to the state k on $(0, t]$, with compensator $\gamma_t^k dt$ of $d\nu_t^k$ such that $\gamma_t^k = \gamma_k(t, X_t)$ (for some continuous functions $\gamma_k(t, x)$).

Equation

We consider the following ABSDE for Y in $\mathcal{S}_2^l$, the space of $\mathbb{R}^l$ valued càdlàg adapted processes Y such that $\|Y\|_{\mathcal{S}_2^l}^2 = \mathbb{E}\big[\sup_{0\le t\le T} |Y_t|^2\big] < +\infty$:

$$Y_t = \mathbb{E}_t\Big[\phi(\mathcal{X}_T) + \int_t^T f\big(s, \mathcal{X}_s, Y_s, \mathbb{ES}_s(\Phi_{\bar{s}}(M))\big)\, ds\Big], \quad t \le T, \tag{3}$$

where

- $\mathbb{E}.$ and $\mathbb{ES}.$ are conditional expectation and expected shortfall[2],
- M, also required to belong to $\mathcal{S}_2^l$, is the canonical Doob-Meyer martingale component of the special semimartingale Y, and

[2] expected loss given the latter exceeds its value-at-risk at some fixed quantile level $\alpha = 97.5\%$ in our numerics.

- $\Phi_{\bar{t}}(M) := \Phi\big(t; \mathcal{X}_{[t,\bar{t}]}, M_{[t,\bar{t}]} - M_t\big)$, (e.g. $M^1_{(t+1)\wedge T} - M^1_t$), for some deterministic maps $\bar{t}$ of time t satisfying $\bar{t} \in [t, T]$ and Φ of time t and càdlàg paths $\mathbf{x}$ and $\mathbf{m}$ on $[t, \bar{t}]$ such that $\mathbf{m}_t = 0$, with Φ Lipschitz with respect to its last argument in the sense that for every $t \in [0, T]$,

$$|\Phi(t; \mathrm{x}, \mathrm{m}) - \Phi(t; \mathrm{x}, \mathrm{m}')| \leq C|\mathrm{m}_{\bar{t}} - \mathrm{m}'_{\bar{t}}| \tag{4}$$

holds for all càdlàg paths $\mathrm{x}, \mathrm{m}, \mathrm{m}'$ on $[t, \bar{t}]$ such that $\mathrm{m}_t = \mathrm{m}'_t = 0$.

Theorem 1

Under the above and otherwise standard (Lipschitz and square integrability) assumptions on the data:

- *the ABSDE* (3) *has a unique special semimartingale solution* Y *in* $\mathcal{S}_2^l$ *with martingale component* M *in* $\mathcal{S}_2^l$*;*
- *The process* Y *is the limit in* $\mathcal{S}_2^l$ *of the Picard iteration defined by* $Y^{(0)} = 0$ *and, for* $j \geq 1$,

$$Y_t^{(j)} = \mathbb{E}_t\Big[\phi(\mathcal{X}_T) + \int_t^T f\big(s, \mathcal{X}_s, Y_s^{(j-1)}, \mathbb{ES}_s(\Phi_{\bar{s}}(M^{(j-1)}))\big)\,ds\Big], \tag{5}$$

where $M^{(j-1)} \in \mathcal{S}_2^l$ *is the martingale part of the special semimartingale* $Y^{(j-1)} \in \mathcal{S}_2^l$.

- No Feynman-Kac representation for $(Y, \mathbb{ES}_\cdot(\Phi_{\bar{\cdot}}(M)))$ so far..

Outline

Outline

- $0 = t_0 < t_1 < \ldots < t_n = T$ with $\Delta t_{i+1} := t_{i+1} - t_i \le h$.
- Explicit time discretization for $(Y, \mathbb{ES}_{\cdot}(\Phi_{\bar{\cdot}}(M)))$ (with M the martingale part of the solution Y to (3)):
Process (Y^h, ρ^h) defined at grid times by $Y^h_{t_n} = \phi(\mathcal{X}^h_T), \rho^h_{t_n} = \Phi^h_T(0)$ and, for i decreasing from $n-1$ to 0,

$$
\begin{aligned}
Y^h_{t_i} &= \mathbb{E}_{t_i}\Big[Y^h_{t_{i+1}} + f\big(t_i, \mathcal{X}^h_{t_i}, Y^h_{t_{i+1}}, \rho^h_{t_{i+1}}\big)\Delta t_{i+1}\Big] \\
\rho^h_{t_i} &= \mathbb{ES}_{t_i}\Big(\Phi^h_{\bar{t}_i}\big(Y^h_{t_l} + \sum_{k<l} f(t_k, \mathcal{X}^h_{t_k}, Y^h_{t_{k+1}}, \rho^h_{t_{k+1}})\Delta t_{k+1}, l = 0, \ldots, n\big)\Big).
\end{aligned}
\tag{6}
$$

