Imperial College London

Department of Mathematics

Deep structural-based CVA with Wrong Way Risk

Author: Marianne Jocelyn Toscano Montoya (CID: 01894118)

A thesis submitted for the degree of

MSc in Mathematics and Finance, 2020-2021

Declaration

The work contained in this thesis is my own work unless otherwise stated.

To my loved family

mamá, papá y hermanitas

Acknowledgements

I would like to express my most sincere gratitude to my project supervisor, Professor Damiano Brigo. Thank you for guiding me through this project, for the assistance, the feedback, the valuable advice, and for being an incredible professor.

I also would like to thank MSc Luis Eduardo Pavón, for his valuable advice and all the insightful comments for this project.

I thank Banco de México for providing me with financial support for doing this master. And I also thank all the valuable people from the bank that helped me to achieve this goal.

From the bottom of my heart, I thank my family. Without them, I would have not completed this work. You are my strength and source of inspiration José, Leticia, Jessica y Johanna.

Finally, I would like to extend my gratitude to my relatives, friends, and people who gave me support and encouragement during this year. Special thanks to Luis, Leticia, Víctor, Alejandro, Alison, Noemi.

Abstract

Two important areas that have stayed mostly separate are Machine Learning and Credit Risk Valuation. This thesis explores the effectiveness of combining the rigorous mathematics of credit risk theory with the practical power, efficiency and precision of deep learning models to efficiently compute the impact of Wrong Way Risk in the CVA of a portfolio. We evaluate the capability of machine learning to learn the effect of WWR correlations in the CVA prices by training a feedforward neural network in the structural Merton Model, that correlates the market risk in the financial industry with the credit risk from the counterparty. We find, with a numerical study, that a deep neural network can learn effectively the calculation of CVA WWR for options in the context of the Merton Model.

Keywords— Unilateral CVA, Merton Model, Structural model, Wrong Way Risk, Feedforward Neural Network, Deep Learning, Firm Value Model

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis explores the effectiveness of combining the mathematical rigour of credit risk theory with the practical power, efficiency and precision of deep learning models to efficiently compute the impact of Wrong Way Risk (WWR) in the Credit Valuation Adjustment (CVA) of a portfolio. We aim to compute the unilateral CVA through a feedforward neural network trained with several CVA's. We include WWR correlation in the valuation to examine the impact of the correlation on the price. We make this approach by a numerical study, generating sufficient data that spans the vast majority of possible scenarios in the CVA variables, and using this data to make a neural network learn from them.

Credit risk valuation was becoming crucial since 2008 when it opened the door to new concepts and techniques under the necessity of achieving a price of the derivatives that incorporates all the risks involved in the transaction. However, the complexity of the models in this area is still not embraced by the practitioners in the industry. The models are genuinely complex. Introducing WWR in the computation of the CVA involves the interaction of market risks and credit risks, whose measure is complicated to incorporate. There have been several attempts to linearise the features in a WWR model, but we get an approximation error in them that we try to contain.

When we are interested in incorporate credit and market risks properly, we adopt a model. There are two types of models: the simplified models and the structural models. The first ones also called intensity models, are mostly used for relative value valuation. Contrarily, the structural approach (also known as Firm Value models), attains the risks correlation within modelling the firm value. The current work applies an structural Merton's framework.

In some cases, it is necessary to incorporate simulation methods to implement these models. We need to bear in mind the approximation error, that only could be reduced with a large number of samples, sacrificing the variance and the computing time. For Merton's model, there is a closed form solution for the CVA. However, for computing this quantity we need a numerical approximation. Frequently, it is tough to implement efficiently heavy numerical results. This thesis explains the utility of machine learning to make efficient those computations. A neural network only takes some milliseconds to run.

The deep learning approach in this research project was almost a necessity. Given the abundance of data and the computational power available, deep learning is an exceptional technique to adopt. Moreover, the exactitude and accuracy of existing models and the efficiency of delivering results make them popular in finance. It seems that deep learning will likely remain close to us for some time to come.

The main caveat on the use of deep neural networks is that these are hard to interpret. The influence of single inputs on the behaviour of the network and an understanding of their nonlinear and non-local behaviour is the subject of interpretability. As we mention briefly in the conclusions, interpretability of the neural networks used in this work can be studied in further research.

The structure of the document is as follows. Chapter 2, sets the context for the present work, introducing technical definitions of counterparty credit risk-related topics and machine learning definitions such as unilateral CVA, Firm Value Models, deep neural networks, activation functions, numerical integration. Chapter 3 contains all the information needed to explore the main topic in this thesis. We present formally the model built for this thesis; we analyse the edges and the entrails of the model and how to deal with the data generated to incorporate a neural network. In Chapter 4, we shall provide several examples of architectures to build a model, and we analyse the results. We conclude in Chapter 5.

Chapter 2

Motivations

In this section we introduce CVA, starting with the intuition of the concept and describing a scenario in which a CVA computation is presented, followed by the mathematical definition. After that, we jump into the historical reasons which the CVA has become critical in recent years. Then, we describe the most common models used in the market when computing unilateral CVA with WWR developing all these concepts with due detail. Finally, we introduce how machine learning is helping to solve some problems with high technology and computational power. In this context, we explain in precise detail the deep learning technique we are developing.

2.1 What is CVA and why do we need it?

In finance, quantitative analysts that do asset pricing work on find the fair value of the traded assets, for instance, derivatives. This task can be archived using analytical tools like arbitrage-free pricing theory. The best practices incorporates the the default risk of the parties in the price of the derivative. This default risk is measured with CVA.

2.1.1 CVA definition

We start with an intuition of the concept. CVA stands for Credit Valuation Adjustment. The name of this object suggests that a CVA measures something related to Credit. Hence, the money lent of borrowed between market participants, and the ability to payback, could be part of the CVA calculation. The word *Valuation* suggests the CVA is related to pricing. The name *Adjustment* suggest that CVA is an amendment or correction to other value.

Bearing this in mind, in order to understand the definition and inspiration to compute the CVA, lets picture a scenario between two entities; a bank and a company. Assume these two participants trade a basic instrument or a portfolio of instruments that involves exchange of money. Pretend that the agreement between them ends at a predefined future date called T. From the point of view the bank, if the bank receive a cash flow, this quantity is considered a positive cash flow and today we can see this quantity discounted to now. The money that the bank agreed to pay to the company is a negative quantity that today worth the discounted value of the negative cash flow. In a scenario that a default happens from the company before the time of maturity of the deal between bank and company, there is a procedure called "closeout". Now we need to ask how the contract can be completed including the consequences of the default event in the trade. There could be money that the bank still needs to give to the company or money that the bank will receive from the company, this is easy to measure just by adding up all the future cash flows from the time to default to the time to maturity. If this quantity is positive, the banks receives this money, otherwise the bank pays.

The problem is that if the remaining quantity in the portfolio is positive, the bank will receive just a proportion of this quantity. What proportion? it depends the recovery rate. However, if this quantity is negative, the bank will pay in full to the counterparty the money they owe. Clearly here is an asymmetry among the the two scenarios for the bank in the case the company defaults. To incorporate this asymetry in the prices the CVA exists.

Some particularities not mentioned before is that the liquidation is made at the fair value. This computation involves the risk neutral expectation of the cash flows of the portfolio. We are interested in change the payoff of the instruments in the portfolio to incorporate the credit risk of the counterparty in the price. This adjustment is called "unilateral" because we only take into account the default probability of the counterparty and assume that the bank with whom the trade is made is default-free.

The mathematical formulation of the previous reasoning is as follows.

We denote the Net Present Value of the portfolio of instruments traded between the bank and the company, seen from the point of view of the Bank at time t and with maturity T as $\texttt{NPV}_B(t,T)$. We define the $\texttt{NPV}_B(t,T) = \mathbb{E}[\Pi(t,T)|\mathcal{F}_t]$ the expected value under the risk neutral measure of the discounted portfolio of cash flows, conditional to the information up to time t. From now and onwards, the notation used to express $\mathbb{E}^{\mathbb{Q}}[\,\cdot\,|\mathcal{F}_t]$ the risk neutral expectation conditional on the $\mathcal{F}_t$ σ-field (information up to t) will be $\mathbb{E}_t[\cdot]$. $\mathcal{F}_t$ is the complete market filtration; the default-free market information up to time t and the explicit monitoring of default, up to time t. We denote $\texttt{Rec}$, the recovery rate, and $\texttt{LGD} = (1 - \texttt{Rec})$ the Loss given default. We denote as $\mathbb{1}_{\{A\}}$ the event that is one when A is true and zero otherwise. Finally, we use the notation $(\,\cdot\,)^+$ to reduce the expression $max(\,\cdot\,, 0)$.

Lets say that the counterparty defaults at a future random time τ_C and today is t. Then the portfolio of discounted cash flows is divided into two terms:

$$\begin{aligned}\Pi_B(t,T) = \;& \underbrace{\mathbb{1}_{\{\tau_C > T\}}\Pi_B(t,T)}_{\text{no default before maturity}} \\ &+ \underbrace{\mathbb{1}_{\{t<\tau_C\leq T\}}\left[\Pi_B(t,\tau_C) + D(t,\tau_C)\left(\texttt{Rec}_C(\texttt{NPV}_B(\tau_C,T))^+ - (-\texttt{NPV}_B(\tau_C,T)^+)\right]\right.}_{\text{default between today and maturity}}\end{aligned} \tag{2.1.1}$$

In case there is no default before the maturity of the contract between the default-free bank and the company, then the first element in expression (2.1.1) is basically the portfolio viewed from the point of view of the bank. In case there is a default before maturity and after today, the portfolio becomes the sum of two terms:

$$\underbrace{\Pi_B(t,\tau_C)}_{\text{portfolio up to default}} + \underbrace{D(t,\tau_C)(\overbrace{\texttt{Rec}_C(\texttt{NPV}_B(\tau_C,T))^+}^{\text{recovery rate proportion of the positive residual NPV}} - \overbrace{(-\texttt{NPV}_B(\tau_C,T)^+}^{\text{negative residual NPV}})}_{\text{remaining period from default}}$$

The first term is the portfolio until the default time, and the second term are the two discounted NPV in two different scenarios. First scenario, there is still money to receive from the counterparty so the expression indicates that the bank receive the recovery rate times the positive part of the NPV in the portfolio since the default time. Second scenario, the bank owes to the company money. For this case the NPV is negative, so -NPV is positive, and this is the quantity that the bank liquidates to the company in full.

As it can be seen, there is an asymmetry in this expression that involves default from the conterparty. If we compute the expected value of this portfolio in expression 2.1.1 we get

$$\mathbb{E}_t[\Pi_B(t,T)] = \mathbb{1}_{\{\tau_C>T\}}\mathbb{E}[\Pi_B(t,T)] - \mathbb{E}_t\left[\texttt{LGD}_C\mathbb{1}_{\{t<\tau_C\leq T\}}D(t,\tau_C)(\texttt{NPV}_B(\tau_C,T))^+\right]$$

All this, let us to go with the proper definition of unilateral CVA as presented in [14]:

Definition 2.1.1 (Unilateral Credit Valuation Adjustment)**.** *This is defined as the difference between the value of a position traded with a default-free counterparty and the value of the same position when traded with a given counterparty. Formally, if τ_C is the default time of the counterparty, the UCVA is written as*

$$\text{UCVA}_t = \mathbb{E}\left[\texttt{LGD}\mathbb{1}_{\{t<\tau_C\leq T\}}D(t,\tau_C)(\texttt{NPV}(\tau_C,T))^+|\mathcal{F}_t\right] \tag{2.1.2}$$

A few remarks of this formula and its consequences:

- The expectation in the whole expression of the definition is taken under the risk neutral measure because We are pricing.

- The loss given default (LGD) rate is arguably considered constant. For example, Altman, Resti and Sironi (2004) [1] reviewed the recovery rate in several credit risk models to address the problems that come from not considering the volatility of those rates. For simplicity, we consider the LGD a constant or deterministic number that can be taken out of the expectation in 2.1.2.

- Previously we defined `NPV` as the expected value of the portfolio cash flows. We can think of this portfolio as contained any instrument we know, such as options, forwards, swaps, bonds, etc. If we pick even the simplest asset that does not need a stochastic model to give value to those instruments, saying a simple bond, the formula 2.1.2 put this bond into a $max(\cdot, 0)$ function, starting from a random date τ_C, until the date of default. This converts the valuation formula into a option with random time.

- We need to add default risk models of the counterparty in this formula because the default time is uncertain.

- It is important to keep in mind that the complexity of the unilateral CVA expression comes with the difficulty of find a model that correlates the model used to price the option from the portfolio of instruments, and the model that determines the default risk of the counterparty. In section 3 we discuss this complexity and the possible ways to solve it, in this thesis we develop one popular model that includes the characteristics needed to incorporate these models and their correlations.

2.1.2 The impact of the crisis

The global financial crisis of 2008 is key for credit risk valuation and his evolution. We will talk about the derivatives valuation and the credit risk in two sections, the pre-crisis, and the post-crisis.

The pre-crisis

Back in 1988, the first edition of the famous book by Professor John Hull came up, *Options, Futures, and Other Derivatives* [23]. At the same time, the practitioners where constantly introducing more and more complex products with complex payoffs or events in a context that the financial derivatives world was been revolutionised after Black and Scholes proved the price formula [6] of a European option being the solution of a stochastic differential equation.

Back then, in the end of the XX century and the beginig of the XXI, valuating for example interest rate swaps was unambiguous and the discounting curve was driven by interbank rates also called LIBOR rates. In fact, all modelling approach in classic derivatives as the ones indexed to a interest rate, was simpler. There was only one curve to make discounting and protections. The swap curve was obtained by a simple process called bootstrapping[1].

In risk management topics, before the global financial crisis of 2007-2009, the banks used to believe that they were default-free. The CVA was neglected and before 2002, the first time this concept arise, it was called *counterparty risk pricing*. People in the financial sector used to put insignificant importance to the metric, furthermore, the spreads were smaller and easily neglected. The founding cost were not explicitly studied in the price valuation. In regulation material, Basel I[2] regulation had been introduced by that time, but all the management of the capital from the banks used to be considered as a *back office* function.

In definitive, a crisis does not happens suddenly. There are plenty literature that agrees with this phrase, as in Szegö (2009) [40] article or in Sorking (2009) [38] book[3].

[1]We suggest the reader [20] for bootstrap technique.

[2]The most recent publications from The Basel Committee on Banking Supervision (BCBS) is in the web page `https://www.bis.org/bcbs/index.htm` from the Bank of International Settlements.

[3]The popularity of Sorking's book and the interest of financial readers to have the answer of many more question related to uncertainty let to many writers and practitioners to go beyond and analyse the consequences of new practices. The article [22] pointed out that there is still not enough in regulatory topics.

According to [21], when the crisis began in 2007, the banks used multi-currency discounting framework frequently, with the US dollar being the currency with no basis points included and all the other curves referenced to this one. The CVA was calculated following the 2006 regulation in the Financial Accounting Standards Board (FAS) 157[4], where the wording suggested a bilateral computation of the risk (for more detail, go to section (2.1.3) in the DVA and BVA subsections).

At the same time, the Basel II regulation was already out with a regulatory slot of advice in computing the CVA. In accordance to the work of Pykhtin and Zhu (2006) [35], in Basel II framework the "minimum capital requirements for counterparty credit risk are to be calculated according to the corporate loan rules applied to the appropriate exposure at default (EAD) calculated at the netting set level". Which confirms that the CVA used to exist by that moment, but not in the same way as today is.

The post-crisis

A global crisis needed to happen to make aware all the financial market that the spreads between entities for their own risk not need to be neglected anymore. With the crisis, the spread between the LIBOR rates and OIS rates widen speedily (see Figure 2.1)as a signal of the aggravate in credit conditions in the economy. This incited the financial institutions, mainly banks, to act promptly and change the mechanism to discount and project flows. Risk-free rates started to diverge for different maturities, therefore, a multiple curves model emerged (see section of Brigo's lecture notes for a brief summary in the topic [8, The crisis (2008-current). Multiple curves]).

