CVA – AAD

Workshop: Advanced Techniques in Finance
14 March 2017 - Dauphine

Adil REGHAI - Othmane KETTANI

Acknowledgment & Disclaimer

- Thanks to Pr. Constantin Mellios, Marouen Messaoud and Julien Razafinimanana for useful discussions

- The opinions expressed in this presentation and on the following slides are solely those of the presenters and not necessarily those of Natixis

Summary

1 Introduction

CVA: context

- In the Basel II market risk framework, firms were required to hold capital against the volatility of their derivatives in the trading book. This was limited to the volatility of the default-free market value of derivatives, i.e. irrespective of any counterparty

- Basel II framework addressed counterparty credit risk via a combination of default risk and credit migration risk using the CCR default risk charge. This was not sufficient to capture the market risk nature of CVA risks (CVA P&L losses)

- During the crisis, CVA losses increased dramatically. A loss-attribution exercise conducted by the UK Financial Service Authority 12 on the losses incurred on their market operations by large UK banks during the period 2007-2009 concluded that CVA losses were five times the amounts of actual default losses

CVA: context

- CVA losses arise from:
  - ✓ An increase in the institution’s counterparties credit spreads
  - ✓ An increase in the institution’s exposures to its counterparties

- During the crisis, CVA losses occurred from the global deterioration of credit quality of most participants in the derivative markets

- The fact that some large banks recognized billions in CVA losses in some instances led the Basel Committee to consider CVA risks as a potential source of financial instability against which capital should be held.

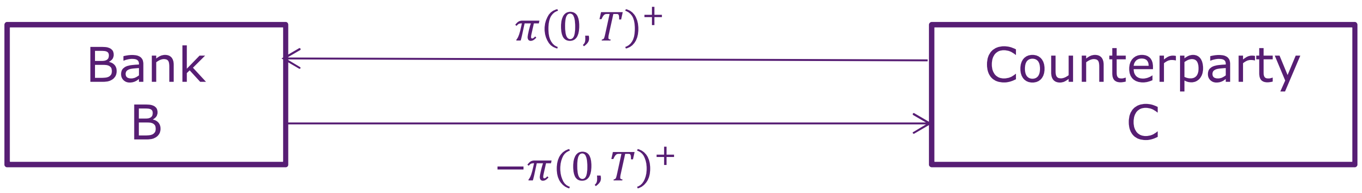
- In response, the Basel III standards introduced a capital charge against CVA risk in 2010

  ➔ Seek to capitalize against the volatility of the credit risk component of unilateral CVA (i.e. changes in the CVA due to changes in the counterparty’s credit quality)

2 The valuation framework

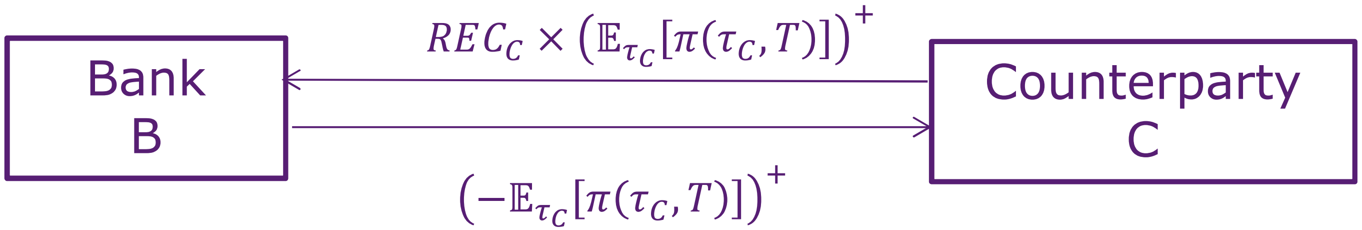
NATIXIS

The valuation framework



$\pi(0,T)^+$ represents the discounted portfolio CFs between 0 and T

When default occurs at τ_C, we have the following closeout CFs:



NATIXIS

CVA with a Risk-Free closeout

$$CVA = LGD \times \mathbb{E}\left[1_{\tau \leq T}\, D(0,\tau)\big(V(\tau)\big)^{+}\right]$$

- LGD: is the Loss Given Default. It represents the fraction of loss incurred by the investor upon default of the counterparty
- $D(u,v)$: is the discount factor between times u and v
- T: is the maturity of the contract
- τ: random variable representing default time of the counterparty
- $1_{\tau \leq T}$: equals 1 if the counterparty defaults before maturity and 0 if it doesn't.
- $\mathbb{E}$: represents the expectation operator.