- Picard iteration associated with the implicit time discretization for $(Y, \mathbb{ES}.\Phi_{\cdot}(M))$:
  Sequence of discrete time processes $(Y^{0,h}, \rho^{0,h}) = (0, 0)$ and, for each j increasing from 1 to ∞: $Y^{j,h}_{t_n} = \phi(\mathcal{X}^h_T), \rho^{j,h}_{t_n} = \Phi^h_T(0)$ and, for i decreasing from $n-1$ to 0,

$$Y^{j,h}_{t_i} = \mathbb{E}^h_{t_i}\Big[Y^{j,h}_{t_{i+1}} + f\big(t_i, \mathcal{X}^h_{t_i}, Y^{j-1,h}_{t_i}, \rho^{j-1,h}_{t_i}\big)\Delta t_{i+1}\Big],$$

$$\rho^{j,h}_{t_i} = \mathbb{ES}^h_{t_i}\Big(\Phi^h_{\bar{t}_i}\big(Y^{j,h}_{t_l} + \sum_{k<l} f(t_k, \mathcal{X}^h_{t_k}, Y^{j-1,h}_{t_k}, \rho^{j-1,h}_{t_k})\Delta t_{k+1}, l = 0, \ldots, n\Big)$$

- The time-consistency of these schemes, i.e. the convergence of the Y^h (resp. $Y^{h,j}$) to Y as h goes to 0 (resp. h goes to 0 and j goes to infinity), can be studied by the techniques initiated in Bouchard and Touzi (2004) and Zhang (2004).
- Only partial results so far, due to the absence of a Feynman-Kac representation for the limiting ABSDE.
- Our focus hereafter is the discretization in space of (6) and (7).

Outline

Whenever a process M^h on the time grid is such that $M^h_{\{t=t_i,\ldots,\bar{t}_i\}} - M^h_{t_i}$ is a measurable functional of $(t_i, \mathcal{X}^h_{t_i}), \ldots, (\bar{t}_i, \mathcal{X}^h_{\bar{t}_i})$ with $\Phi^h_{\bar{t}}(M^h)$ square integrable, an application of Barrera, Crépey, Gobet, Nguyen, and Saadeddine (2022, Theorem 2.3)[3] yields

$$\mathbb{ES}^h_t(\Phi^h_{\bar{t}}(M^h)) = \phi^*(t, \mathcal{X}^h_t),$$

where

$$\phi^*_\cdot(t,\cdot) = \mathrm{argmin}_{\phi.(t,\cdot)\in\mathcal{B}}(1-\alpha)^{-1}\mathbb{E}[(\Phi^h_{\bar{t}}(M^h)\mathbb{1}_{\{\Phi^h_{\bar{t}}(M^h)\geq\varphi^*(t,\mathcal{X}^h_t)\}} - \phi(t,\mathcal{X}^h_t))^2$$

in which (Rockafellar and Uryasev 2000)

$$\varphi^*_\cdot(t,\cdot) = \mathrm{argmin}_{\varphi.(t,\cdot)\in\mathcal{B}}\,\mathbb{E}[(\varphi(t,\mathcal{X}^h_t) + (1-\alpha)^{-1}(\Phi^h_{\bar{t}}(M^h) - \varphi(t,\mathcal{X}^h_t))^+], \quad (9$$

both minimizations bearing over the set $\mathcal{B}$ of the Borel functions of (x, k).

[3] additionally assuming $\Phi^h_{\bar{t}}(M^h)$ atomless given $\mathfrak{F}_t$.

By (nonparametric) quantile regression estimates of $\varphi^*(t, \mathcal{X}_t^h)$ and $\phi^*(t, \mathcal{X}_t^h)$, we mean $\widehat{\varphi}^*(t, \mathcal{X}_t^h)$ and $\widehat{\phi}^*(t, \mathcal{X}_t^h)$ where the functions $\widehat{\varphi}^*(t, \cdot)$ and $\widehat{\phi}^*(t, \cdot)$ are obtained by replacing in (8)-(9):

- $\mathcal{B}$, by a to-be-specified hypothesis space of functions,
- $\mathbb{E}$, by the sample mean over a sufficiently large number of independent realizations of $\mathcal{X}^h$,
- minimization, by approximate numerical minimization through Adam stochastic gradient descent,
- φ^* in (8), by $\widehat{\varphi}^*$.