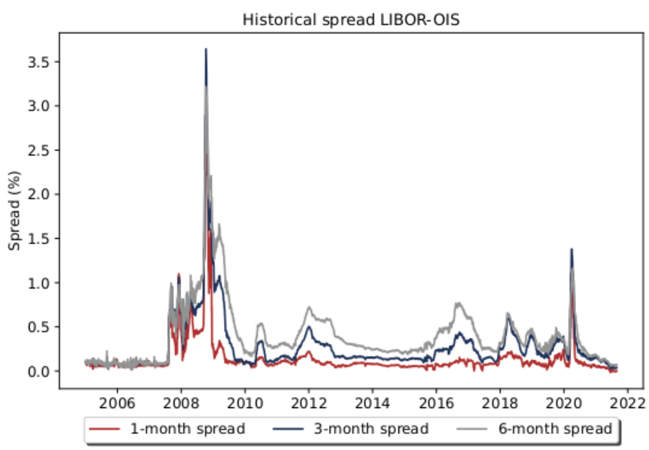


Figure 2.1: The spread between the LIBOR rates and the OIS rates for different maturities.

After a series of unfortunate events of one month, from the 7th of September of 2008 to the 8th of October 2008, in which seven firms faced severe financial problems namely credit events (Fannie Mae, Freddie Mac, Lehman Brothers, Washington Mutual, Landsbanki, Glitnir and Kaupthing) [14], many banks changed the former framework of computing unilateral CVA to a bilateral approach. Specially those with a bigger size compared to the overall banking market (tier one banks). Then banks CDS[5] reacted promptly in a path of expansion.

The consequences of including a bilateral framework in CVA on the prices where also visible. The competitiveness in prices started to incentivize banks to include DVA in pricing computations [21]. Now the CVA become an important work in the schedule of a bank. The banks needed to

[4]Consult the summary of the statement in `https://www.fasb.org/summary/stsum157.shtml` for more detail.

[5]CDS stans for Credit Default Swap. Is a derivative contract that have been designed to offer protection against default in exchange for a periodic premium.

have, not the regular trading desk that trade and hedge CVA, but instead their own trading desk in charge of the counterparty risk operations of the bank. In strictly sense, the *normal* trading desk could operate ordinarily as no crisis would happened, and the new CVA desk, could operate everything related with the pricing and risk account for the counterparty in every transaction operation.

Components of the price before and after the global financial crisis	
Pre-Crisis	**Post-Crisis**
Risk-neutral price (LIBOR discounting)	Risk-neutral price (OIS discounting)
Hedging costs	Hedging costs
CVA	CVA and DVA
Profit	Profit
	FVA (including cost of liquidity buffers)
	KVA (lifetime cost of capital)
	MVA (for initial margin)
	XVA (TVA, LVA, etc.)

Table 2.1: List of components of the price before and after the crisis of 2008[21]

.

Promptly, the regulatory requirements for capital management started to play a crucial role in the industry. New requirements and versions of Basel regimes, and also pertinent changes in accounting regulation. The structure of the banks evolved to focalize regulatory activities for capital requirement to key offices inside the firms. At the same time, new relevant metrics came out to settle a better pricing of the derivatives products, and with this modification the new XVA desks emerged. In [21] the author clearly describes the new world post-crisis:

> In 2015 pricing a "vanilla" interest rate swap involves multiple projection and discount curves for the baseline valuation and a large-scale Monte Carlo simulation at counterparty level to calculate CVA, FVA and KVA; it is a longway from the single yield curve discount models of the mid-1990s.

2.1.3 The XVA's

In this section we present the XVA family.

The CVA is included in a larger set of various Valuation Adjustments usually called the XVA models. Authors included a letter X as a generic symbol in which they incorporate all the different subjects with that an asset can be involved to take into account in the valuation. This XVA's can be as complex as one would like every time we want to incorporate metrics more rigorously in the valuation.

It is not easy to incorporate advanced metrics in the price of an asset because we need to take into account that some of them could be correlated in between. Furthermore, there could exist a relationship between them, because in finance most prices, entities or practices are intercorrelated. Obviously, the easiest way to incorporate in a price a adjustment is adding or subtracting the adjustment to the price, as

$$V^* = V + \mathrm{XVA}_1 + \mathrm{XVA}_2 + \cdots + \mathrm{XVA}_n\ , \quad n \in \mathbb{N}$$

where V^* represents the adjusted price of an asset or a portfolio of assets, V is the value pre-adjustment of the same portfolio, XVA_i is the i-th valuation adjustment. The linear world is simpler and preferable to deal with. A simple sum of factors that affect the price is easier. The complexity comes when each of these factor is difficult to obtain. In the following, we present some of the most studied and important XVA's.

CVA

We start with the already known CVA. We revisit the formula and meaning once again, this time under a more general perspective trying not to repeat the already said in the previous section when every detail about the formula was explained within a hypothetical case. This adjustment incorporates the credit risk of an entity considered defaultable. The unilateral CVA seen from the point of view of a default-free entity is

$$\mathbb{E}_t[\bar{\Pi}(t,T)] = \mathbb{E}_t[\Pi(t,T)] - \underbrace{\mathbb{E}_t\left[\mathrm{LGD}\mathbb{1}_{\{t<\tau\leq T\}}D(t,\tau)(\mathrm{NPV}(\tau),T)^+\right]}_{\text{Unilateral CVA(t,T)}} \tag{2.1.3}$$

This formula can be read as the expected value of the net cash flows of the claim seen from the point of view of the default free entity (we called this entity the bank, but not necessarily need to be a bank, could be almost any default-free investor), that trades with a defaulted counterparty (we called this the company), the valuation time is t and the maturity T is equal to the expected value of the same net cash flows portfolio this time traded with a default-free counterparty, minus the expected value of the loss given default (LGD), times the indicator function that equals one when the default time happen after the valuation date and/or before the maturity, times the discounted net present value (NPV) of the residual payoff in the portfolio of cash flows when positive.

The UCVA is an expectation of the product of four non-negative quantities; the LGD, which is a rate the indicator function, the discount factor (see definition 1.1.2 of [13]), and the positive part of the NPV. Then, when we subtract UCVA to the payoff, we reduce the price to the default-free investor when it incorporates the risk of no see finalized the contract with a counterparty that is likely to default before the contract could end successfully. Logically and mathematically, the more risky the counterparty is, the greater is the discount the bank can have to trade with this entity. In the next section, we explain the mathematical reason for this event.

DVA

We continue with another closely related adjustment, the Debit Valuation Adjustment (**DVA** for short). What happens if we ask whether the Bank can default leading to not finish the contract with its counterparty? As explained in section (2.1.2), it is not crazy to think of the possibility that a bank can default. Therefore, it is natural to ask for an adjustment in the price when is likely to not finish the contract. The popularity of the DVA came after the crisis and it deals with a own Credit Risk of the investor. DVA measures the fair quantity that an investor would accept as an increment of the price when is incorporating him as a default-risky institution.

To define DVA we start from the CVA. This adjustment seeing from the point of view of the default-risky is not a reduction in the price but a charge in the price, because seen from the counterparty point of view, the counterparty is riskier that the entity they are trading with. The CVA for the investor is the DVA for the risky counterparty. In mathematical terms the following relation is true if the bank is the default-free investo and the company is the default-risky counterparty

$$UCVA_{bank} = UDVA_{company}$$

Remember, the letter U stands for unilateral, in the sense that the valuation is made under the assumption that one entity is default-free and the other has default risk as we have said several times.

The formal mathematical expression for the DVA is

$$\mathrm{UDVA}_C(t,T) = \mathbb{E}_t\left[\mathrm{LGD}_C\mathbb{1}_{\{t<\tau_C\leq T\}}D(t,\tau_C)(-\mathrm{NPV}(\tau_C),T)^+\right] \tag{2.1.4}$$

and the price is affected with DVA as in the following equation

$$\mathbb{E}_t[\bar{\Pi}_C(t,T)] = \mathbb{E}_t[\Pi_C(t,T)] + \mathrm{UDVA}_C(t,T) \tag{2.1.5}$$

where the expression $\bar{\Pi}_C(t,T)$ refers to the adjusted net cash flows portfolio seen from the company point of view and $\Pi_C(t,T)$ is the same but before the adjustment, that means, when we are assuming a default-free scenario.

Ever since the incorporation of this debit risk measurement, a wide discussion arose around this metric. For a highly detailed reference of the features that generated all the controversy, we encourage the reader to consult [14, Section 10.5, page 253]). Here, we explain just a breve summary of the edges of the topic.

We start by the apparently greatest disvantage of the DVA. This metric induces the odd situation in which an entity makes profit when their own credit quality profile is worsen. This sounds a bit estrange but here is a basic explanation: insofar as the default-risky company trades with a bank, and this bank becomes more likely to default (this means his own credit profile is worsen), then the default term in the DVA valuation becomes larger which means that the bank is receiving a discount in his debt for become riskier. Everyone would appreciate a discount on his prices, and if the discount depends of yourself being worst payer, it is easy to get easy money then.

Another disadvantage is the hedging problem of the DVA. Remember that to hedge a derivative means that we need to sell the replicating portfolio for all possible scenarios of the derivative. In a DVA hedging process, one could sell protection against the credit risk of our own risk, another odd feature of the CVA. What does that mean? Well, imagine you are the bank. If your credit profile worsen, the charge in the price of your DVA becomes smaller. To hedge the DVA you might sell protection against you, say CDS. In the scenario of a credit event (you defaulting), you have the obligation to pay the protection you sold, but you defaulted! Who would buy this CDS from an entity that would pay when a default ocurrs? According to [14], what is most common is to do proxy hedging, meaning sell protection against my own default event, but the firms that are correlated with my credit risk.

The regulators do not agree on the use of DVA and this is also a huge problem. In the Financial Accounting Standard (FAS) 157 and in the International Accounting Standards (IAS) 39, its written that the liabilities and all financial assets should be accounted at its fair value. Recall that the computation of DVA is the expected value under the risk-neutral measure of the discounted net cash flows in the contract. No doubtely the DVA valuation corresponds to the definition of a fair value, so might be included. However, Basel III is against including DVA because they argue that the nature of the metric generate an incentive to behave badly, taking advantage of the profit that one could make when be worsen our credit profile. As an example of how delicate was the situation around Basel regulation, there is this article published in the International Investment magazine in February, 2012[6]:

> Banks could see billions of dollars removed from their stock of capital if regulators go through with a plan to exclude debit value adjustment (DVA) on derivatives portfolios from equity capital calculations.

Summarising all been said about DVA, the actual implementation of this metric lets to many doubts regarding the correctness of incorporating the adjustment in the price. Additionally, thinking of a real world situation, the unique way in which two entities could agree on the price is by both considering their own credit risks, suggesting that it might be included. It its arguably to include it and it has to be threaten with caution, but the true is that since 2006, when FAS 157 issued a requirement to introduce the CVA and the DVA reporting in accounting sheet, the DVA become more and more important.

The institutions and financial entities interpreted the accounting requirement from FAS 157 as if a bilateral model should be used. If we investigate literature about this approach we can find one of the first calculations around a bilateral credit adjustment appeared in Duffie and Huang (1996) [19] with swaps and forwards. Nonetheless, one popular article that introduces a bilateral CVA on Credit Default Swaps (CDS) on a concrete crisis situation is Brigo and Capponi (2009) [9]. The authors computed the bilateral CVA in a arbitrage-free framework. Taking into account that a CVA is a short position in a call option under the remaining (from the default time until the end of the contract) net cash flows with strike equal zero, and inversely the DVA is a long position in a put option with the same characteristics. Thus, they analysed the bilateral asymmetry situation in this framework.

[6]https://www.internationalinvestment.net/internationalinvestment/news/3704077/basel-dva-capital-deduction-cost-banks-billions

BVA

The Bilateral Valuation Adjustment (BVA) is defined as the difference between the DVA and the CVA. However, is not as simply as if we compute separately the CVA and the DVA from the point of view of an investor and then we substract the two quantities. There is a level of complexity because we need to take into account some scenarios in which the two entities default. For this analysis we assume that the probability of the two entities defaulting at the exact same time is zero (meaning, $\mathbb{Q}(\tau_B = \tau_C) = 0$).

Following the construction of [14], we consider the next six scenarios,

$$\begin{array}{ll} \mathtt{A} = \{\tau_B < \tau_C \leq T\} & \mathtt{B} = \{\tau_B \leq T \leq \tau_C\} \\ \mathtt{C} = \{\tau_C < \tau_B \leq T\} & \mathtt{D} = \{\tau_C \leq T \leq \tau_B\} \\ \mathtt{E} = \{T \leq \tau_B < \tau_C\} & \mathtt{F} = \{T \leq \tau_C < \tau_B\} \end{array}$$

where τ_B is the default time of the Bank, τ_C is the default time of the Company, T is the maturity time.

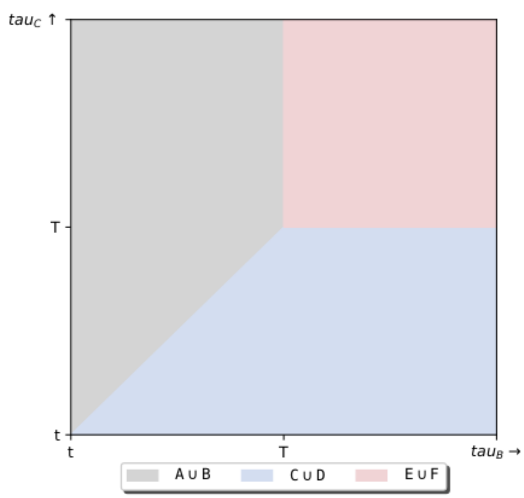


Figure 2.2: The three possible scenarios of the default of two entities B and C in a Bilateral Valuation Adjustment. Adapted from Kenyon and Stamm (2012) [24, Figure 8.1, page 141].

Observe that in scenarios $\mathtt{A}$ and $\mathtt{B}$, the first to default is the bank, whereas in scenarios $\mathtt{C}$ and $\mathtt{D}$, is the company which defaults first. In scenarios $\mathtt{E}$ and $\mathtt{F}$, the contract ends before any of the two entities could default. We can just pay attention to the intersection of the scenarios $\mathtt{A} \cup \mathtt{B}$ and $\mathtt{C} \cup \mathtt{D}$ and also threat as a unique event $\mathtt{E} \cup \mathtt{F}$ (see figure (2.2)), the event that none neither the bank nor the company, defaulted before the end of the contract.

Definition 2.1.2 (Bilateral Valuation Adjustment)**.** *When both parties in a financial contract (B and C) agrees to consider both as risky, the value of the portfolio between between them is*

$$\begin{aligned} \widehat{\Pi}_B(t,T) = \mathbb{1}_{\{\mathtt{E}\cup\mathtt{F}\}}\Pi_B(t,T) \\ &+ \mathbb{1}_{\{\mathtt{C}\cup\mathtt{D}\}}\left[\Pi_B(t,\tau_C) + D(t,\tau_C)\left(\mathtt{Rec}_C(\mathtt{NPV}_B(\tau_C,T))^+ - (-\mathtt{NPV}_B(\tau_C,T))^+\right)\right] \\ &+ \mathbb{1}_{\{\mathtt{A}\cup\mathtt{B}\}}\left[\Pi_B(t,\tau_B) + D(t,\tau_B)\left((\mathtt{NPV}_C(\tau_B,T))^+ - \mathtt{Rec}_B(-\mathtt{NPV}_C(\tau_B,T))^+\right)\right] \end{aligned}$$

and the BVA is the difference between DVA and CVA in the portfolio fair value assuming default-

free. Taking risk-neutral expectation of the above portfolio, from the point of view of B

$$\mathbb{E}_t\left[\widehat{\Pi}_B(t,T)\right] = \mathbb{E}_t\left[\Pi_B(t,T)\right] + \underbrace{\mathbb{E}\left[\mathtt{LGD}_B \cdot \mathbb{1}_{\{t<\tau^{1st}=\tau_B\le T\}} D(t,\tau_B)(-\mathtt{NPV}(\tau_B,T))^+\right]}_{\mathrm{DVA}_B(t,T)} - \underbrace{\mathbb{E}\left[\mathtt{LGD}_C \cdot \mathbb{1}_{\{t<\tau^{1st}=\tau_C\le T\}} D(t,\tau_C)(\mathtt{NPV}(\tau_C,T))^+\right]}_{\mathrm{CVA}_B(t,T)}$$

where all the symbols and notation is the same as in the previous sections τ^{1st} *is the first to default time, and the sets* A, B, C, D, E, *and* F *are as defined before.*

The proof of the definition 2.1.2 is found in [9].

Final commentaries of the BVA. Of course there hare critiques to this model, mostly related to the previously disadvantages of low point of the DVA, however, the incorporation of the bilateral credit adjustment in the prices after the crisis was a necessity and a good improvement to the later techniques, what let to agree in a fair price to trade between entities.