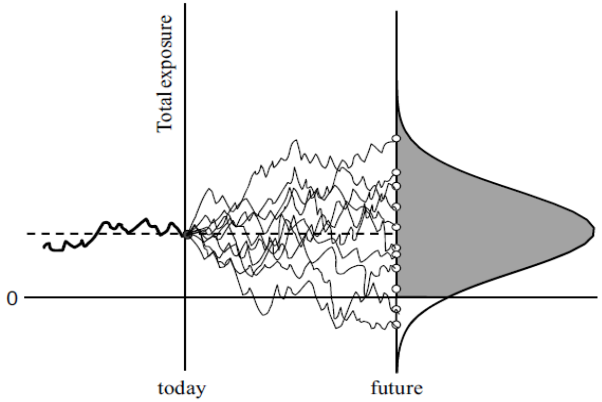
NATIXIS

CVA with a Risk-Free closeout

$$CVA = LGD \times \mathbb{E}\left[1_{\tau \leq T}\, D(0,\tau)\big(V(\tau)\big)^{+}\right]$$

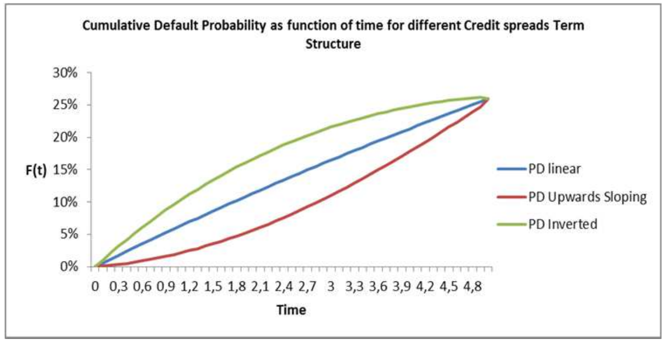
Exposures

- ✓ $Exposure = E(t) = \max(V(t), 0)$
- ✓ $EE(t) = \mathbb{E}\big(E(t)\big) = \mathbb{E}(\max(V(t), 0))$



Default probabilities

- ✓ $F_\tau(t) = \mathbb{Q}(\tau \leq t)$
- ✓ Shape of $F_\tau(t)$ + Level of credit spreads
  ➔ important impact on CVA



NATIXIS

CVA modeling

What impacts CVA?

- Model risk:
  - ✓ Underlying dynamics
  - ✓ Default dynamics
  - ✓ Correlation between the underlying and the default of the counterparty (in particular Wrong Way Risk)

- Payoff risk
  - ✓ Unilateral CVA or bilateral CVA → inclusion of DVA ?
  - ✓ First to default check
  - ✓ Inclusion of collateral in the valuation framework (liquid and secured asset or cash)
  - ✓ Netting : netting all the positions towards a counterparty in the event of default
  - ✓ Closeout conventions

Focus of this presentation: CVA for equity derivatives

Existing Approaches

CVA computation

Monte Carlo Approach

- ✓ Weighted sum of discounted Expected Exposures (EE)
- ✓ Weights are PD of CP
- ✓ Exposures distributions obtained at each time from inception to maturity
- ✓ 100 time steps – 1000 paths

➔ Heavy computation requirements!

PDE Approach

- ✓ Recently introduced (2011)
- ✓ Different closeout conventions
  ➔ Different PDEs!
- ✓ Quickly unworkable for complex trades (multi-asset portfolios)

➔ Curse of dimensionality!

The valuation framework: Assumptions

- *(H1)* A risk free closeout
- *(H2)* Independence between underlying dynamics and interest rates dynamics
- *(H3)* Random default time assumed to be exponentially distributed with a default intensity

Default probabilities of the counterparty are given by:

$$\mathbb{Q}(\tau \leq t) = \mathbb{Q}_\tau(t) = 1 - e^{-\lambda t} = F_\tau(t)$$

Where λ represents the deterministic default intensity of the counterparty, and F_τ its cumulative density function. We assume that the credit risk of the counterparty is traded so that the hazard rate λ is known at all times.

Our approach

- Both existing approaches for CVA computation suffer from serious drawbacks!