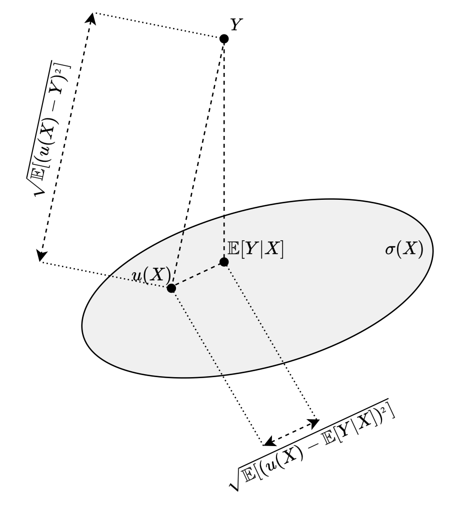
- In practice we use hypothesis spaces of functions represented by feedforward neural networks.
- Knowing a (value-at-risk) candidate $\widehat{\varphi}^*$ in neural network form, one can look for an (expected shortfall) approximator $\widehat{\phi}^*$ using a neural network with the same architecture as the one used for φ^*, set the weights of all hidden layers to those of the $\widehat{\varphi}^*$ network and then freeze them. The training of $\widehat{\phi}^*$ then falls down to a linear regression to determine the weights of the output layer.
- The fully (time and space) discrete counterparts of (6) and (7) follow by estimating, at each grid time $t = t_i$ going backward, the embedded conditional expectations (resp. expected shortfalls) through nonparametric least-squares regression against $\mathcal{X}_t^h$ (resp. quantile regression against $\mathcal{X}_t^h$ as explained above).

Outline

- Neural net parameterizations for the targeted conditional expectations and expected shortfalls lead to “nonlinear regressions” that can only be performed by numerical nonconvex minimization.
- We can state some concentration inequalities yielding a nonasymptotic a priori error control on the corresponding regression errors (at least, locally at each given time step (Barrera, Crépey, Gobet, Nguyen, and Saadeddine 2022)), but this is assuming global minimization is achieved in all training tasks.

- Instead, our focus below is on a way to a posteriori Monte Carlo validate this spatial regression error.

RSME vs. projection error in an estimation problem for $\mathbb{E}(Y|X)$



- Let $Y^{(1)}$ and $Y^{(2)}$ denote two independent copies of Y conditional on X[4]. Given any $u(X)$ (e.g. estimator of $\mathbb{E}(Y|X)$), we have by conditional independence and the tower rule:

$$\mathbb{E}[(u(X) - \mathbb{E}[Y|X])^2] = \mathbb{E}[u(X)^2 - (Y^{(1)} + Y^{(2)})u(X) + Y^{(1)}Y^{(2)}]. \quad (10)$$

- Thus, one can approximate the L_2 error of any estimator for the conditional expectation, without any knowledge on the latter, using only two inner paths (as opposed to a much heavier nested Monte Carlo).
- Extended to quantile regression in Barrera, Crépey, Gobet, Nguyen, and Saadeddine (2022, Section 4.4).

[4] The conditional independence means that for any Borel bounded functions h_1 and h_2, we have $\mathbb{E}[h_1(Y^{(1)})h_2(Y^{(2)})|X] = \mathbb{E}[h_1(Y^{(1)})|X]\mathbb{E}[h_2(Y^{(2)})|X]$.

- In a standard Λ_f-Lipschitz $f = f(t, x, y)$ BSDE time-discretizated setup (skipping $\cdot^h$ on this slide for notational simplicity), based on the estimate $\widehat{Y}_{t_{i+1}}$ for $Y_{t_{i+1}}$ at the next time step t_{i+1}, let $\epsilon_{t_i} = |\widetilde{Y}_{t_i} - \widehat{Y}_{t_i}|$, where $\widetilde{Y}_{t_i}$ and $\widehat{Y}_{t_i}$ denote the theoretical (with true conditional expectation) and empirical (with trained conditional expectation) dynamic programming estimates of Y_{t_i}.