And more

The list of XVA's can be quite large. Explaining all of them is beyond the purpose of this document. Still, they became an important part in valuation topics since the global crises. For instance, the Funding Valuation Adjustment (FVA) aims to capture the cost of funding of the investors involved in a derivatives contract. The metric was not popular pre-crisis, although in 2011 since his creation (see table (2.2)), become a highly regarded and wide used in pricing. A few more XVA's are Capital Valuation Adjustment (KVA), Margin Valuation Adjustment (MVA), Collateral Valuation Adjustment (COLVA or OIS), Liquidity Valuation Adjustment (LVA), etc.

Derivative Valuation Adjustments XVA		
Adjustment	**Description**	**Aplicable to**
CVA (2002+)	Impact of counterparts credit risk.	Uncollateralised derivative assets.
DVA (2002+)	Benefit of the own credit risk (the 'other side' of CVA).	Uncollateralised derivative liabilities.
OIS/COLVA (2010+)	Cost of funding a collateralised derivative position, at new 'risk free' rate.	Collateralised derivative assets.
FVA (2011+)	Funding cost of uncollateralised derivatives above the 'risk free rate'.	Uncollateralised derivative assets.
KVA (2015+)	Cost of holding regulatory capital as a result of the derivative position.	Derivative contracts that are not cleared.
MVA (2015+)	Cost of posting 'initial margin' against a derivative position	Derivative contracts that are cleared.

Table 2.2: Derivative Valuation Adjustments XVA with description and product to apply. Adapted from [29].

We enccourage to the interested reader to consult the paper 'XVA explained' [29] to have a big picture of the effect and use of several XVA. We also recommend the paper by Morini and Prampolini (2010) [31] to investigate more about funding related topics and liquidity costs in derivatives valuation. Likewise, the paper by Brigo, Pallavicini and Perini [34] with a more general framework about the FVA. In collateral topics, the best paper we refer to the reader is the one by Brigo, Capponi, Pallavicini and Papatheodorou (2011) [10]. In KVA we refer the Brigo, Francischello, and Pallavicini paper [11].

2.2 Modelling counterparty default

In this section, we set the general framework to the two paradigms of modelling counterparty default. We put high emphasis in the Firm Value model, as this model corresponds to the work done in the thesis. This work ins based in [14, Part I, Chapter 3, 47-88].

2.2.1 Firm Value Models

In the family of Firm Value models, also called structural models (because they model the evolution of the firm's capital structure [2]), we have an economic reason to explain when and why happened the default time. If a company has a debt much higher than the value of the company itself, is highly probable that the company struggles when it is time to pay back the acquired debt. Then is not so strange to think of a model of the value of the company and its life closely linked to its ability to pay its debt. This reasoning has its foundations in Merton's (1974) [30] work.

The three ingredients to model structural models are the following:

- The value of the company is modeled as a stochastic process, denoted by $V(t)$ or $\{V_t\}_{t\geq 0}$.
- The debt of the company and safety covenants function, $t \mapsto H(t)$. This function occurs as a barrier in the model.
- The default time τ, that represents the first time that the value of the firm V reaches the debt H.

Suppose that a firm has a single liability with a payment day (maturity)at T. If, at maturity, the company is not able to pay its obligations, then there was a clear credit event (definition in the Appendix **??**). Therefore, the model revises only at maturity for a default. Merton adopted this approach, triggering the credit event when the firm value is below the liability at maturity. Nevertheless, some more sophisticated models seek at the end of the contract whether the obligations were filled or not and also check during the contract's life. Examples of these models are the Black and Cox Model (1976) or the AT1P Model [14, Proposition 3.1.2, pp. 57].

For the general setting in the value of the firm in this type of models, we have:

Proposition 2.2.1 (The Geometric Brownian assumption for the value process)**.** *The risk neutral dynamics for the firm value process V is characterized by a risk-free rate r_t, a payout ratio k_t and an instantaneous volatility σ_t according to the equation:*

$$dV_t = (r_t - k_t)V_t dt + \sigma_v V_t dW_t^v \tag{2.2.1}$$

For simplicity we assume that r_t and k_t are constant in time, so we can neglect the t sub index. Further work with time dependant parameters are in [14].

Under Ito's theory, a stochastic differential equation (SDE) that follows a Geometric Brownian Motion dynamics (as in (2.2.1)) has a lognormally distributed solution. The lognormal assumption in that equation is basically the heritage of Black and Scholes, and this approach has helped to model default time in credit risk several years now. Moreover, there is concrete evidence in the literature that recognize this approach "quite robust" [17], and states that the empirical data tailor the lognormal construction with its hypothesis in a acceptable manner.

Merton's model

The value of the firm follows the structure in proposition 2.2.1, with the risk free rate, the payout ratio and the volatility constant. The debt's maturity is T and the debt's face value is L. We say the company defaults if at final maturity the firm value V_T is below the debt L. For Ito's calculus we know that the solution of the SDE for the value process is

$$V(T) = V(0)\exp\left\{\left(r - k - \frac{\sigma_v^2}{2}\right)T + \sigma_v W^v(T)\right\} \tag{2.2.2}$$

Using the fact that if $X \sim lognormal$ then $\log(X) \sim normal$, then define the random variable $N_1 \sim \mathcal{N}(0,1)$

$$\log V(T) = \log V(0) \ + \left(r - k - \frac{\sigma_v^2}{2}\right) T + \sigma_v \sqrt{T} N_1 \tag{2.2.3}$$

with N_1 being a standard normal random variable.

Thus, we can measure the probability of the default event

$$\begin{aligned} \{V(T) \leq L\} &= \{\log V(T) \leq \log L\} \\ &= \left\{ \log V(0) \ + \left(r - k - \frac{\sigma_v^2}{2}\right) T + \sigma_v \sqrt{T} N_1 \leq \log L \right\} \\ &= \left\{ N_1 \leq \frac{\log\left(\frac{L}{V(0)}\right) \ + \left(r - k - \frac{\sigma_v^2}{2}\right) T}{\sigma_v \sqrt{T}} \right\} \end{aligned} \tag{2.2.4}$$

using the risk neutral measure and the knowledge of the normal random variable

$$\mathbb{Q} \left\{ N_1 \leq \frac{\log\left(\frac{L}{V(0)}\right) \ + \left(r - k - \frac{\sigma_v^2}{2}\right) T}{\sigma_v \sqrt{T}} \right\} = \Phi \left(\frac{\log\left(\frac{L}{V(0)}\right) \ + \left(r - k - \frac{\sigma_v^2}{2}\right) T}{\sigma_v \sqrt{T}} \right) \tag{2.2.5}$$

where Φ is the cumulative distribution function (CDF) of the standard normal $N_1(0,1)$.

Some points to consider with respect to equation (2.2.5) are the following:

- Inside the cumulative distribution function r, k, and σ are constants over time. The cumulative distribution function is an increasing function, then $\log(L/V(0))$ is equivalently to $\log(L) - \log(V(0))$. This, last expression is the addition of two log functions, one increasing in L and the other one decreasing in $V(0)$. This means that the bigger the debt is, the highest probability of default. At the same time, the higher the initial firm value is, the less probable the default is.

- When $V(0) \to +\infty$ and *ceteris paribus*, the log function inside the CDF $(\log(L) - \log(V(0))) \to -\infty$. Hence, $\lim \Phi(x)$ when $x \to -\infty$ is zero. The interpretation: when the initial firm value is gigantic (super super big), then the probability of default is zero, so it is no likely a default.

- When $L \to +\infty$ and *ceteris paribus*, the log function inside the CDF $(\log(L) - \log(V(0))) \to \infty$. Hence, $\lim \Phi(x)$ when $x \to \infty$ is one. This could be economically interpreted as when the the debt is too much higher than the initial firm value, almost surely there is going to be a default.

The term *hazard rate* is referred to the intensity function $t \mapsto \lambda(t)$ in the intensity models (see next section). In firm value models, the hazard rate is defined as

$$\lim_{T \to 0} \frac{\mathbb{Q}[\tau \leq T]}{T}$$

For the Merton's model we do the hazard rate computation in a particular case when r, k, and σ are zero constants.

$$\lim_{T\to 0}\frac{\mathbb{Q}\{\tau \le T\}}{T} = \lim_{T\to 0}\frac{\mathbb{Q}\{V(T)\le L\}}{T} \tag{2.2.6}$$

$$= \lim_{T\to 0}\frac{\mathbb{Q}\left\{N_1 \le \frac{\log\left(\frac{L}{V(0)}\right)}{\sigma_v\sqrt{T}}\right\}}{T}$$

$$= \lim_{T\to 0}\frac{\Phi\left(\frac{\log\left(\frac{L}{V(0)}\right)}{\sigma_v\sqrt{T}}\right)}{T}$$

$$= \lim_{T\to 0}\frac{\phi\left(\frac{\log\left(\frac{L}{V(0)}\right)}{\sigma_v\sqrt{T}}\right)\left(-\frac{1}{2T\sqrt{T}}\frac{\log\left(\frac{L}{V(0)}\right)}{\sigma_v}\right)}{1} = 0 \tag{2.2.7}$$

where the equation (2.2.6) is the definition in Merton's Model of the hazard rate, the equality (2.2.7) is obtained with L'Hopital theorem after we notice that the previous limit is a indetermination of the type *zero over zero*. We denote $\phi(\,\cdot\,)$ the probability distribution function (PDF) of a normal standard.

So, the hazard rate is zero in Merton's model. A feature not always desirable in credit risk models, considering that this characteristic is going generate a severe complication when trying to calibrate probabilities of default in short term maturities. Later on, in the next part of models presented here to compute credit default, the intensity models show up with the hazard rate being non-zero even in very short maturities (instantaneous rates). For now we disclose that the main drawback of the structural models, in specific the Merton model, is its ineffectiveness of generate short-term credit spreads.

Before introducing additional models in this structured models type, we point out another attribute of Merton's approach.

The debt value at time $t < T$ is

$$\begin{aligned}
D(t) &= \mathbb{E}[D(t,T)\min(V_T, L)] \\
&= \mathbb{E}[D(t,T)[V_T - (V_T - L)^+]] \\
&= \mathbb{E}[D(t,T)[L - (L - V_T)^+]] \\
&= \mathbb{E}[D(t,T)L] - \mathbb{E}[D(t,T)(L - V_T)^+] \\
&= P(t,T)L - \mathrm{Put}_t(T; V_T, L)
\end{aligned}$$

and assuming that the value of the firm is equivalent to add the debt of the firm and the equity vale, then

$$V(t) = D(t) + S(t) \tag{2.2.8}$$

we have that the equity value is a call option, saying

$$\begin{aligned}
S(t) &= V(t) - D(t) \qquad (2.2.9)\\
&= V(t) - [P(t,T)L - \mathrm{Put}_t(T; V_T, L)] \\
&= V(t) - P(t,T)L + \mathrm{Put}_t(T; V_T, L) \\
&= \mathrm{Call}_t(T; V_T, L) \\
&= V(t)\Phi(d_1) - P(t,T)L\Phi(d_2)
\end{aligned}$$

where

$$\begin{aligned}
d_1 &= \frac{\log\left(\frac{V(t)}{L}\right) + \left(r - k + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}} \\
d_2 &= d_1 - \sigma\sqrt{T-t}
\end{aligned}$$

we assumed determinant interest rates $P(t,T) = \exp(-r(T-t))$, we denoted $\Phi(\,\cdot\,)$ the CDF of a normal standard, we denoted $\mathrm{Call}_t(T; V_T, L)$ and $\mathrm{Put}_t(T; V_T, L)$ the Call (Put) option at time

t, with maturity T, underlying $V(T)$ and strike L. Note that in (2.2.9) we incorporate the price of a plain vanilla call under Black-Scholes-Merton framework. Also note that we used the put-call parity for European options on non-dividend paying stocks. You can see the same construction with different approach in [27] or [26].

We accept as given the equation (2.2.8) and we refer to the *Enterprise Value* definitions [7] for more detail about this economic supposition.

An additional hitch of Merton's model is related to its null ability to identify a default previous the maturity of the debt. In this sense, Kwok (2008) [26] claims that the inexistent surprise in the occurrence of the default is not observable in the real world. The basis of the model is originated in a diffusion process in a finite time setting that just can be analysed that reaches the barrier debt until the very end.

There are some improvements to the model visited in this section. For instance, the likewise classic model proposed by Black and Cox (1976) [7]. The model by Kijima and Suzuki (2001) [25] involves a jump-difussion process. More recently, the models AT1P Model and SBTV Model [14] that corresponds to improvements to the Black-Cox's model.

Black and Cox Model

We will give a brief introduction of the Black and Cox's model originally presented in [7]. This classical model could be useful to see to get a better understanding of the firm value models. The content in this section is primarly based on [14] and we suggest consult the book for more detail and concrete examples.

The Black and Cox model introduces the safety covenants[8] as a determinant feature to indicate when the firm provokes an early default while touching this barrier or "safety level" $H(t)$.

The default time τ can be defined as

$$\tau = \inf\{t \geq 0 : V(t) \leq H(t)\}$$

namely, the first time the function V hits the barrier H (hence the term *first passage models*[14]).

The value process is as in (2.2.1), we rewrite the process with constant parameters

$$dV(t) = (r - k)V(t)dt + \sigma_v V(t)dW^v(t) \tag{2.2.10}$$

The safety level is

$$H(t) = \begin{cases} L, & t = T \\ Ke^{-\gamma(T-t)}, & t < T \end{cases} \tag{2.2.11}$$

assuming that $Ke^{-\gamma(T-t)} < Le^{-r(T-t)}$. With γ, K positive parameters.

When the dynamics parameters are constant the default probabilities can be expressed in a closed form (see [4] or [14]).

AT1P Model

One brilliant extension of Black and Cox's model is the ATP1 Model, that stands for Analytically-Tractable First Passage Model [14].

The model capture the essence of Black and Cox, but adds attributes of relatively high importance in modelling the default scenarios of a firm. For instance, the behaviour of the barrier function $H(t)$ has an economic interpretation. H is a functional dependant on three elements of the company: the level of liabilities, its safety covenants and its characteristics of the capital structure. As the same time, in AT1P model the barrier can be curved shaped with volatility time dependence, desired characteristics that approximates to more real world cases.

The models is stated below. For a proof of the proposition see the paper [15].

[7] `https://valuechasing.com/what-is-enterprise-value-ev/`

[8] A covenant is a promise in an indenture, or any other formal debt agreement, that certain activities will or will not be carried out or that certain thresholds will be met.

Proposition 2.2.2 (Analytycally-Tractable First Passage (AT1P) Model)**.** *The risk neutral dynamics for the value of the firm follows the dynamics of* (2.2.1)*. The default barrier $H(t)$ is*

$$H(t) = H \exp\left(\int_0^t (r_u - k_u - B\sigma_u^2)du\right) \tag{2.2.12}$$

with two parameters H and B. Define τ as the first time the value of the firm hits the default barrier from above,

$$\tau = \inf\{t \geq 0 : V(t) \leq H(t)\}$$

Thus, the survival probability is given by

$$\mathbb{Q}\{\tau > T\} = \Phi\left(\frac{\log\left(\frac{V_0}{H}\right) + \frac{2B-1}{2}\int_0^T \sigma_u^2 du}{\sqrt{\int_0^T \sigma_u^2 du}}\right) - \left(\frac{H}{V_0}\right)^{2B-1} \Phi\left(\frac{\log\left(\frac{H}{V_0}\right) + \frac{2B-1}{2}\int_0^T \sigma_u^2 du}{\sqrt{\int_0^T \sigma_u^2 du}}\right) \tag{2.2.13}$$

2.2.2 Intensity models

The family of intensity models (or reduced form models) is conformed for those whose default time τ is the first jump of a stochastic process with deterministic or stochastic parameter. The Poisson process is a jump process whose parameter is called the intensity and the models are commonly modeled with this processes.

The default is triggered by exogenous reasons, unlike the firm value models. Furthermore, the default time is not explained by any economic reason, but its merely a function that can be calibrated easily with market data and this models are very good suited for credit spreads.

The mathematical formulation for the default time follow the following structure of default probability under the risk neutral world

$$\mathbb{Q}\{\tau \in [t, t + dt)|\tau > t, \text{ market information up to time } t\} = \lambda(t)dt \tag{2.2.14}$$

where the function λ is describing the probability of default in the instant dt. We assume that the firm have not defaulted before time t. For simplicity lets assume $\lambda > 0$. We call this function the intensity or the hazard rate.