  ➔ Enhance both the flexibility (higher dimensions) and the computational burden of CVA

In this presentation, we:

- Place ourselves in a Monte Carlo setting
- Show how to approximate expected exposures at any future time using expected prices and sensitivities.
- Show how to cleverly compute sensitivities using AAD techniques.

3 CVA: A first order approximation

NATIXIS

CVA: First order approximation

- Assume we have a payoff paying Cash Flows $\boldsymbol{C_1, C_2, \ldots, C_n}$ at times $\boldsymbol{t_1, t_2, \ldots, t_n}$

- The contract depends on a number 'm' of underlying assets driven by:

$$\frac{dS_{i_t}}{S_{i_t}} = r_t dt + \sigma_i dW_i(t)$$
$$d < W_i, W_j >_t = \rho_{ij} dt$$

- The price of the contract at time u satisfies:

$$V(u) = \sum_{t_i \geq u}^{T} \mathbb{E}_{\mathrm{u}}[D(\mathrm{u}, t_i) C(t_i)]$$

- CVA at time t (time of valuation):

$$CVA_t = LGD \sum_{i=1}^{M} D(t, t_i)\, \mathbb{E}_t(V(t_i)^+)\, \mathbb{Q}(t_{i-1} \leq \tau \leq t_i)$$

NATIXIS

CVA: First order approximation

- Our idea: we regress $V(u)$ onto the variables $\left(S_{i_u}\right)_{i=1\ldots m}$ and find the unbiased first order Least Square estimator of the price $V_{LS}(u)$, which is given by:

$$V_{LS}(u) = \sum_{t_i \geq u}^{T} D(\mathrm{u}, t_i)\, \mathbb{E}_{\mathrm{t}}\big(C(t_i)\big) + \sum_{t_i \geq u}^{T} D(\mathrm{u}, t_i) \sum_{j=1}^{m} \frac{\partial \mathbb{E}_{\mathrm{t}}\big(C(t_i)\big)}{\partial F_{j_u}} * \left(S_{j_u} - F_{j_u}\right)$$

$$F_{i_u} = \mathbb{E}_t\left(S_{i_u}\right) \qquad i = 1 \ldots m$$

- This representation of $V(t)$ around its mean allows us to derive a distribution of the contract's value at time t as we write:

$$S_{i_u} - F_{i_u} \approx F_{i_u}\, \sigma_i\, W_u = F_{i_u}\, \sigma_i \sqrt{u}\, X \qquad X \sim N(0{,}1)$$

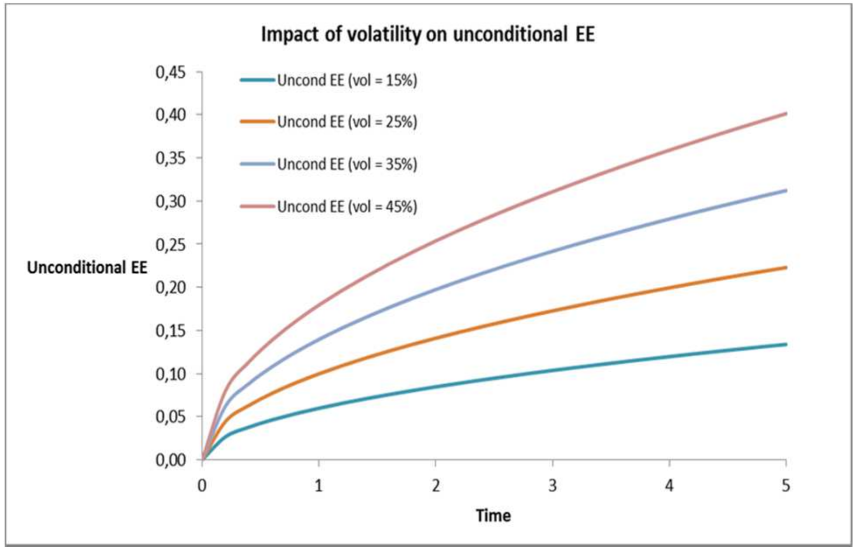
- At each time u, $V(u)$ has a normal distribution with:
  - ✓ **Mean**: $m(u) = \mathbb{E}_t(V_u) = \sum_{t_i \geq u}^{T} D(\mathrm{u}, t_i) \mathbb{E}_t\big(C(t_i)\big)$
  - ✓ **Standard Deviation**: $\alpha(u) = \sqrt{\sum_{p=1}^{m} \sum_{q=1}^{m} \gamma_p \gamma_q F_{q_u} F_{p_u} \sigma_q \sigma_p\, \rho_{qp}\, u}$ With $\gamma_p = \sum_{t_i \geq u}^{T} D(u, t_i) \frac{\partial \mathbb{E}_t\big(C(t_i)\big)}{\partial F_{p_u}}$