Theorem 2

$\mathbb{E}[|Y_{t_i} - \widehat{Y}_{t_i}|] \leq \sum_{k=i}^{n-1}(1+\Lambda_f h)^{k-i}\sqrt{\mathbb{E}[\epsilon_{t_k}^2]}$, *where the*

$$\begin{aligned}\mathbb{E}[\epsilon_{t_k}^2] = \mathbb{E}\Big[&\widehat{Y}_{t_k}^2 - 2\widehat{Y}_{t_k}(\widehat{Y}_{t_{k+1}} + f(t_k, X_{t_k}, \widehat{Y}_{t_{k+1}})h) + \\ &(\widehat{Y}_{t_{k+1}}^1 + f(t_k, X_{t_k}, \widehat{Y}_{t_{k+1}}^1)h)(\widehat{Y}_{t_{k+1}}^2 + f(t_k, X_{t_k}, \widehat{Y}_{t_{k+1}}^2)h)\Big]\end{aligned} \tag{11}$$

can be estimated by Monte Carlo based on two conditionally independent[a] *copies* $\widehat{Y}_{t_{k+1}}^1$ *and* $\widehat{Y}_{t_{k+1}}^2$ *of* $\widehat{Y}_{t_{k+1}} = u_{k+1}(X_{t_{k+1}})$, *in which* u *is the regressed functional form of* $\widehat{Y}_{t_{k+1}}$.

[a] corresponding to two independent realizations of $X_{t_{k+1}}$ given the same starting point X_{t_k}.

- In the anticipated BSDE case, the analogous propagation of the local regression error terms ϵ_{t_i} and e_{t_i} (where e_{t_i} is for the expected shortfall) into global regression error controls for $\mathbb{E}[|Y^h_{t_k} - \widehat{Y}^h_{t_k}|]$ and $\mathbb{E}[|\rho^h_{t_k} - \widehat{\rho}^h_{t_k}|]$ is much more involved.

Outline

XVA Metrics in a Nutshell

What do they capture?

- CVA is the expected cost for the bank of the default risk of its clients
- FVA is the expected cost for the bank of its own default risk
  - via the implications of this risk in terms of funding spread for the bank
- KVA is the cost for the bank of having to remunerate its shareholders at some hurdle rate r for their capital at risk
  - capital required by the regulator as a safety cushion against the residual risk left uncovered by reserve capital $\mathrm{CVA} + \mathrm{FVA}$ (as default risk cannot be hedged by the bank)

Albanese, Crépey, Hoskinson, and Saadeddine (2021):

$$
\begin{aligned}
\mathrm{CVA}_t &= \mathbb{E}_t \left[\sum_c \int_t^T (\mathrm{MtM}_s^{(c)})^+ \delta_{\tau^{(c)}}(ds) \right] \\
\mathrm{FVA}_t &= \mathbb{E}_t \Big[\int_t^T \gamma_s^{(b)} \Big(\sum_c \mathbb{1}_{\{s<\tau^{(c)}\}} \mathrm{MtM}_s^{(c)} - \mathrm{CVA}_s \\
&\qquad -\mathrm{FVA}_s - \max(\mathrm{EC}_s, \mathrm{KVA}_s) \Big)^+ ds \Big] \\
\mathrm{KVA}_t &= \mathbb{E}_t \left[\int_t^T r(\mathrm{EC}_s - \mathrm{KVA}_s)^+ ds \right],
\end{aligned}
\tag{12}
$$

where $\mathrm{EC}_t = \mathbb{ES}_t[L_{(t+1)\wedge T} - L_t]$, in which the loss process L satisfies (starting from 0 at time 0)

$$
\begin{aligned}
dL_t &= \sum_c (\mathrm{MtM}_t^{(c)})^+ \delta_{\tau^{(c)}}(dt) + d\mathrm{CVA}_t + d\mathrm{FVA}_t \\
&\gamma_s^{(b)} \left(\sum_c \mathbb{1}_{\{t<\tau^{(c)}\}} \mathrm{MtM}_t^{(c)} - \mathrm{CVA}_t - \mathrm{FVA}_t - \max(\mathrm{EC}_t, \mathrm{KVA}_t) \right)^+ dt.
\end{aligned}
\tag{13}
$$

XVA Computational Strategy

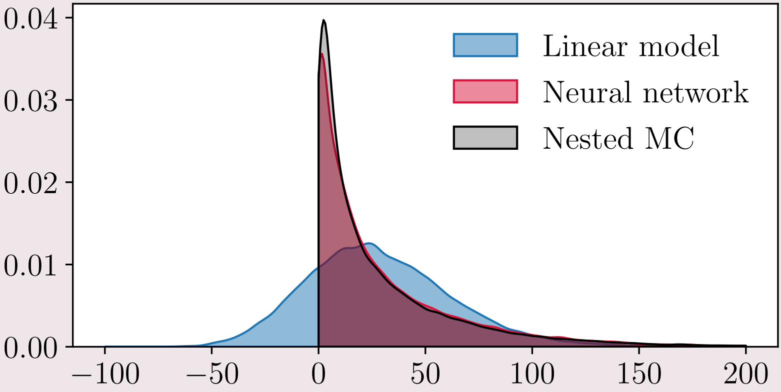
- CVA estimated first by regression-based Monte Carlo, then above numerical schemes applied for obtaining $Y = (\mathrm{FVA}, \mathrm{KVA})$
  - Hybrid market and default model $\rightarrow$ Adopting the hierarchical simulation scheme of Abbas-Turki, Crépey, and Saadeddine (2021) is key to the numerical stability of all the below-displayed results
- On the XVA side, the use of regression-based Monte Carlo simulations is not new. It was already presented in Cesari, Aquilina, and Charpillon (2010) as a key CVA computational paradigm, intended to avoid nested Monte Carlo.