If we accumulate the hazard rate we obtain a function called *cumulated intensity(hazard rate)* but also is commonly called *Hazard function*

$$\Lambda(t) := \int_0^t \lambda(s)ds$$

As there is a Poisson process the one that describes the default time in a jump, we can apply the hazard function to τ to obtain a random variable which distribution we know, saying

$$\Lambda(\tau) =: \xi$$

where ξ is a exponential random variable with mean equal to one. Or seen the other way round

$$\tau = \Lambda^{-1}(\xi)$$

2.2.3 Differences between Firm Value models and Intensity models

To summarise this section, we present a comparative table that highlights the main features of Merton's Firm Value model and the general intensity models. We choose to work on a Merton's models setting for this research notwithstanding that the work developed in this document is pioneer to work on many more variants of credit default models.

Comparative table between the most popular credit default time models	
Firm Value models (Merton's case)	**Intensity models**
Also called structured models	Also called reduced form models
Suited to fundamental valuation	Suited to relative value
There is economic reasoning that explains the default time	No economic reasoning whatsoever
The default probability is the probability of the event $\{V(T) \leq L\}$ (the firm value being below the debt at T)	The default probability has two interpretations: a local default probability $\lambda_t dt$ and an instantaneous credit spread
The firm value is lognormally distributed (a GBM dynamics) and the debt can be obtained by analysing the spreadsheets	The hazard function is a Poisson process with exponential jump to default
Firm value is not a tradable asset unless it is modelled as one, but there is evidence that the model is robust enough	The model is robust enough
The default time can only happen at maturity	Default happen randomly in the time
Short term default spread is almost impossible to obtain	The intensity is well defined for any term
The intensity is the constant zero	The intensity can be constant, deterministic and random.
The survival probabilities are in many cases analytically explicit	The survival probabilities from default behave exactly as zero coupon prices in interest rate theory
The equity level is a call option on the firm value	The price of a defaultable bond is like a default-free bandwidth an interest rate r and a spread λ
There is no surprise when the default is going to happen	The jump to default ξ is totally unpredictable

Table 2.3: Comparative table of the principal characteristics of Merton's model as part of the family of Firm Value Models and of a general Intensity model.

2.3 Machine Learning in Finance

Machine learning is the one of the most attractive area within Artificial Intelligence. In fact, in recent years there has been an increasing focus on some machine learning techniques within finance. This section gives a brief summary of the impact of this method of data analysis and focuses on deep learning. The content of this section is inspired by the work of [33] and [42].

Certainly exists a variety of machine learning models and many of them had had a huge impact in Finance through solving problems for the processes improvement. Some examples: in costumer service, there are classification models that help to predict the costumer preferences. There are also models for predicting the value of a variable, and with these predictions the decision making process is easier. In derivatives pricing, the models that construct a dynamic hedging. In trading, there are new intelligent algorithms that work for optimal execution. In retail banking, to detect fraud. In more general and technological topics, there are sequential decision processes that work on self-driving cars.

The main algorithm types in machine learning according to [44] are: Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, transduction and learning to learn. Inside machine learning, there are various methods: Random Forests, Support Vector Machines, Gaussian Process and Deep Learning [33]. This last one is the most famous subfield of machine learning and the one that this thesis is focused on.

In the successive of this section, we will introduce the neural networks in deep learning, starting from include some historic context and all the relevant definitions to understand the technique developed later on.

2.3.1 Deep Learning

The new researches in Deep learning are still part of the state of the art in quantitative finance. Since some years now, deep learning is a hyped area within Machine Learning and the new advances are still flourishing with success.

Why deep learning?

Deep learning is been around since many years now. However, deep learning is in his best years. Below we listed the principal reasons that deep learning is taking part of an important amount of research in the academia and applications within the industry practitioners:

- *We are in the big data era.* There are a number of sources of data, and this number is been increasing in the resent years. The source of data could come from technological devises or due to the digitization of new technologies. Deep learning in fact employs big amounts of data to work.
- *The computational power.* As the time goes by, the ability of machines to support abundant information in a reduced space is higher. Computational power grows, evolves and improves, permitting to reach more deep learning tasks.
- *Algorithmic, software and research innovation* Modern open source libraries (for example, *Keras*[9] with *TensorFlow*, *PyTorch*[10], etc), as well as constant research development.

A brief history of deep learning

We present a short timeline that aims to give a historical context and perspective of deep learning:

1943 The first neuron model by McCulloch-Pitts.

1958 Rosenblatt introduces the perceptron.

1970's Minsky and Papert introduced its book *Perceptrons: an introduction to computational geometry*, where demonstrated some limitations of the perceptron model, generating discussion within the Artificial Intelligence community and a declined interest.

1980's The improvements in digital computers returned the interest of these models.

1986 Rumelhart presented the backpropagation algorithm for neuron networks in its paper *Learning representations by back-propagating errors.*

1986 Le Cun presented an application of backpropagation procedure to hand-written digit recognition[11].

mid 1990's Second decline of deep learning. Prevailing other methods such as kernels and graphical models.

2006 and 2009 New interest in neural networks. ImageNet released.

[9] `https://keras.io/about/`
[10] `https://pytorch.org/`
[11] `http://yann.lecun.com/exdb/publis/pdf/lecun-90c.pdf`

2015 AlphaGo, a computer program beat the best Go player in the world [18]. No computer had been able to do so.

There are many miles travelled, but there is still much more to do in this field. The next part presents the mathematical formulation of the neural networks and the fundamentals to understand their application.

2.3.2 Artificial Neural Networks

Artificial neural networks (sometimes just neural networks) are functions constructed by the alternate composition of two types of functions; affine functions (linear function plus a constant) and strictly non-linear functions.

Feedforward Neural Networks

Definition 2.3.1 (Feedforward Neural Network)**.** *Is a function $f : \mathbb{R}^I \to \mathbb{R}^O$ that can be expressed as a composition of two types of functions*

$$f = \sigma_r \circ L_r \circ \cdots \circ \sigma_1 \circ L_1 \tag{2.3.1}$$

where, the dimension I refers to the dimension of the input data, the dimension O refers to the dimension of the output data, for each $i \in \{1, \ldots, r\}$, L_i is an affine function $L_i : \mathbb{R}^{d_{i-1}} \to \mathbb{R}^{d_i}$ and can be written as

$$L_i(\bar{x}) = W^i\bar{x} + b^i, \;\; \bar{x} \in= (x_1, ..., x_{d_i})\mathbb{R}^{d_i},$$

with $d_0 = I$ and $d_r = O$, and the function $\sigma_i : \mathbb{R}^{d_i} \to \mathbb{R}^{d_i}$ is called the activation function

$$\sigma_i(\bar{x}) := (\sigma_i(x_1), ..., \sigma_i(x_{d_i}))\,, \;\; \bar{x} = (x_1, ..., x_{d_i}) \in \mathbb{R}^{d_i}.$$

applied component-wise in the d_i dimensional vector.

The figure 2.3 corresponds to a graphical representation of a neural network with four *layers* (the columns of the structure), of which the first one and the last one are the input and output layers, respectively. The two in the middle are the *hidden layers*. The input layer contains three *units* (also called neurons), the first hidden layer contains seven units, the second hidden layer contains five units, and the output layer have two units.

This construction is consistently considered powerful. Furthermore, there is a mathematical foundation called the *Universal approximation property*, that gives solid foundations to the functional structure that builds deep learning. This principle can be quoted in simple words here:

Any "reasonably" function can be approximated by a suitable neural network

We refer the reader to the paper of Leshno *et al* (1993) [28] where is the proof of the universal approximation property in the general case.

This research applies feedforward neural networks (for short FNN), although the world of NN structures is big. We recommend the reader have a look at the diagram "A mostly complete chart of Neural Networks" in the article *The mostly complete chart of Neural Networks, explained*[12] (this diagram also present in the thesis [41, Apendix A]). This diagram is a big picture of the variety of neural network models. The cited article highlights the relevant aspects of each neural network shape.

The notation we use to describe an artificial neural network [33] is the following:

Definition 2.3.2 (The class of feedforward neural networks)**.** *We denote the class of such functions as*

$$\mathcal{N}_r(I, d_1, \ldots, d_{r-1}, O; \sigma_1, \ldots, \sigma_r) \tag{2.3.2}$$

where $I, d_1, \ldots, d_{r-1}, O$ indicates the number of units in each layer, starting from I the input layer and finalizing in O the output layer. And then $\sigma_1, \ldots, \sigma_r$ represents the activation functions that characterizes the neural network.

[12] https://towardsdatascience.com/the-mostly-complete-chart-of-neural-networks-explained-3fb6f2367464

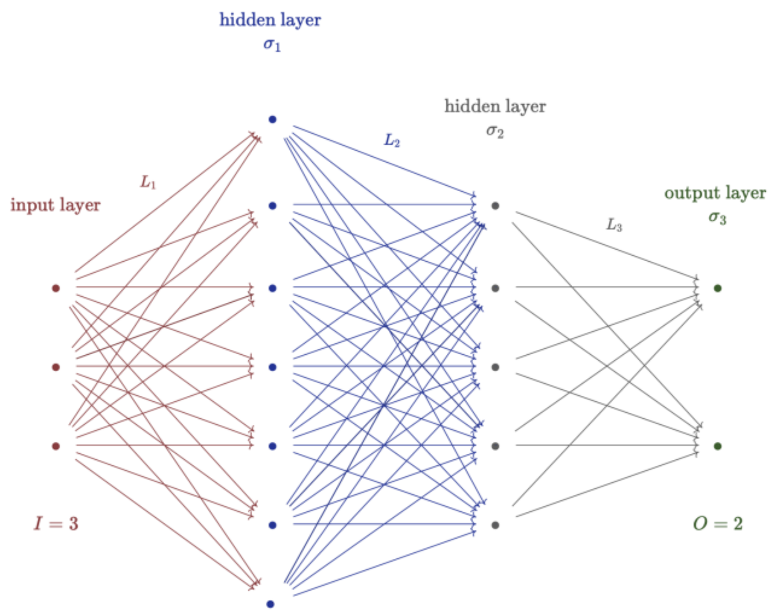


Figure 2.3: Diagram of a neural network with two hidden layers, three inputs and two outputs.

The hyperparameters of a neural network are $I, d_1, \ldots, d_{r-1}, O, \sigma_1, \ldots, \sigma_{r-1}$, and σ_r

Activation functions

When we constructed a neural network, the element σ_i in the FNN structure remained unclear. In this section, we will describe this functions and stand out its characteristics.

Back in the 80's, there was a renewed interest within the academia to investigate how a large number of neurons can work together to become "intelligent agents" [42]. This is called *Conexionism* and it is related to the research within neuroscience, that was now being associated with artificial intelligence (see Schwartz (1988) [36]). This research labor let to move along in the activation function's field.

The basic idea is to use a non-linear smooth activation functions. We will see later that the smooth characteristic enables to apply optimizers based on gradient methods while training.

Below, some of the most popular and used activation functions and its characteristics:

- Linear activation function
  - This is the *trivial* activation function. This function is commonly used only in the last layer, typically when considering a regression problem, and when we want the NN to be able to produce as an output any real value. We need to be careful when we use it, if we use this function in all the layers, we end up with a regression type function and all the point of a NN is to build a non-linear function.
  - $f(x) = x$
  - $f'(x) = 1$
  - $f \in \mathbb{R}$
- Rectified Linear Unit (ReLU)

- ReLU function is a popular activation function with a non-saturating property (the output range is not bounded). The highest advantage of this functions is its derivative that can be computed efficiently. This function is not differentiable in 0, but in practice we can define this single point as 1 and the problem is solved.
  - $f(x) = \max\{x, 0\}$
  - $f'(x) = \begin{cases} 0, & x < 0 \\ 1, & x > 0 \end{cases}$
  - $f \in [0, \infty)$

- Exponential Linear Unit (ELU)
  - ELU functions is an improvement of ReLU function. This function can achieve higher accuracy than ReLU in terms of its continuous differentiability (see Clevert et al. (2016)[16]).
  - $f(x) = \begin{cases} \alpha(e^x - 1), & x < 0 \\ x, & x \geq 0 \end{cases}$
  - $f'(x) = \begin{cases} \alpha, & x < 0 \\ 1, & x > 0 \end{cases}$
  - $f \in (-\alpha, \infty)$

- Leaky-ReLU (or PReLU)
  - Leaky-ReLU function, also called Parametric rectified linear unit (PReLU) when, instead of using 0.01, we use a parameter $\alpha \in \mathbb{R}^+$. In PReLU function, α is a learnable parameter.
  - $f(x) = \max\{0.01x, x\}$
  - $f'(x) = \begin{cases} 1, & x > 0 \\ 0.01, & otherwise \end{cases}$
  - $f \in \mathbb{R}$

- Softplus
  - The Softplus function is a smooth approximation of the ReLU function. The derivative is the Sigmoid function.
  - $f(x) = \log(1 + e^x)$
  - $f'(x) = \frac{1}{1+e^{-x}}$
  - $f \in (0, \infty)$

- Sigmoid function
  - A single neuron model with a Sigmoid activation function returns a logistic regression model. This function is infinitely differentiable and the range of the function is a real number between zero and one. This function is useful when the output of the NN model is bounded between these quantities.
  - $f(x) = \frac{1}{1+e^{-x}}$
  - $f'(x) = f(x)(1 - f(x))$
  - $f \in (0, 1)$

- Hyperbolic tangent function (tanh)
  - Together with Sigmoid, hyperbolic tangent use to be popular for hidden layers. The tanh function is also bounded and infinitely differentiable. This function is similar in shape to the Sigmoid but with a range in the oper interval (-1,1).
  - $f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
  - $f'(x) = 1 - f(x)^2$
  - $f \in (-1, 1)$

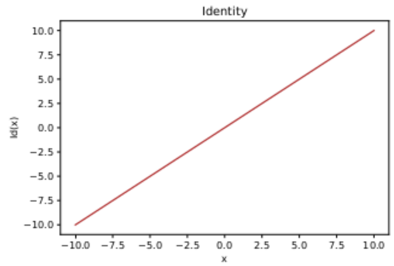
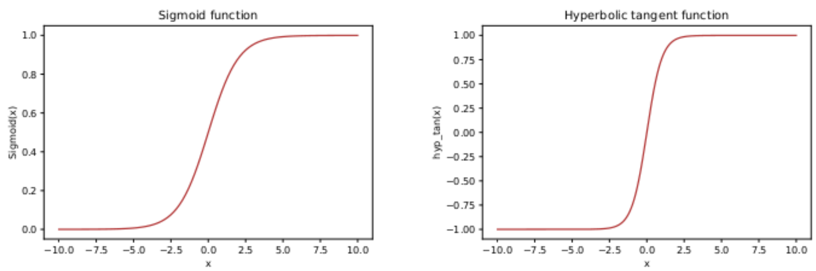


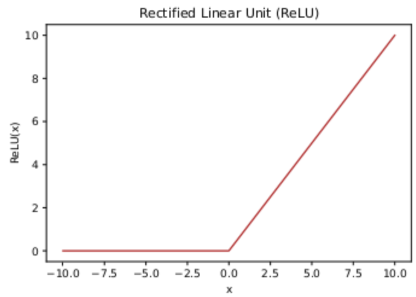
Figure 2.4: Linear activation function, indentity function $f(x) = x$.



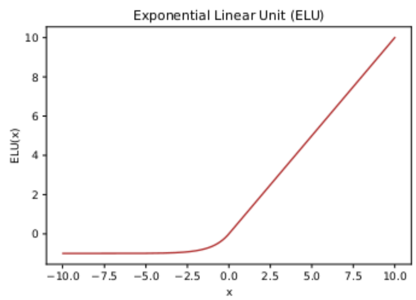
(a) $f(x) = \frac{1}{1+e^{-x}}$

(b) $f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$

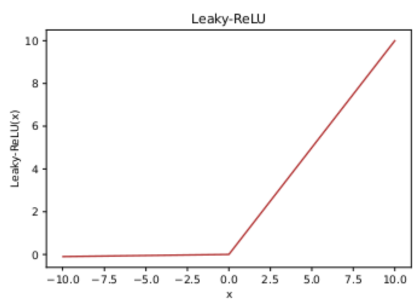
Figure 2.5: The most popular saturating activation functions. (a) Sigmoid function and (b) Hyperbolic tangent function.



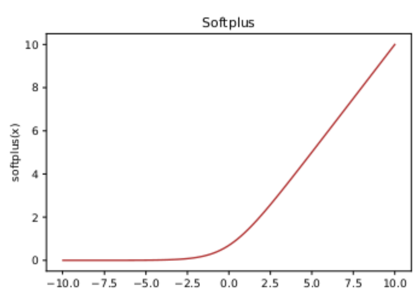
(a) $f(x) = \max\{x, 0\}$



(b) $f(x) = \begin{cases} \alpha(e^x - 1), & x < 0 \\ x, & x \geq 0 \end{cases}$



(c) $f(x) = \max\{0.01x, x\}$



(d) $f(x) = \log(1 + e^x)$

Figure 2.6: Activation functions with similar in shape. (a) ReLU, (b) ELU, (c) Leaky-ReLU or PReLU and (d) Softplus functions.