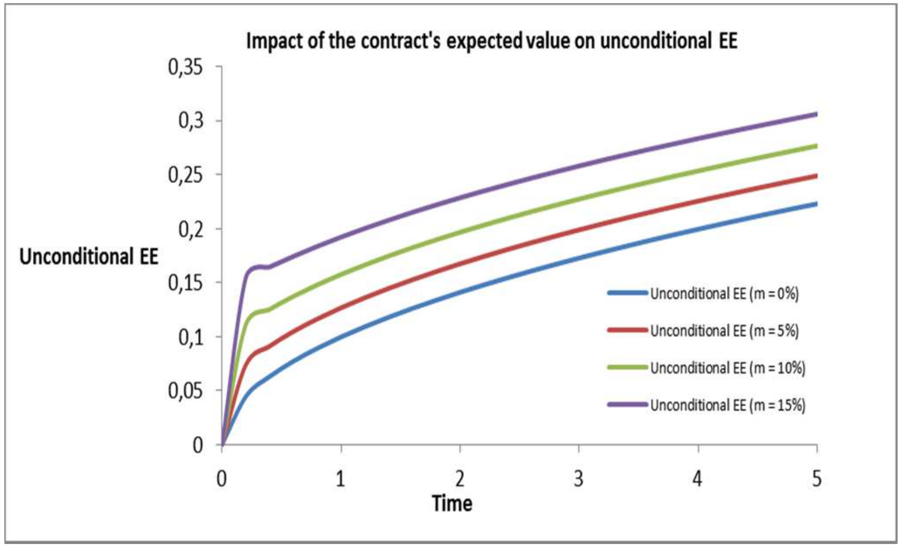
NATIXIS

CVA: First order approximation

$$\mathbb{E}_t(V(t_i)^+) = \alpha(t_i)\, p\left(\frac{m(t_i)}{\alpha(t_i)}\right) + m(t_i)\, N\left(\frac{m(t_i)}{\alpha(t_i)}\right)$$

$$p(x) = \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-\frac{x^2}{2}} \qquad N(x) = \int_{-\infty}^{x} p(u)du$$





NATIXIS

4 AAD and CVA

Adjoint Algorithmic Differentiation

- Adjoint Differentiation (Griewank 2000) refers to a set of techniques used to efficiently calculate the derivatives of functions seen as computer programs.

  Principle: "Any such function can be interpreted as a composition of basic arithmetic and intrinsic operations, easily differentiable"

- When applied to calculate sensitivities, this method turns out computationally efficient compared to standard procedures such as finite differences

- Useful case: When one needs the derivatives of a small number of outputs w.r.t a large number of inputs. In this case, the calculation can be highly optimized by applying the chain rule through the instructions of the program in reversed order with regards to its original evaluation

Adjoint Algorithmic Differentiation

- Assume we have a function $\mathrm{Y} = \mathrm{Y}(\mathrm{X})$ mapping a vector X in $\mathbb{R}^{\mathrm{n}}$ to a vector Y in $\mathbb{R}^{\mathrm{m}}$ through a sequence of steps:

$$X \to \cdots \to U \to W \to \cdots \to Y$$

Where the real vectors U and W represent intermediate variables used in the calculation, its adjoint counterpart reads :

$$\bar{X} = \bar{X}(X, \bar{Y})$$

- Where the adjoint of the output $\bar{\mathrm{Y}}$ is an arbitrary vector in $\mathbb{R}^{\mathrm{m}}$ and the adjoint of the input $\bar{\mathrm{X}}$ is given by :

$$\bar{X}_i = \sum_{j=1}^{m} \bar{Y}_j \frac{\partial Y_j}{\partial X_i}$$

- The adjoint function is implemented in a reversed order compared to its original formulation

$$\bar{X} \leftarrow \cdots \leftarrow \bar{U} \leftarrow \bar{W} \leftarrow \cdots \leftarrow \bar{Y}$$