- However, from such traditional XVA computations to the neural net regressions of Huge and Savine (2020), the regressions are only used for computing the $\mathrm{MtM}^{(c)}$s
    - mark-to-market cube of the prices of all the contracts of the bank with all its clients at all times of a simulation time-grid, out of which the XVAs of the bank at time 0 (and only it) are obtained by integration proxies.
- By contrast, in our case, we aim at learning the genuine XVA metrics as processes, i.e. at every node of a simulation for all risk factors
    - based on a mark-to-market cube computed by model analytics (or/and standard regressions) at the forward simulation stage.

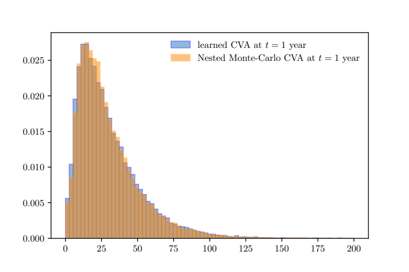
- Gnoatto, Reisinger, and Picarelli (2021) deep-hedge and learn the CVA and the FVA, but this is in a purely diffusive setup, after the default of the bank and its (assumed single) counterparty have been eliminated from the model by the reduction of filtration technique of Crépey and Song (2015).
- This technique of reduction of filtration is not extendible to the realistic case of a bank involved in transactions with several (in practice, several thousands) clients, the default times of which enter the ensuing FVA (and KVA) equations in a nonlinear fashion, so that there is then no other choice but simulating these defaults and including them in the training.
- The genuine XVA equations are not amenable to a deep-hedging approach, mainly for memory limitation reasons.

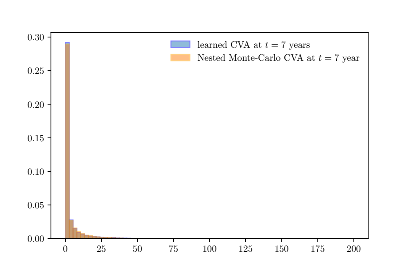
CVA Warm-Up

Density plot of the CVA of a vanilla call, at mid-life of the option.

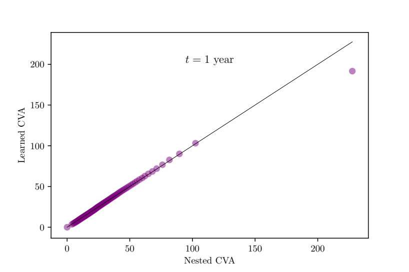


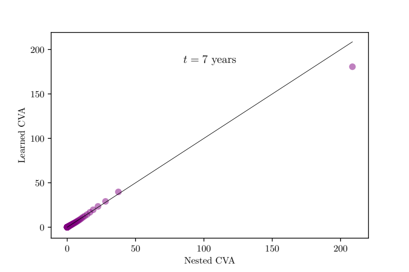
Random variables CVA_1 and CVA_7 (respectively observed after 1 and 7 years) obtained by learning (blue histogram) versus nested Monte Carlo (orange histogram). All histograms are based on out-of-sample paths.



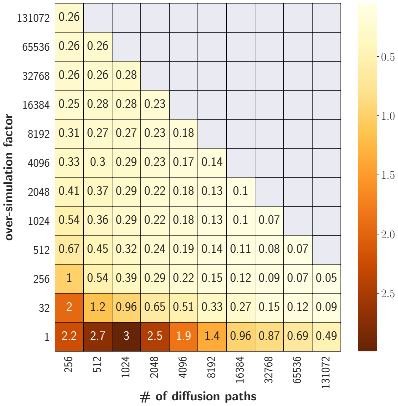


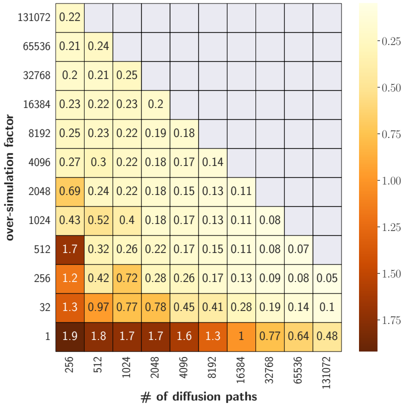
QQ-plot of learned versus nested Monte Carlo CVA for the random variables CVA_1 (*left*) and CVA_7 (*right*). Paths are out-of-sample.