Neural Network optimization

The backpropagation algorithm is one of the reasons the Artificial Intelligence community regained interest in deep learning. In the previous section, we described the most popular activation functions. A notable characteristic of them is that they are smooth functions. This particularity enables a practical computation of gradients over a loss function in the parameters of a neural network. Backpropagation is an algorithm that efficiently computes these gradients.

We can start with knowing a algorithm that is suitable to train a feedforward neural network. However, we first focuss on the origin of this algorithm.

Suppose we have a dataset $\mathcal{D} = (x_i, y_i)_{i=1}^N,\ \ x_i, y_i \in \mathbb{R}$ and a function to optimize $f : \mathbb{R} \to \mathbb{R}$.

Algorithm: Gradient Descent		
Model function	f_θ	
Compute the loss function	$\mathcal{L}(\theta; \mathcal{D}) = \frac{1}{N} \sum_{x_i, y_i \in \mathcal{D}} l(y_i, f_\theta(x_i))$	
Inicialize parameterts	θ_0	
Iterate along all the number of iterations		
	Compute the gradient	$\nabla_\theta \mathcal{L}(\theta_t; \mathcal{D})$
	Update the parameters	$\theta_{t+1} = \theta_t - \nu \nabla_\theta \mathcal{L}(\theta_t; \mathcal{D})$.

Table 2.4: Gradient Descent algorithm explained.

Some arguments play against using this algorithm for training a neural network. The most important is that the nature of a NN involves a large dataset, making the algorithm very costly or even impossible to compute.

Suppose now, that we have a dataset $\mathcal{D} = (x_i, y_i)_{i=1}^N$, and m subsets called *minibatches* with $|\mathcal{D}_m| = M << N\ \ x_i, y_i \in \mathbb{R}$ and a function to optimize $f : \mathbb{R} \to \mathbb{R}$.

Algorithm: Stochastic Gradient Descent		
Model function	f_θ	
Inicialize parameterts	θ_0	
Iterate along all the number of iterations		
	Sample a minibatch	$\mathcal{D}_m$
	Compute the loss function	$\mathcal{L}(\theta; \mathcal{D}_m) = \frac{1}{M} \sum_{x_i, y_i \in \mathcal{D}_m} l(y_i, f_\theta(x_i))$
	Compute the gradient	$\nabla_\theta \mathcal{L}(\theta_t; \mathcal{D}_m)$
	Update the parameters	$\theta_{t+1} = \theta_t - \nu \nabla_\theta \mathcal{L}(\theta_t; \mathcal{D}_m)$.

Table 2.5: Stochastic Gradient Descent algorithm explained.

Backpropagation has its origins in the (backward-mode) algorithmic differentiation, which computes the gradient of empirical risk for a feedforward neural network. In general, all algorithmic differentiation is based on the chain rule. A detail explanation of this technique is in Pakkanen (2020) [33, Section 3.3].

Loss function

The loss function is a function $l : \mathbb{R}^O \times \mathbb{R}^O \to \mathbb{R}$ that given $\bar{x} \in \mathbb{R}^I$, and $\bar{y} \in \mathbb{R}^O$, computes $l(f(\bar{x}), \bar{y})$.

In the present work, we use the squared loss: $l(\hat{y}, y) = (\hat{y} - y)^2$, with $\hat{y}, y \in \mathbb{R}$.

Chapter 3

CVA analysis under Merton's model

In this section, we will explain in depth how to price unilateral CVA in a Merton's framework. We will contrast the CVA under the independent framework, against the correlated computation with Wrong Way Risk (WWR) in the Merton Model. We present a mathematical expression that can be programmed through a numerical integration process. Under this procedure, in the next chapter we will present the data base that will be the input for training a neural network. Finally, we conclude this chapter with an analysis of the boundaries of the CVA.

3.1 The Wrong Way Risk with Merton's model

We start with the definition 2.1.1 from the previous chapter. We know that the unilateral CVA is

$$\mathrm{UCVA}_t = \mathbb{E}\left[\mathtt{LGD}\,\mathbb{1}_{\{t<\tau_C\leq T\}}D(t,\tau_C)(\mathtt{NPV}(\tau_C))^+|\mathcal{F}_t\right]$$

For simplicity we will assume the valuation is made at time $t = 0$ and we will use the contracted notation for the expected value given some information as follows

$$\mathrm{UCVA}_0 = \mathbb{E}_0\left[\mathtt{LGD}\,\mathbb{1}_{\{0<\tau_C\leq T\}}D(0,\tau_C)(\mathtt{NPV}(\tau_C))^+\right] \tag{3.1.1}$$

In this study case, we assume that the derivative portfolio for which we are computing the CVA contains a long position in a Call option under the stock S. We assume the dynamics of the stock process follows the Geometric Brownian Motion (GBM) under the risk neutral measure

$$dS_t = r^s S_t dt + \sigma_s S_t dW_t^s$$

For Proposition 2.2.1, we know that the value process in a structured framework follows a GBM as well. The stochastic differential equation for the firm value process under the risk neutral measure is

$$dV_t = (r_v - k)V_t dt + \sigma_v V_t dW_t^v$$

To compute the *Wrong Way Risk* (WWR), we need to define a way to correlate the market risk of the NPV of the portfolio, and the credit risk involved in the transaction. In this model the correlation is incorporated in the random bits as

$$\rho\, dt = d\langle W_t^s, W_t^v\rangle,$$

where $d\langle\cdot\rangle$ is the quadratic variation of the two Brownian motions in the dynamics of the stock and the firm value.

If we assume independence of default from the stock, the CVA factors in equation (3.1.1) would become

$$\mathrm{UCVA}_0^{ind} = \mathtt{LGD}\,\mathbb{E}_0\left[\mathbb{1}_{\{0<\tau_C\leq T\}}\right]\ \mathbb{E}_0\left[D(0,\tau_C)(\mathtt{NPV}(\tau_C))^+\right] \tag{3.1.2}$$

which means that the CVA is just the product of the LGD rate, the probability of an indicator function over a set (which becomes the probability of the set), and the present value of the positive part of the netting portfolio. In a portfolio with an option, the equation (3.1.2) can be reduced to:

$$\text{UCVA}_0^{ind} = \texttt{LGD} \cdot \mathbb{Q}\{\tau_C \leq T\} \cdot \text{Call price}(0). \tag{3.1.3}$$

This computation is very easy and super fast. Each one of the terms in (3.1.3) is feasible and with a straight computation, the UCVA can be achieved effectively and expeditiously. However, we should be cautious with the error of computation when assuming independence in the stock process and the default risk.

From section 2.2.1, we know that Merton's model checks the default only at maturity. The NPV of a portfolio with a call option under S stock, with strike K, at maturity, is its payoff $(S_T - K)^+$. Hence, the UCVA under Merton's framework can be written as

$$\text{UCVA}_0 = \mathbb{E}_0[(1 - \texttt{Rec})\ \mathbb{1}_{\{V_T < L\}}(D(0,T)(S_T - K)^+)] \tag{3.1.4}$$

Some remarks for the previous equation:

- We are doing pricing, so we use the risk neutral expectation (expectation under the risk-neutral measure $\mathbb{Q}$).
- Recall that $(1 - \texttt{Rec}) = \texttt{LGD}$.
- Note that the discount factor $D(0,T)$ could be stochastic or deterministic. For simplicity we consider it deterministic.
- The NPV at default time is $(\texttt{NPV}(T))^+ = ((S_T - K)^+)^+$ but there is no necessity to write the positive part twice. Then $(\texttt{NPV}(T))^+ = (S_T - K)^+$.

To solve the equation (3.1.4) we start with the analytical solution for the stock and value processes (see Appendix A.3 for the detail in the computation to solve a GBM) at maturity time,

$$\begin{aligned} V(T) &= V_0 \exp\left\{\left(\mu_v - \tfrac{\sigma_v^2}{2}\right)T + \sigma_v\sqrt{T}N_1(0,1)\right\} \\ S(T) &= S_0 \exp\left\{\left(\mu_s - \tfrac{\sigma_s^2}{2}\right)T + \sigma_s\sqrt{T}N_2(0,1)\right\} \end{aligned} \tag{3.1.5}$$

where μ_s and μ_v are the drifts for the stock and the value processes respectively, σ_s and σ_v their respective volatilities, $V_0 = V(0)$ the initial firm value, $S_0 = S(0)$ is the initial stock price, and $N_i(0,1)$ with $i = 1, 2$ are normal standard random variables.

We can obtain the expression of the UCVA in terms of two normal standard distributions that are correlated. Without loss of generality, we assume $\texttt{Rec} = 0$.

Thus, the UCVA_0 is equivalent to compute

$$\mathbb{E}_0\left[\mathbb{1}_{\left\{V_0 \exp\left\{\left(\mu_v - \frac{\sigma_v^2}{2}\right)T + \sigma_v\sqrt{T}N_1(0,1)\right\} < L\right\}} D(0,T)\left(S_0 \exp\left\{\left(\mu_s - \frac{\sigma_s^2}{2}\right)T + \sigma_s\sqrt{T}N_2(0,1)\right\} - K\right)^+\right]$$

The previous expression is equivalent to compute

$$D(0,T)\ \mathbb{E}_0\left[\mathbb{1}_{\left\{N_1(0,1) < \frac{\log\left(\frac{L}{V_0}\right) - \left(\mu_v - \frac{\sigma_v^2}{2}\right)T}{\sigma_v\sqrt{T}}\right\}}\left(S_0 \exp\left\{\left(\mu_s - \frac{\sigma_s^2}{2}\right)T + \sigma_s\sqrt{T}N_2(0,1)\right\} - K\right)^+\right] \tag{3.1.6}$$

and we observe that

$$S_0 \exp\left\{\left(\mu_s - \frac{\sigma_s^2}{2}\right)T + \sigma_s\sqrt{T}N_2(0,1)\right\} - K \geq 0$$

$$\Leftrightarrow$$

$$N_2(0,1) \geq \frac{\log\left(\frac{K}{S_0}\right) - \left(\mu_s - \frac{\sigma_s^2}{2}\right)T}{\sigma_s\sqrt{T}}$$

Name x and y the variables that represents the values of the normal variables $N_1 = x$ and $N_2 = y$, then

$$\int_{\frac{\log\left(\frac{K}{S_0}\right)-\left(\mu_s-\frac{\sigma_s^2}{2}\right)T}{\sigma_s\sqrt{T}}}^{+\infty} \int_{-\infty}^{\frac{\log\left(\frac{L}{V_0}\right)-\left(\mu_v-\frac{\sigma_v^2}{2}\right)T}{\sigma_v\sqrt{T}}} \left(S_0 \exp\left\{\left(\mu_s - \frac{\sigma_s^2}{2}\right)T + \sigma_s\sqrt{T}y\right\} - K\right) p_{N_1,N_2}(x,y)dxdy \tag{3.1.7}$$

where $p_{N_1,N_2}(x, y)$ is the probability density function of a multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$, with $\mu = \left[\begin{smallmatrix}0\\0\end{smallmatrix}\right]$, and $\Sigma = \left[\begin{smallmatrix}1 & \rho\\ \rho & 1\end{smallmatrix}\right]$.

3.2 Numerical Integration Convergence

In the expression (3.1.7), there is a feasible formula to accurately compute the unilateral CVA under Merton's model, incorporating the wrong way risk correlation. The expression is a double integral of a deterministic quantity times a PDF of a bivariate normal. Hence, we needed to find a suitable software that allows to program methods or use pre-builded functions that could build numerical approximations of integrals over an infinite space.

For this task it was used `scipy.integrate.dblquad`[1], a *Python*[2] method that returns the double definite integral of a given function. To compute a numerical integral over an infinite space, it is precise to cut off the extremes of the double integral in a "reasonable" level. But, what is reasonable in this case? We used a analytical procedure that allows to identify up to what point and onwards, the value of the integral changes so little that can be negligible to the purpose of the task.

The tool used to compute the PDF of a normal is the method

$$\texttt{bi\_stand\_norm} = \texttt{multivariate\_normal}(\texttt{mean} = [0,0], \texttt{cov} = [[1,\rho],[\rho,1]])$$

from the library `scipy.stats`[3], together with its attribute $\texttt{bi\_stand\_norm.pdf}([\cdot,\cdot])$, that indicates the probability function of the object.

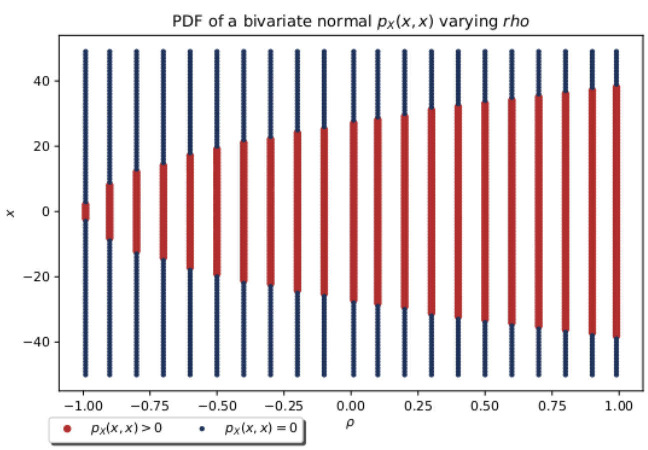


Figure 3.1: Convergence speed to zero, of a probability density function for a bivariate standard normal distribution $f(x, x)$ with Scipy.

[1]https://docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.dblquad.html
[2]https://www.python.org/
[3]https://docs.scipy.org/doc/scipy/reference/stats.html

The figure 3.1 is a graphical resource that shows the convergence speed to zero of a probability distribution, corresponding to a bivariate standard normal within different levels or correlation. We observe that the higher is the correlation, the slower is the convergence. Thus, with probability one, choosing 40 and -40 as boundaries for the numerical integration, the integrals converges to the true CVA, and this approximation is the most accurate value that we can obtain within a numerical approximation.

3.3 A CVA boundary

In the CVA, the component related to the firm value process ($\mathbb{1}_{\{V_T<L\}}$) is decreasing in V. That means that each time $V(T)$ is bigger than L, the indicator function $\mathbb{1}_{\{V_T<L\}}$ is zero. Additionally, the payoff $(S_T-K)^+$ in the CVA is increasing in S, which means that for a fixed strike K, if S_T increases, the function $(S_T-K)^+$ also increases.

A correlation equal to one between the stock process and the value process means that both processes move in the same direction. Conversely, a correlation equal to minus one means that an increment in the process S comes with a decrement in the process V.

An upper boundary for the CVA is achieved in the scenario that $\mathbb{1}_{\{V_T<L\}}$ is large, and $(S_t-K)^+$ is large. The first is large when V is deceasing, the second is large when S is increasing. Thus, the CVA is large when V and S are negatively correlated.

Suppose that the interest rates are deterministic functions and the recovery rate is $\texttt{Rec}=0$.