- Where the adjoint of any intermediate U_{k} is defined as :

$$\overline{U_k} = \sum_{j=1}^{m} \bar{Y}_j \frac{\partial Y_j}{\partial U_k}$$

AAD: Example 1

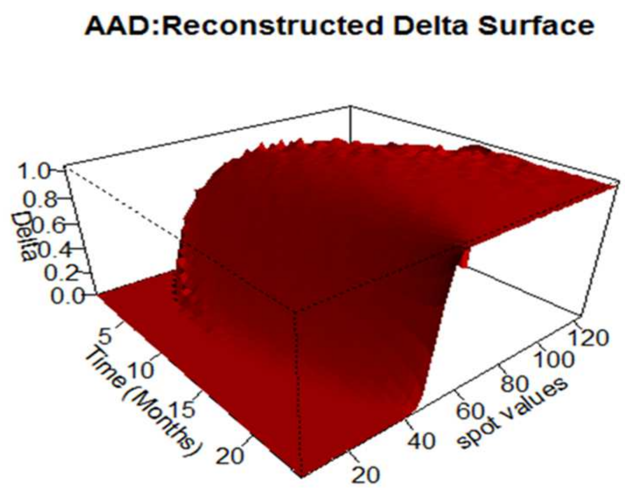
- AAD applied to a Call Option:

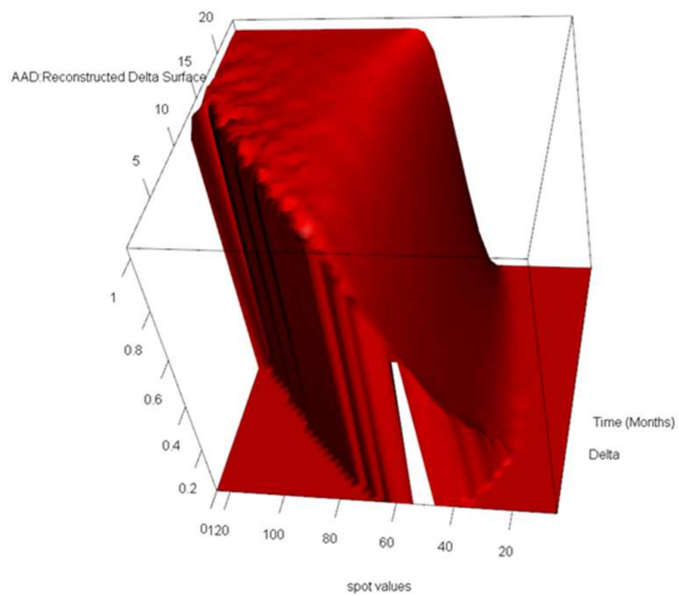
payoff<- function(logS){
 	
  S= exp(logS)
  w= S -strike
  d = exp(-r*T)
  x= d*w
  y =max(x,0)
  return(y)
}
	payoff_b <- function (y_b,logS){
	  S= exp(logS)
	  w= S -strike
	  d = exp(-r*T)
	  x= d*w
	  y =max(x,0)
	
	  x_b = d_b = w_b = S_b = logS_b =0
	  if (x>0){
	    x_b = y_b
	  }else{
	    x_b = 0
	  }
	    d_b =x_b*w
	  w_b =x_b *d
	  r_b = -T*d_b*d
	  S_b =w_b
	  logS_b= S_b*S
	  return(logS_b)
	}

NATIXIS

AAD: Example 1

- Reconstructed Delta Surface of a Call option





Delta surface obtained with the following parameters: $S_0 = 50$, $r = 2\%, \sigma = 25\%$, $T = 2y, K = 50$

AAD: Example 2 – In practice… (1/3)

Price

Pricer

- Market Data/Calibration
  - ✓ Forward
  - ✓ Dividends
  - ✓ Volatilities
- Models
  - ✓ Spot Diffusion model (BS, LV, LSV)
  - ✓ Rates model (Determinist, HJM)

+

Payoff

- Term sheet
  - ✓ Issuer
  - ✓ Notional
  - ✓ Strike date – Maturity
  - ✓ Observations Dates
  - ✓ Coupon
  - ✓ Final Redemption Amount
  - ✓ Other intrinsic information

AAD: Example 2 – In practice… (2/3)

- LV model:

$$S_{p_t} = F_{p_t} M_{p_t}$$

$$M_{p_0} = 1$$

$$M_{p_{t+dt}} = M_{p_t} e^{-\frac{\sigma^2\left(t, M_{p_t}\right)}{2} t + \sigma\left(t, M_{p_t}\right)\sqrt{dt}\, X}$$

- F_{p_t} is the forward at time t of the underlying p. M_{p_t} is the exponential martingale. It is defined recursively using the local volatility surface $\sigma(t, M_t)$.