Relative RMSE of the prediction against a nested Monte Carlo benchmark (left) and the prediction against the ground-truth CVA (right) at the pricing time $i = 5$ years, for different combinations of the number of market paths M and of the hierarchical simulation factor N.





XVA Case Study

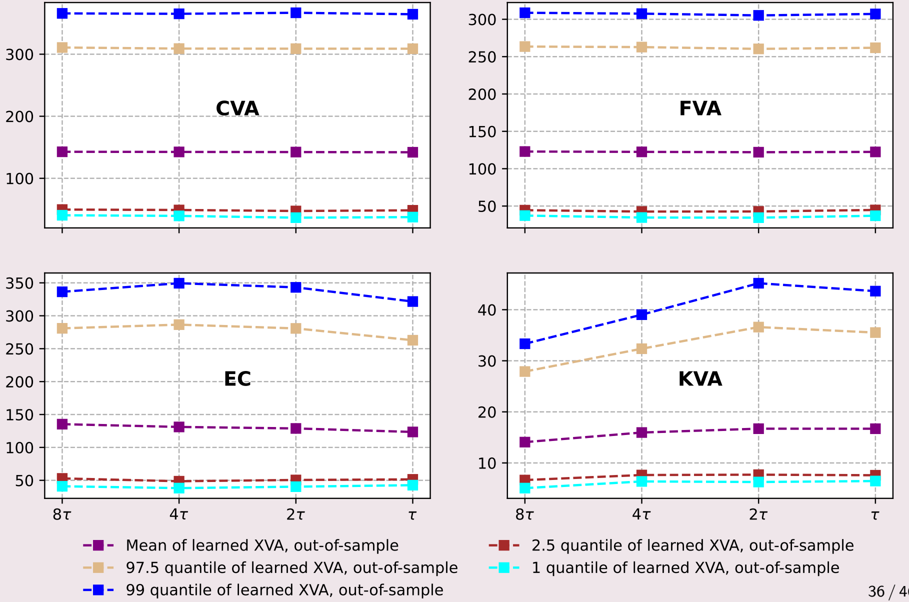
- Risk factors: 10 IR Vasicek, 9 FX GBM, 9 CR CIR and 8 default indicator processes, adding up to 36 risk factors used as deep learning features
- Portfolio of the bank comprised of 100 interest rate swaps with random characteristics (notional, maturity, counterparty and currency). In particular, their maturities are between 0.9375 and $T = 10$ years.
- Python $\xrightarrow{\text{numba}}$ CUDA GPU implementation of the simulations
  - including MtM computations
- Learning with PyTorch
  - for its proximity to the CUDA programming model

available on https://github.com/BouazzaSE/NeuralXVA

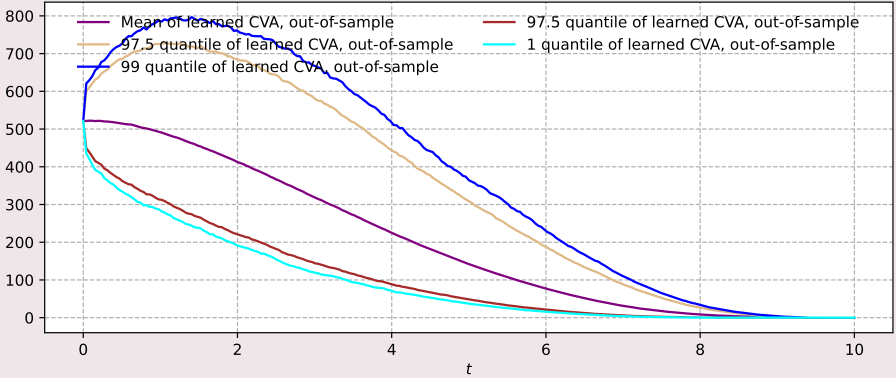
FVA_0 under the Picard iteration scheme with reuse of weigthts across time steps vs. the explicit scheme.