Define the $\text{CVA}^{\text{boundary}}$ the scenario with $\rho=-1$

$$\mathbb{E}_0\left[\mathbb{1}_{\{V_T<L\}}(D(0,T)(S_T-K)^+)\right] \leq \mathbb{E}_0\left[\mathbb{1}_{\{V_T<L\}}\left(D(0,T)(S_T-K)^+\right) \mid \rho=-1\right] = \text{CVA}^{\text{boundary}}$$

So, when $N_1, N_2 \sim \mathcal{N}(0,1)$ are two normal random variables negatively correlated such that $N_1=-N_2$, we have

$$\begin{aligned}
\text{CVA}^{\text{boundary}} &= D(0,T)\,\mathbb{E}_0\left[\mathbb{1}_{\{N_1<\mathcal{A}\}}\left(S_0\exp\left\{\left(\mu_s-\frac{\sigma_s^2}{2}\right)T-\sigma_s\sqrt{T}N_1\right\}-K\right)^+\right] && (3.3.1)\\
&= D(0,T)\int_{-\infty}^{\mathcal{A}}\int_{-\infty}^{\mathcal{B}}\left(S_0\exp\left\{\left(\mu_s-\frac{\sigma_s^2}{2}\right)T-\sigma_s\sqrt{T}x\right\}-K\right)p_{N_1}(x)dx\\
&= D(0,T)\int_{-\infty}^{\min(\mathcal{A},\mathcal{B})}\left(S_0\exp\left\{\left(\mu_s-\frac{\sigma_s^2}{2}\right)T-\sigma_s\sqrt{T}x\right\}-K\right)p_{N_1}(x)dx && (3.3.2)
\end{aligned}$$

where $p_{N_1}(x)$ is the PDF of N_1 random variable,

$$\mathcal{A} := \frac{\log\left(\frac{L}{V_0}\right)-\left(\mu_v-\frac{\sigma_v^2}{2}\right)T}{\sigma_v\sqrt{T}}$$

and

$$\mathcal{B} := \frac{\log\left(\frac{K}{S_0}\right)+\left(\mu_s-\frac{\sigma_s^2}{2}\right)T}{\sigma_s\sqrt{T}}.$$

The proof of $\mathcal{A}$ and $\mathcal{B}$ in the following lines:

$$\{V_T<L\} \Leftrightarrow \left\{N_1 < \frac{\log\left(\frac{L}{V_0}\right)-\left(\mu_v-\frac{\sigma_v^2}{2}\right)T}{\sigma_v\sqrt{T}} =: \mathcal{A}\right\} \qquad (3.3.3)$$

$$\left(S_0 \exp\left\{\left(\mu_s - \frac{\sigma_s^2}{2}\right)T - \sigma_s\sqrt{T}N_1\right\} - K\right)^+$$

$$\Leftrightarrow$$

$$S_0 \exp\left\{\left(\mu_s - \frac{\sigma_s^2}{2}\right)T - \sigma_s\sqrt{T}N_1\right\} - K \geq 0$$

$$\Leftrightarrow$$

$$\exp\left\{\left(\mu_s - \frac{\sigma_s^2}{2}\right)T - \sigma_s\sqrt{T}N_1\right\} \geq \frac{K}{S_0}$$

$$\Leftrightarrow$$

$$\left(\mu_s - \frac{\sigma_s^2}{2}\right)T - \sigma_s\sqrt{T}N_1 \geq \log\left(\frac{K}{S_0}\right)$$

$$\Leftrightarrow$$

$$\sigma_s\sqrt{T}N_1 \leq \log\left(\frac{K}{S_0}\right) + \left(\mu_s - \frac{\sigma_s^2}{2}\right)T$$

$$\Leftrightarrow$$

$$N_1 \leq \frac{\log\left(\frac{K}{S_0}\right) + \left(\mu_s - \frac{\sigma_s^2}{2}\right)T}{\sigma_s\sqrt{T}} := \mathcal{B}$$

Then, the CVA boundary when the correlation is minus one, is the equation (3.3.2).

Chapter 4

The learning process in the CVA

In this chapter, we present the methodology of training a neural network that learns from numerous CVA calculations. The first section is intended to show the parameters that the neural network will use to learn the impact of the correlation ρ in the model. We will show the structure of the input data and the CVA that will be used to train a FNN. The following section is aimed to characterize a few different neural networks and evaluate his effectiveness to compute CVA.

4.1 Building a Neural Network that learns CVA with correlation

Recall the formula for the UCVA

$$\text{CVA}_0 = \mathbb{E}_0\left[(1-\texttt{Rec})\mathbb{1}_{\{V(T)<L\}}(D(0,T)(\texttt{NPV}(T))^+)\right] \tag{4.1.1}$$

The methodology follows the following assumptions:

- The recovery rate is zero ($\texttt{Rec} = 0$).
- The interest rates are deterministic $D(0,T) = e^{-rT}$, therefore, the discount factor can be taken out of the expected value.
- The portfolio contains a Call option under a stock S, maturity $T = 1$ year, strike $K = 100$, volatility σ_s. The stock follows a Geometric Brownian Motion (GBM) dynamics, with initial value $S(0) = S_0$, drift μ_s under the phisical measure $\mathbb{P}$, and drift $r = 0.01$ (interest rate of 1%) under the risk neutral measure $\mathbb{Q}$.
  - The dynamics for the stock value

    $$dS_t = rS_t dt + \sigma_s S_t dW_t^s$$
- (This assumption is for Merton's model) The firm value process of the counterparty follows a GBM with initial value $V(0) = V_0 > 0$.
  - The dynamics for the value process

    $$dV_t = (r-k)V_t dt + \sigma_v V_t dW_t^v$$

    for the drift, the interest rate $r = 0.01$ and the payout ratio $k = 0$.
- Given the debt level L, we consider the the debt ratio as $\frac{L}{V_0}$.

Thus, the unilateral CVA becomes a function depending on five variables

$$\text{CVA}_0\left(\frac{L}{V_0}, S_0, \sigma_s, \sigma_v, \rho\right) = e^{-0.01}\,\mathbb{E}_0\left[\mathbb{1}_{\left\{\frac{V(T)}{V_0}<\frac{L}{V_0}\right\}}(S_T - 100)^+\right] \tag{4.1.2}$$

where

$$V(T) = V_0 \exp\left\{\left(0.01 - \frac{\sigma_v^2}{2}\right) + \sigma_v N_1\right\}$$
$$S(T) = S_0 \exp\left\{\left(0.01 - \frac{\sigma_s^2}{2}\right) + \sigma_s N_2\right\}$$

and $[N_1, N_2]$ conforms a multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$, with $\mu = \left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right]$, and $\Sigma = \left[\begin{smallmatrix} 1 & \rho \\ \rho & 1 \end{smallmatrix}\right]$.

4.1.1 The structure of the input data and the labels

The input data we use in our problem is a matrix of dimension $N \times 5$, the features matrix:

$$\begin{pmatrix} \frac{L}{V_0}^1 & {S_0}^1 & {\sigma_v}^1 & {\sigma_s}^1 & \rho^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{L}{V_0}^N & {S_0}^N & {\sigma_v}^N & {\sigma_s}^N & \rho^N \end{pmatrix}$$

The elements that correspond to the values the neural network estimates, the labels matrix:

$$\begin{pmatrix} \mathrm{CVA}_0\left(\frac{L}{V_0}^1, {S_0}^1, {\sigma_v}^1, {\sigma_s}^1\right) \\ \vdots \\ \vdots \\ \mathrm{CVA}_0\left(\frac{L}{V_0}^N, {S_0}^N, {\sigma_v}^N, {\sigma_s}^N, \rho^N\right) \end{pmatrix}$$

.

We choose that the variables take values from the following sets

$$\frac{L}{V_0} \in \{0.01\} \cup \{0.1, 0.2, \ldots, 0.9\}$$
$$S_0 \in \{82, 84, \ldots, 120\}$$
$$\sigma_v \in \{0.05, 0.1, \ldots, 1\}$$
$$\sigma_s \in \{0.05, 0.1, \ldots, 1\}$$
$$\rho \in \{-0.99\} \cup \{-0.9, -0.8, \ldots, 0.9\}$$

For example, the i-th sample could be $\frac{L}{V_0}^i = 0.01, {S_0}^i = 92, {\sigma_v}^i = 0.45, {\sigma_s}^i = 0.1, \rho^i = -0.7$.

The total number of samples is $1,600,000$. For training, we use $N = 1,600,000$ samples. For validation, we computed further $96,000$ CVA's with parameters not included in the training parameters.

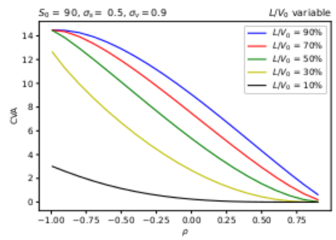
Using equation (4.1.2) (or equation (3.1.7) with the above assumptions), we computed in *Python* $N + N^{val}$ different scenarios of the function $\mathrm{CVA}_0\left(\frac{L}{V_0}, S_0, \sigma_s, \sigma_v, \rho\right)$

4.1.2 Computation time of numerical integration

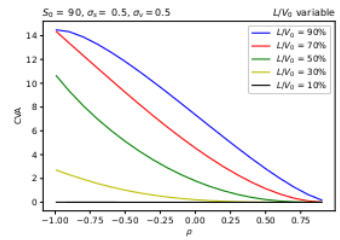
The features and labels for training are quantities approximated with a numerical integration. Regarding the computational time to compute the validation set, on average, each numerical integration takes 0.30 seconds to run. The sample size for the validation set is 96,000 CVA's, it took 28,944 seconds to finish all the calculations (just above the 8 hours).

4.1.3 Influence of parameters in CVA

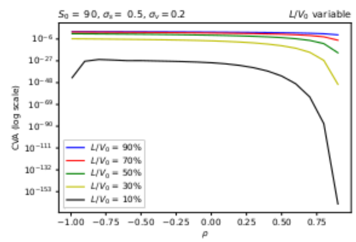
It is in our interest to understand the influence of each parameter in the CVA. The following plots describe the behaviour of the CVA against the parameters of the model.



(a) $\sigma_v = 0.9$



(b) $\sigma_v = 0.5$



(c) $\sigma_v = 0.2$

Figure 4.1: The CVA curves vs correlation of stock and firm value, for five different levels of debt, in three volatility scenarios: (a) high volatility in the firm value process, (b) medium volatility in the firm value process, and (c) low volatility in the firm value process.

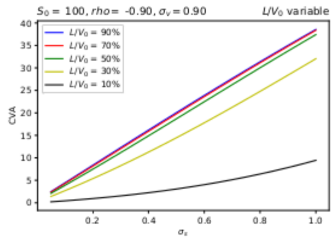
The Figure 4.1 illustrates the function CVA(ρ) with $\frac{L}{V_0}, S_0, \sigma_s$, and σ_v constants. This function is decreasing with the correlation ρ. This behaviour confirms the discussion in section 3.3 where we found that an upper boundary of the CVA is the CVA itself at correlation level $\rho = -1$.

The interpretation of the negative correlation: If V decreases it means that it it is more likely the default. If the stock process S increases, that means the option is more in-the-money (the option is more likely to be exercised). If the default is more likely and the option is more in the money, the CVA is worth more. The interpretation of a positive correlation: If V decreases the default is more likely. If S decreases, the option becomes more out of the money, then the CVA worth less.

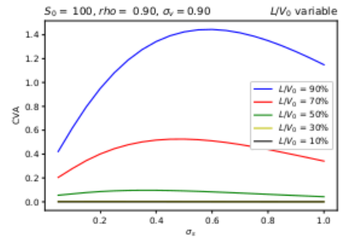
CVA(ρ) is increasing with the debt ratio $\frac{L}{V_0}$. When the debt ratio is high, that means the debt is almost the firm value, so it is more likely the default, the CVA is worth more. When the debt ratio is low, is more likely that the company payback the debt and default does not occur, so the CVA is worth less.

The Figure 4.2 presents the CVA as a function of the volatility for the stock price, for different levels of the debt ratio $\frac{L}{V_0}$. It is appreciated an increasing CVA with respect to σ_s, and also increasing with respect to $\frac{L}{V_0}$. The volatility of the stock is affects the CVA price. The highest level of CVA can be achieved with high levels of volatility in the stock and a negative correlation (as in Subfigure 4.2(a)). On the other hand, the lowest levels of CVA are in the highest correlation. This plot shows a concave shape. Another observation is that the more correlation in the model, the less linear the CVA is.

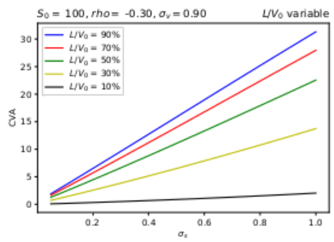
The independent CVA is the model when it is assumed zero correlation between the stock dynamics and the firm value dynamics. In Subfigure 4.3(a) it is shown a scenario when the debt ratio is 10% and the initial stock price is at the money. The surface in the Subfigure 4.3(b)



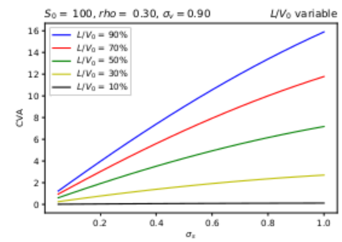
(a) $\rho = -0.9$



(b) $\rho = 0.9$

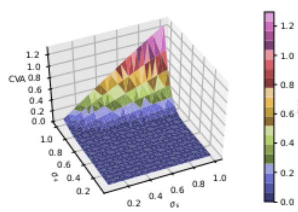


(c) $\rho = -0.3$

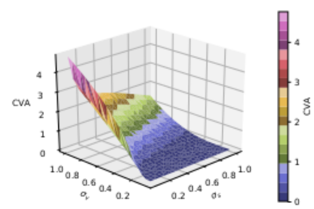


(d) $\rho = 0.3$

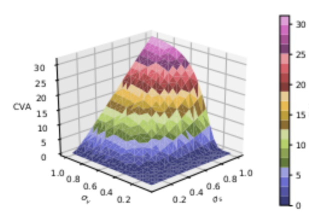
Figure 4.2: The CVA curve vs the volatility of the underlying (σ_s), for five levels of debt, in four correlation scenarios: (a) high negative correlation (WWR), (b) high positive correlation (RWR), (c) low negative correlation (WWR), and (d) low positive correlation (RWR).



(a) $\rho = 0, \frac{L}{V_0} = 0.10, S_0 = 100$



(b) $\rho = 0.90, \frac{L}{V_0} = 0.80, S_0 = 110$

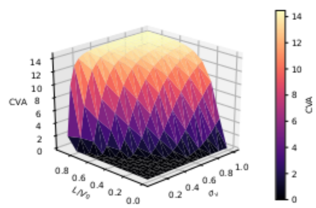


(c) $\rho = -0.90, \frac{L}{V_0} = 0.50, S_0 = 90$

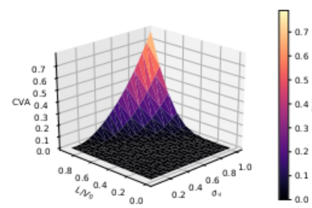
Figure 4.3: CVA and volatility surfaces. (a) Independent CVA, low debt ratio, stock at the money. (b) RWR CVA, high debt ratio, stock in the money. (c) WWR CVA, medium debt ratio, stock out of the money.

represents a RWR correlation in the CVA, with a $\rho = 0.90$, the debt ratio is high reaching 80% and the initial stock price is 10 units above the strike price in the option. The Subfigure 4.3(c) shows a WWR correlation with $\rho = -0.90$, the debt ratio is 50% and the initial stock price is below the strike. It can be seen that the surface reaches quickly high CVA levels.

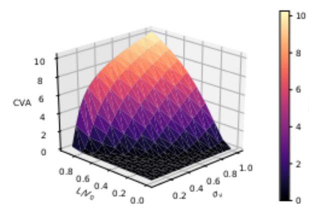
Figures 4.4 and 4.5 show the surfaces of CVA against the parameters of the firm value process and the stock process respectively.



(a) $\rho = -0.90, \sigma_s = 0.50, S_0 = 90$

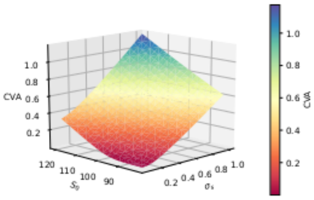


(b) $\rho = 0.90, \sigma_s = 0.50, S_0 = 90$

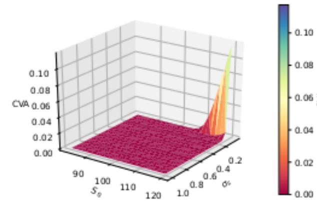


(c) $\rho = -0.10, \sigma_s = 0.50, S_0 = 90$

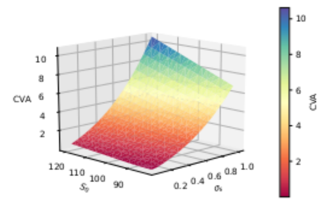
Figure 4.4: The Firm Value parameters and CVA surface. The stock volatility is 50%, the initial value of the stock is 90 and: (a) high WWR correlation. (b) RWR correlation. (c) low WWR correlation.



(a) $\rho = -0.10, \sigma_v = 0.50, \frac{L}{V_0} = 0.30$



(b) $\rho = 0.90, \sigma_v = 0.50, \frac{L}{V_0} = 0.30$



(c) $\rho = -0.90, \sigma_v = 0.50, \frac{L}{V_0} = 0.30$

Figure 4.5: The Stock parameters and CVA surface. The firm value volatility is 50%, the debt rartio is 30% and: (a) low WWR correlation. (b) RWR correlation. (c) high WWR correlation.