- Only few amendments to compute CVA!

AAD: Example 2 – In practice… (3/3)

- Changes to the existing infrastructure

  ✓ **Step 1:** In the paths generating function:

$$\forall\, t_j \in [t, T], \qquad \frac{\partial M_{p_{t_j+dt}}}{\partial M_{p_{t_j}}}$$

$$\frac{\partial M_{p_{t_j+dt}}}{\partial M_{p_{t_j}}} = \frac{M_{p_{t_j+dt}}}{M_{p_{t_j}}}\left(1 + M_{p_{t_j}}\left(-\sigma\left(t_j, M_{p_{t_j}}\right)\frac{\partial \sigma}{\partial K}\left(t_j, M_{p_{t_j}}\right)dt + \frac{\partial \sigma}{\partial K}\left(t_j, M_{p_{t_j}}\right)\sqrt{dt}X\right)\right)$$

  ✓ **Step 2:** Compute (path by path, for all underlying)

$$\forall\, t_i \geq t, \qquad \frac{\partial C_{t_i}}{\partial F_{p_t}} = \frac{\partial C_{t_i}}{\partial S_{p_t}}\frac{\partial S_{p_t}}{\partial F_{p_t}} = \frac{\partial C_{t_i}}{\partial S_{p_t}} M_{p_t}$$

$$\forall\, t_i \geq t, \qquad \frac{\partial C_{t_i}}{\partial S_{p_t}} = \frac{\partial C_{t_i}}{\partial S_{p_T}} \times \frac{\partial S_{p_T}}{\partial S_{p_{T-1}}} \times \cdots \times \frac{\partial S_{p_{t+1}}}{\partial S_{p_t}}$$

$$\forall\, t_k \geq t, \frac{\partial S_{p_{t_k+1}}}{\partial S_{p_{t_k}}} = \frac{\partial S_{p_{t_k+1}}}{\partial M_{p_{t_k+1}}}\frac{\partial M_{p_{t_k+1}}}{\partial M_{p_{t_k}}}\frac{\partial M_{p_{t_k}}}{\partial S_{p_{t_k}}} = \frac{\partial M_{p_{t_k+1}}}{\partial M_{p_{t_k}}}\frac{F_{p_{t_k+1}}}{F_{p_{t_k}}}$$

  ✓ **Step 3:** Aggregate equations in Step (2) across all paths

  ✓ **Step 4:** Aggregate across all CFs

  ✓ **Step 5:** Compute CVA using formula in slide 15

AAD: Numerical results

- We computed CVA for a range of equity products. Our results are summarized in the following table

Derivative contract	Maturity (years)	ID	CVA AAD (bps)	CVA Theoretical (bps)
Call Vanilla Mono	0.76	97378730	29.12	29.29
Put Asian Multi	0.76	95548986	3.14	3.08
AutoCall[15] – Coupons part	1.07	98660678	1.28	1.85
AutoCall PDI	1.07	98660313	0	0
AutoCall – Both legs	1.07	98660509	3.6	***

	1st computation	2nd computation	3rd computation	4th computation
Calibration Time (ms)	490	387	343	359
Computation Time (ms)	2684	950	968	1027
Total time	3174	1337	1027	1386

NATIXIS

Conclusions

- We presented a new approach to compute Credit valuation Adjustment (CVA) for equity derivatives
- Future Expected Exposures (EE) are calculated using closed form solutions. EE is function of portfolio expected value (seen from pricing day) and first order sensitivities w.r.t underlying forwards
- We cleverly compute these sensitivities using AAD techniques in a Monte Carlo setting.
- With some minor amendments to the existing infrastructure, we are able to compute CVA in a computationally appealing manner thanks to AAD and save a huge computational effort.

More details on CVA pricing and AAD

- A. Reghaï, O. Kettani, C. Mellios, ***A sensitivity-based approach for CVA computation***, 2014

- A. Reghaï, O. Kettani, M. Messaoud, ***CVA with AAD and Greeks***, December 2015, *published in Risk Magazine*