	$j = 1$	$j = 2$	$j = 3$	$j = 4$	Explicit
$h = \frac{T}{2^5}$	463.279938	433.832031	434.391296	433.753998	434.65167
$h = \frac{T}{2^6}$	461.329926	433.141876	434.036011	433.835052	433.60974
$h = \frac{T}{2^7}$	461.031097	432.506531	433.631531	431.789215	433.18683
$h = \frac{T}{2^8}$	460.326050	433.123596	431.992859	432.098328	434.29538

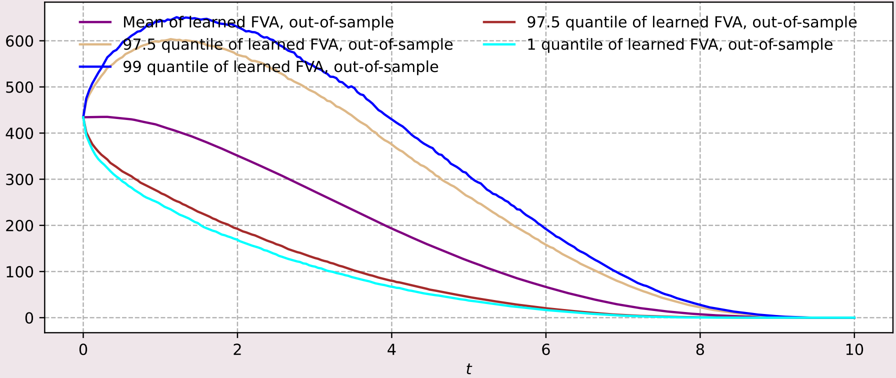
Mean and quantiles of CVA, FVA, KVA and EC learned by the explicit scheme at $t = \frac{T}{2}$ for different sizes of the time step.



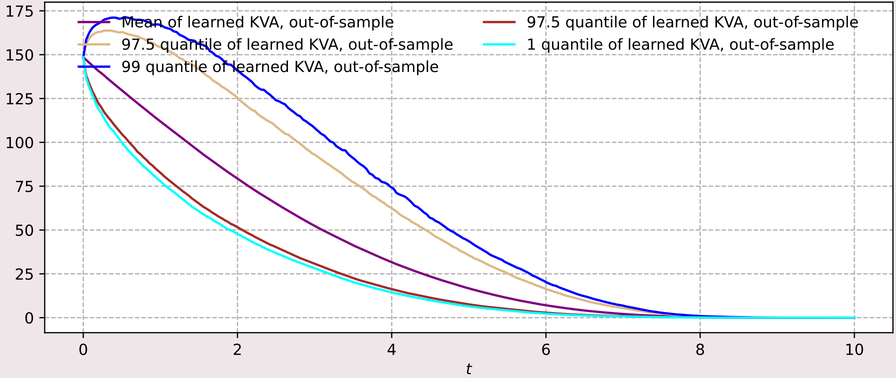
CVA profiles using an explicit scheme and a fine time discretization