4.1.4 Design and Calibration of the Neural Networks

The hyperparameters of the FNN play an important role in the calibration and performance of the neural network. There are many attempts to find a method that gives the optimal set of hyperparameters. From genetic algorithms (for instance, [3]), to Bayesian approaches (as in [43]). However, there is still not a standard method, most of them are based on the knowledge of the problem.

In this section, we attempt to experiment with several architectures of the neural network, aiming to find the most suitable structure with the smallest error. We implemented all the procedures and *python*[1].

The performance of a FNN can variate taking into account the following features:

- Number of layers
  - Use 2 or more hidden layers
- Number of units in the layers
  - Small amount of units in each hidden layer
  - Big amount of units in each hidden layer
- Type of activation function
  - ReLU
  - ELU
  - Leacky-ReLU
- Number of iterations

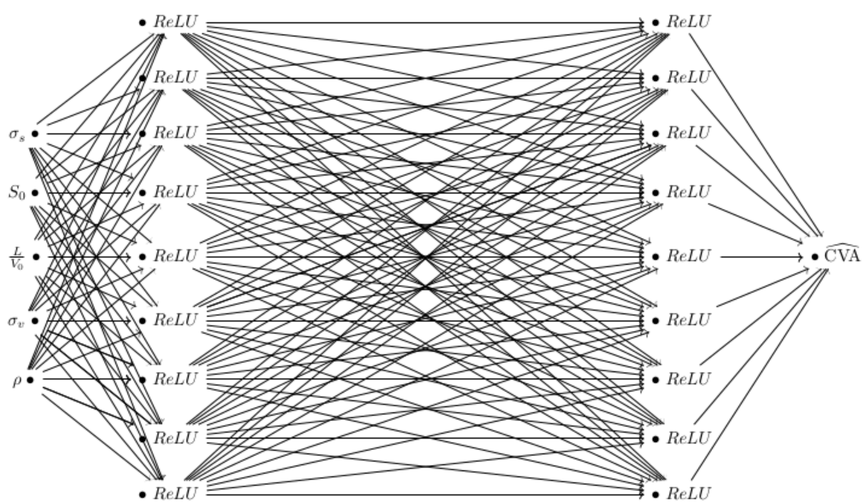
[1]The code is available at `https://github.com/MarianneTM/Deep_CVA`

– Varying the number of batches and epochs

We call "Design", to a formulation of a feedforward neural network with some set of parameters and some optimization characteristics.

The following diagram illustrates the structure of a FNN design:

$$\widehat{\text{CVA}} \in \mathcal{N}_3(5, 9, 9, 1; ReLU, ReLU, Id)$$



In the remaining of this section, several designs are presented together with its characteristics and the performance while learning.

Design 1: Three hidden layers with 200 neurons each, using ELU as activation function

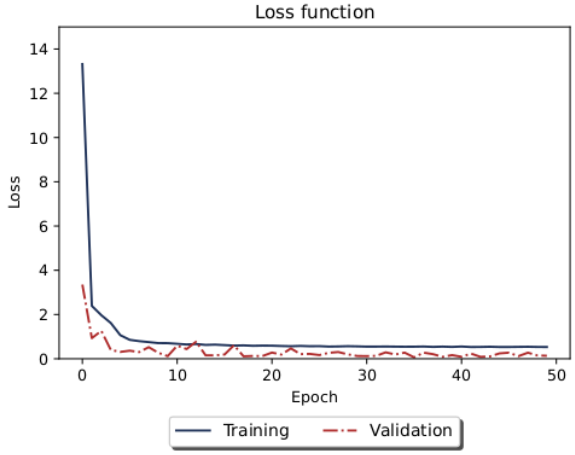


Figure 4.6: The loss function from the training of FNN with Design 1.

The loss functions increases from 13 to almost 0.5, for the training data set. The learning performance in the validation dataset is more erratic, but stays below the loss of the training dataset.

Characteristics of Design 1	
Structure of the NN	
Architecture	$\widehat{\text{CVA}} \in \mathcal{N}_4(5, 200, 200, 200, 1; ELU, ELU, ELU, Id)$
Parameters	81,801
The datasets	
Training	1,600,000 samples
Validation	96,000 samples
Optimizer and loss function	
Optimizer	Adam
Loss function	mean square error
Iterations	
Batch size	1000
Number of epochs	50
Performance of datasets (loss of the last epoch)	
Training	0.53
Validation	0.13

Table 4.1: The characteristics of the Design 1.

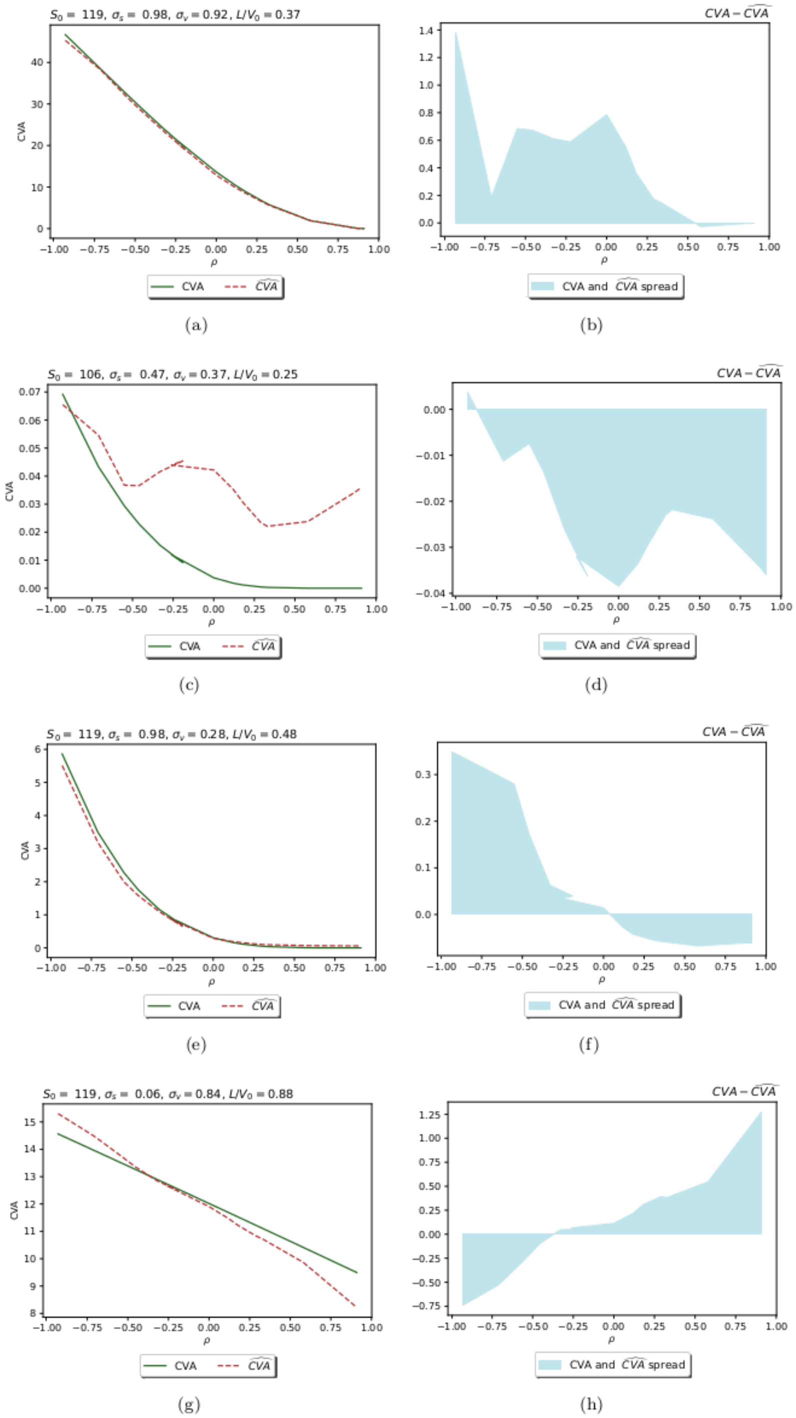


Figure 4.7: Comparative plots of the CVA obtained by numerical integration vs the deep CVA obtained by the neural network under the Design 1.

Design 2: Two hidden layers with 9 neurons each, using ReLU as activation function

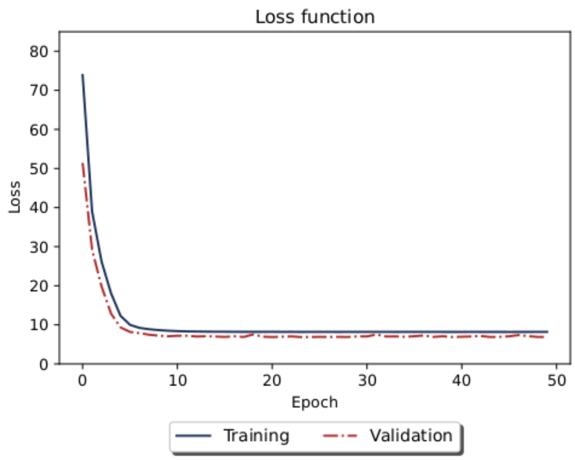
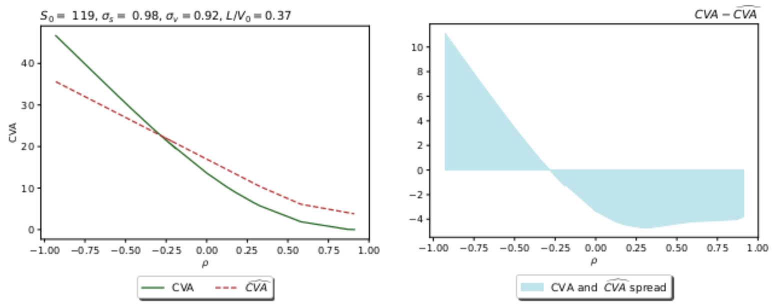


Figure 4.8: The loss function from the training of FNN with Design 2.

The loss function increases from 75 to almost 8.2 for the training data set. An error of 8.2 is significantly large.

Characteristics of Design 2	
Structure of the NN	
Architecture	$\widehat{\text{CVA}} \in \mathcal{N}_4(5, 9, 9, 1; ReLU, ReLU, Id)$
Parameters	154
The datasets	
Training	1,600,000 samples
Validation	96,000 samples
Optimizer and loss function	
Optimizer	Adam
Loss function	mean square error
Iterations	
Batch size	1000
Number of epochs	50
Performance of datasets (loss in the last epoch)	
Training	8.19
Validation	6.89

Table 4.2: The characteristics of the Design 2.



(a) (b)

Figure 4.9: Comparative plots of the CVA obtained by numerical integration vs the CVA obtained by the neural network under the Design 2.

As Figure 4.1 shows, the FNN with Design 2 behaves poorly. For levels of ρ near -1.0 the neural network underestimates the true CVA (in some cases more that 10). On the other hand, levels near $\rho = 1.0$ the estimation is above the true CVA. In Subfigure 4.9(b) it is clear that the difference between the true CVA with the estimated CVA is considerable.

Design 3: Two hidden layers with 50 neurons each, using ReLU as activation function

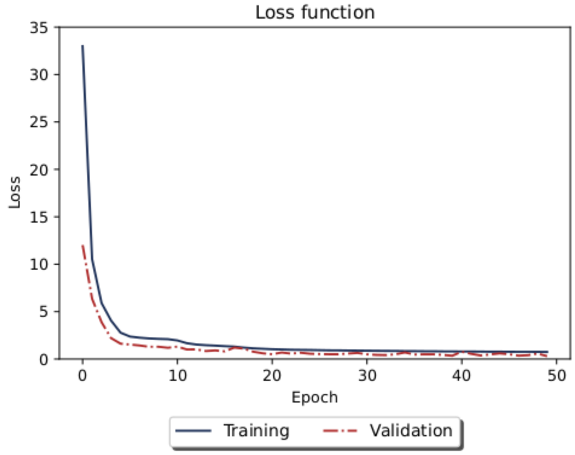


Figure 4.10: The loss function from the training of FNN with Design 3.

The Figure 4.10 shows that the loss function converges to 0.7. The validation dataset performed better than the training dataset.

Characteristics of Design 3	
Structure of the NN	
Architecture	$\widehat{\mathrm{CVA}} \in \mathcal{N}_3(5, 50, 50, 1; ReLU, ReLU, Id)$
Parameters	2,901
The datasets	
Training	1,600,000 samples
Validation	96,000 samples
Optimizer and loss function	
Optimizer	Adam
Loss function	mean square error
Iterations	
Batch size	500
Number of epochs	50
Performance of datasets (loss in the last epoch)	
Training	0.74
Validation	0.30

Table 4.3: The characteristics of the Design 3.

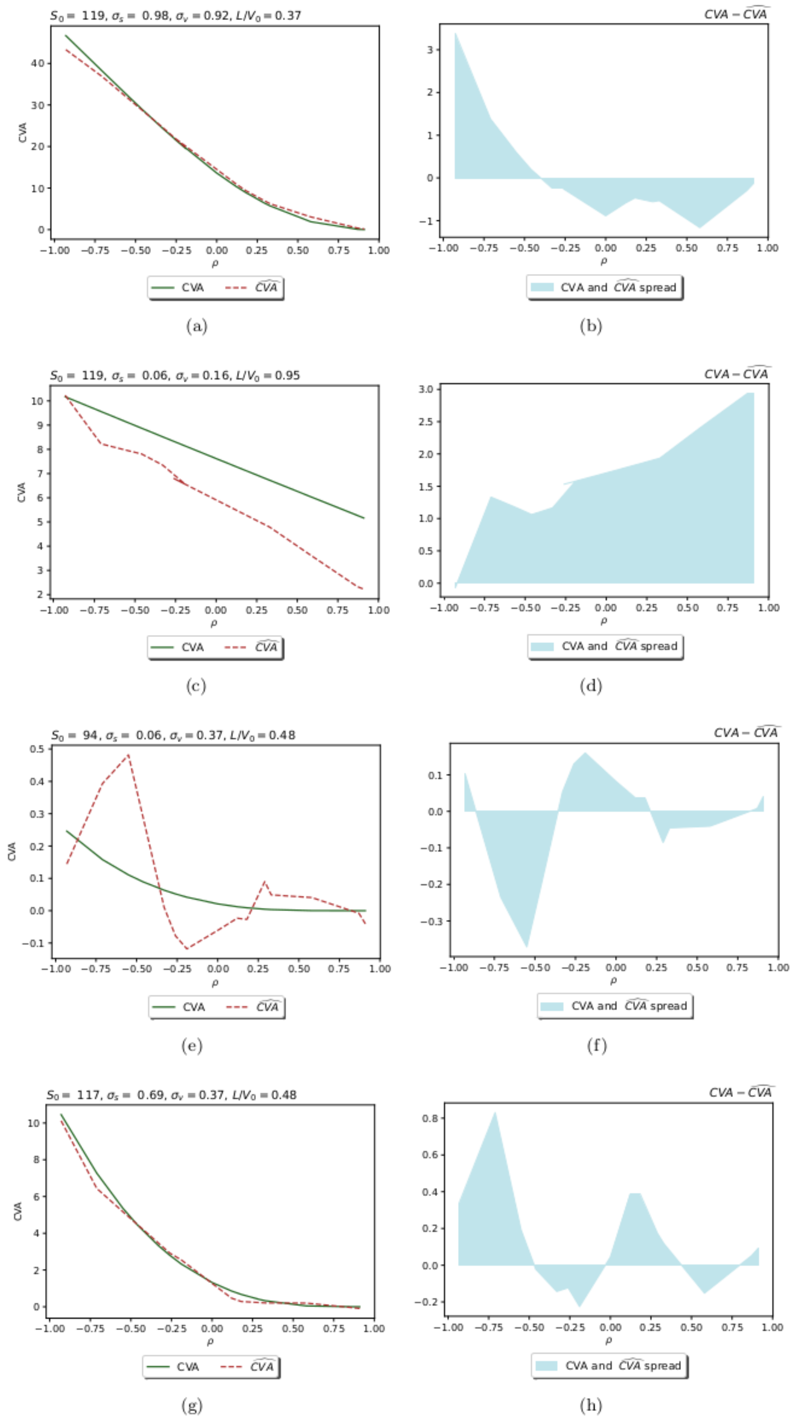


Figure 4.11: Comparative plots of the CVA obtained by numerical integration vs the deep CVA obtained by the neural network under the Design 3.