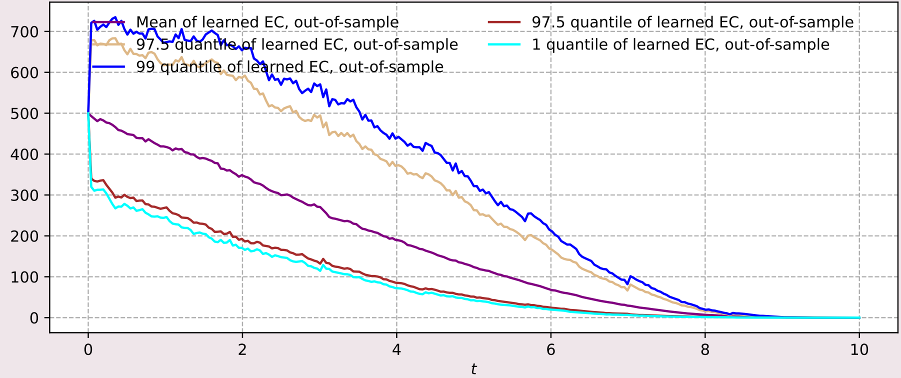
FVA profiles using an explicit scheme



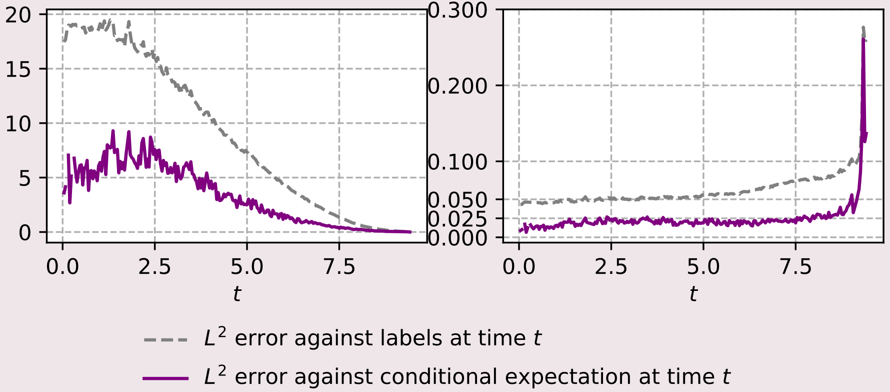
KVA profiles using an explicit scheme



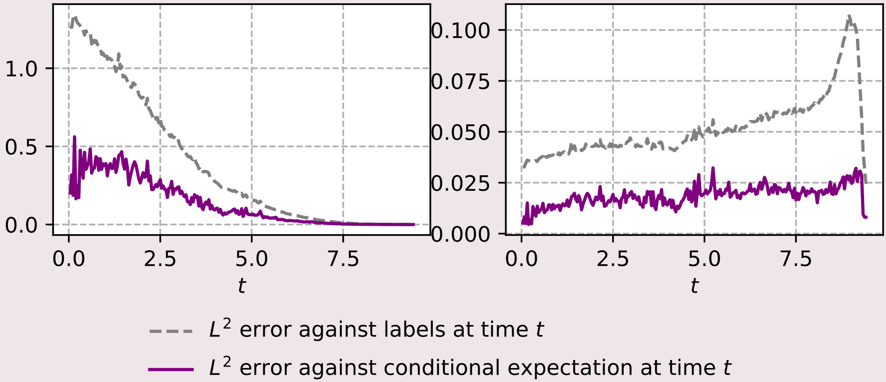
EC profiles using an explicit scheme



Local regression errors $\sqrt{\mathbb{E}[(\epsilon^{fva}_{t_i})^2]}$ (solid purple) vs. L^2 training losses (dashed grey). Left panel: raw errors. Right panel: errors normalized by the $\widehat{\mathrm{FVA}}^h_{t_i}$.



Local regression errors $\sqrt{\mathbb{E}[(\epsilon_{t_i}^{kva})^2]}$ (solid purple) vs. L^2 training losses (dashed grey). Left panel: raw errors. Right panel: errors normalized by the $\widehat{\mathrm{KVA}}_{t_i}^{h}$.



ANNOUNCEMENT

Postdoc Position in Finance and Data Science at LPSM (Paris), Team Mathematical Finance and Numerical Probability (Jean-François Chassagneux, SC, Huyen Pham, Marie-Claire Quenez,...):

- https://www.institutlouisbachelier.org/wp-content/uploads/2022/05/post-doc-announcement-09052022.pdf

Thanks for your attention!

Abbas-Turki, L., S. Crépey, and B. Saadeddine (2021).
Hierarchical simulation for learning with defaults.
Working paper available on https://perso.lpsm.paris/∼crepey.

Agarwal, A., S. D. Marco, E. Gobet, J. López-Salas, F. Noubiagain, and A. Zhou (2019).
Numerical approximations of McKean anticipative backward stochastic differential equations arising in initial margin requirements.
ESAIM: Proceedings and Surveys 65, 1–26.

Albanese, C., S. Crépey, R. Hoskinson, and B. Saadeddine (2021).
XVA analysis from the balance sheet.
Quantitative Finance 21(1), 99–123.

Barrera, D., S. Crépey, E. Gobet, H.-D. Nguyen, and B. Saadeddine (2022).
Learning value-at-risk and expected shortfall.
Working paper available at https://perso.lpsm.paris/∼crepey.

Bouchard, B. and N. Touzi (2004).

Discrete-time approximation and Monte-carlo simulation of backward stochastic differential equations.
Stochastic Processes and their applications 111(2), 175–206.

Cesari, J., J. Aquilina, and N. Charpillon (2010).
Modelling, Pricing, and Hedging Counterparty Credit Exposure.
Springer.

Crépey, S. and S. Song (2015).
BSDEs of counterparty risk.
Stochastic Processes and their Applications 125(8), 3023–3052.

Gnoatto, A., C. Reisinger, and A. Picarelli (2021).
Deep xVA solver–a neural network based counterparty credit risk management framework.
arXiv:2005.02633.

Huge, B. and A. Savine (2020).
Differential machine learning: the shape of things to come.
Risk Magazine, September.

Peng, S. and Z. Yang (2009).
Anticipated backward stochastic differential equations.
The Annals of Probability 37(3), 877–902.

Rockafellar, R. and S. Uryasev (2000).
Optimization of conditional value-at-risk.
Journal of risk 2, 21–42.

Zhang, J. (2004).
A numerical scheme for BSDEs.
The Annals of Applied Probability 14(1), 459–488.