Design 4: Two hidden layers with 50 neurons each, using Leaky-ReLU as activation function

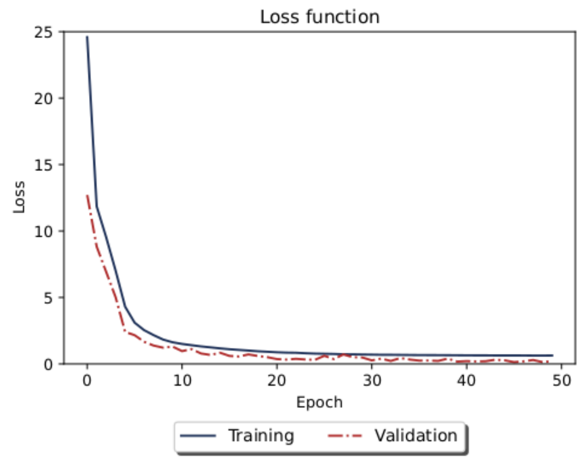


Figure 4.12: The loss function from the training of FNN with Design 4.

Characteristics of Design 4	
Structure of the NN	
Architecture	$\widehat{\text{CVA}} \in \mathcal{N}_3(5, 50, 50, 1; Leaky - ReLU, Leaky - ReLU, Id)$
Parameters	2,901
The datasets	
Training	1,600,000 samples
Validation	96,000 samples
Optimizer and loss function	
Optimizer	Adam
Loss function	mean square error
Iterations	
Batch size	500
Number of epochs	50
Performance of datasets (loss of the last epoch)	
Training	0.63
Validation	0.21

Table 4.4: The characteristics of the Design 4.

It is appreciated a good performance of the NN in the training and the validation set.

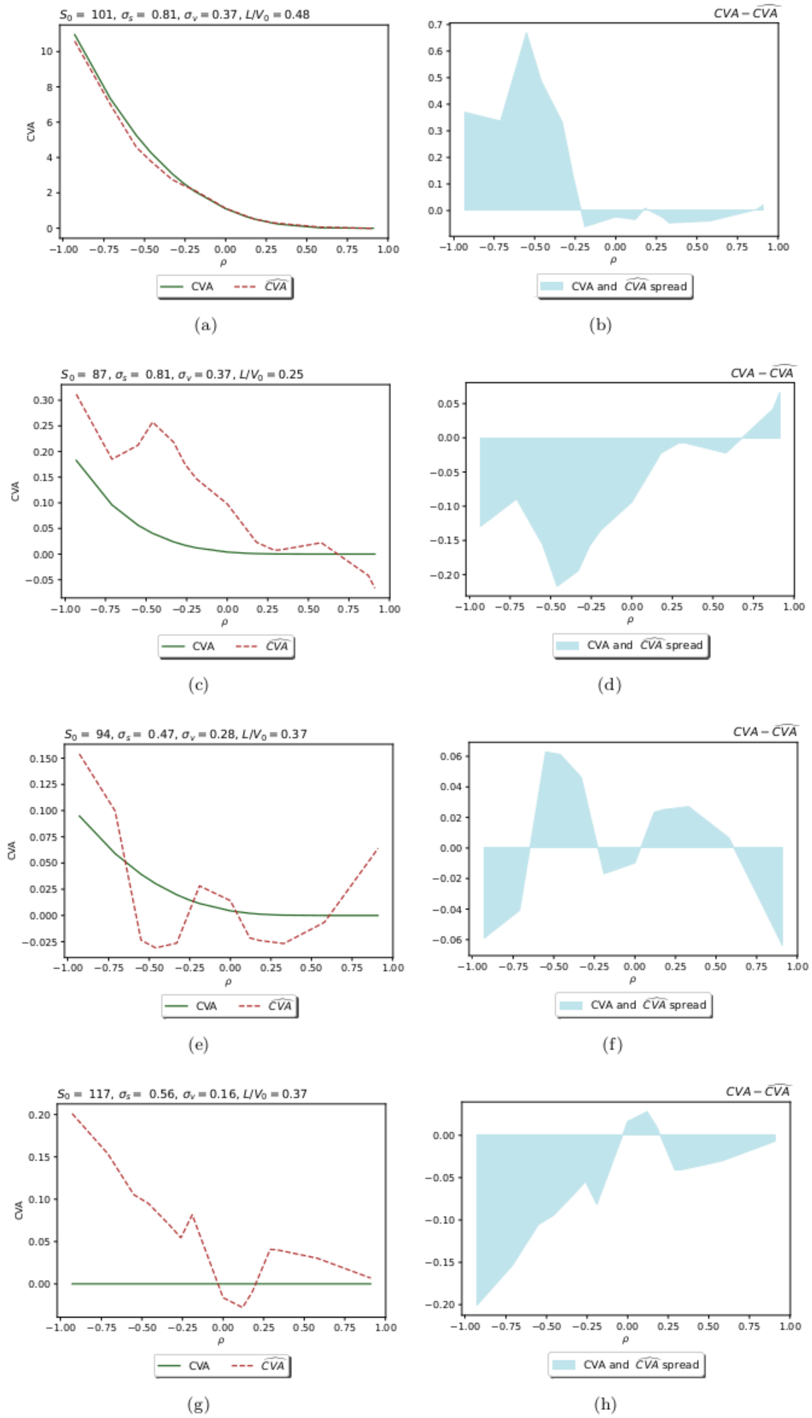


Figure 4.13: Comparative plots of the CVA obtained by numerical integration vs the deep CVA obtained by the neural network under the Design 4.

Design 5: Two hidden layers with 50 neurons each, using ELU as activation function, and ReLU in the output layer

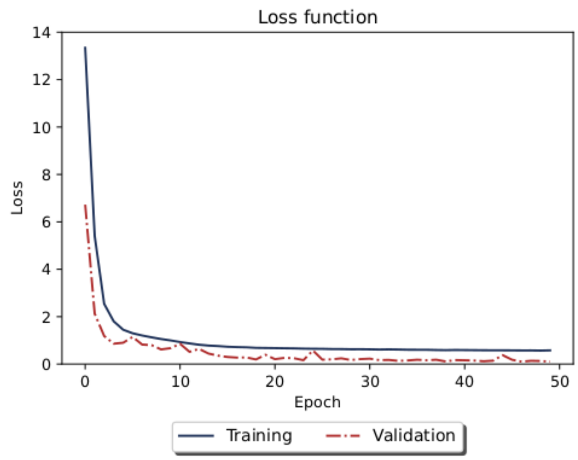


Figure 4.14: The loss function from the training of FNN with Design 5.

Characteristics of Design 5	
Structure of the NN	
Architecture	$\widehat{\text{CVA}} \in \mathcal{N}_3(5, 50, 50, 1; ELU, ELU, ReLU)$
Parameters	2,901
The datasets	
Training	1,600,000 samples
Validation	96,000 samples
Optimizer and loss function	
Optimizer	Adam
Loss function	mean square error
Iterations	
Batch size	500
Number of epochs	50
Performance of datasets (loss of the last epoch)	
Training	0.57
Validation	0.10

Table 4.5: The characteristics of the Design 5.

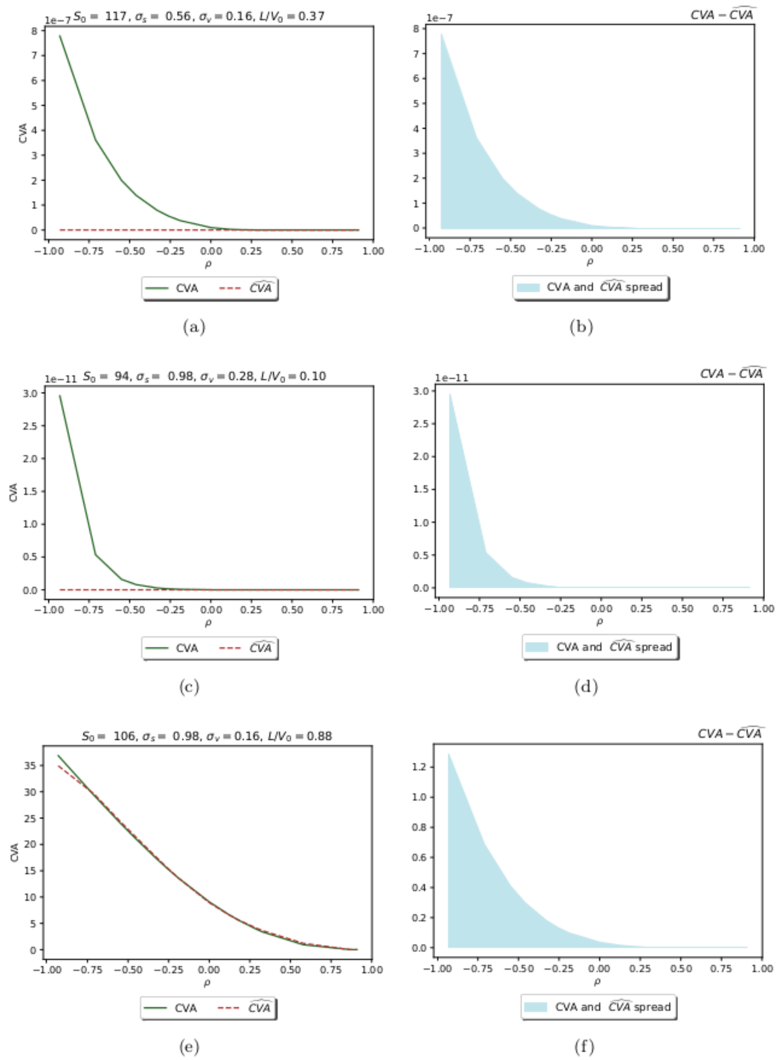


Figure 4.15: Comparative plots of the CVA obtained by numerical integration vs the deep CVA obtained by the neural network under the Design 5.

Design 6: Three hidden layers with 500 neurons each, using ELU and ReLU as activation functions, and the Identity in the output layer

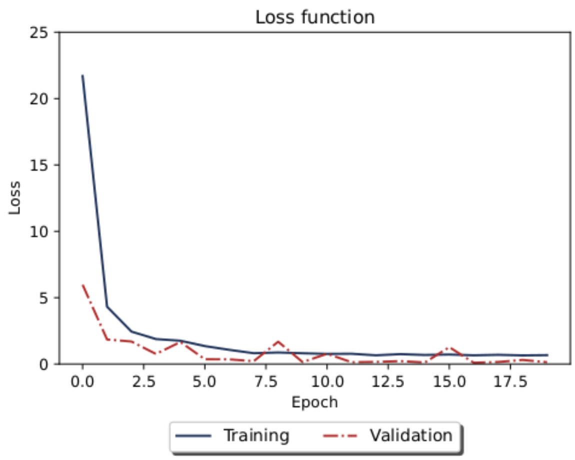


Figure 4.16: The loss function from the training of FNN with Design 6.

Characteristics of Design 6	
Structure of the NN	
Architecture	$\widehat{\text{CVA}} \in \mathcal{N}_4(5, 500, 500, 500, 1; ELU, ELU, ReLU, Id)$
Parameters	504,501
The datasets	
Training	1,600,000 samples
Validation	96,000 samples
Optimizer and loss function	
Optimizer	Adam
Loss function	mean square error
Iterations	
Batch size	2,000
Number of epochs	20
Performance of datasets (loss)	
Training	0.67
Validation	0.14

Table 4.6: The characteristics of the Design 6.

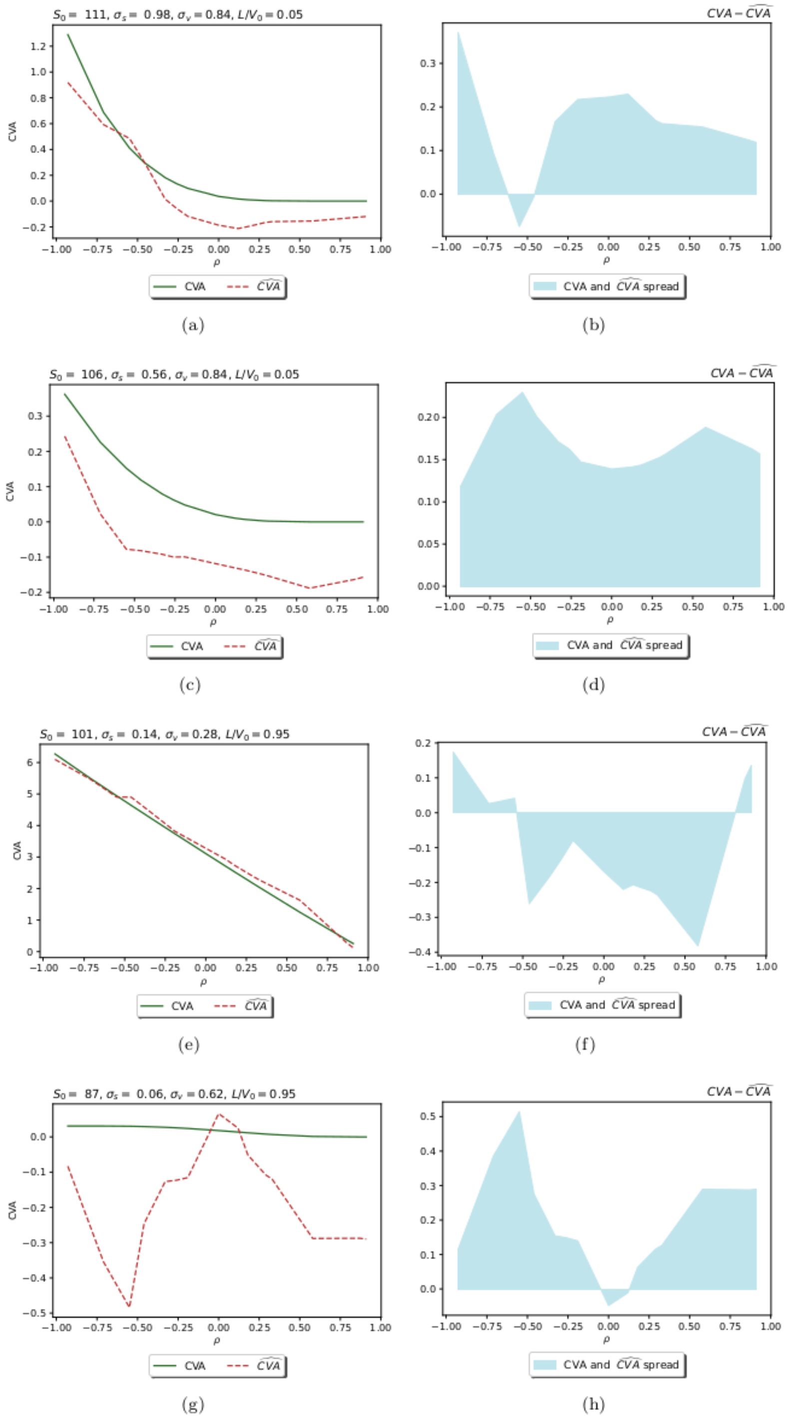


Figure 4.17: Comparative plots of the CVA obtained by numerical integration vs the deep CVA obtained by the neural network under the Design 6.

Design 7: Three hidden layers with 100 neurons each, using ELU as activation functions and dropout to avoid over fitting

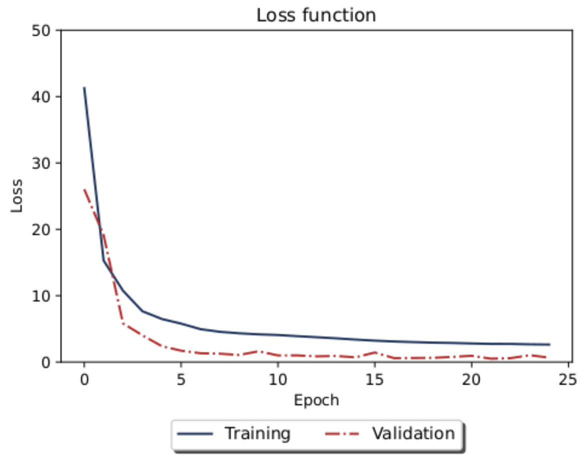


Figure 4.18: The loss function from the training of FNN with Design 7.

Characteristics of Design 7	
Structure of the NN	
Architecture	$\widehat{\text{CVA}} \in \mathcal{N}_4(5, 100, 100, 100, 1; ELU, ELU, ELU, Id)$
Parameters	20,901
The datasets	
Training	1,600,000 samples
Validation	96,000 samples
Optimizer and loss function	
Optimizer	Adam
Loss function	mean square error
Iterations	
Batch size	2,000
Number of epochs	25
Performance of datasets (loss of the last epoch)	
Training	2.64
Validation	0.66

Table 4.7: The characteristics of the Design 7.

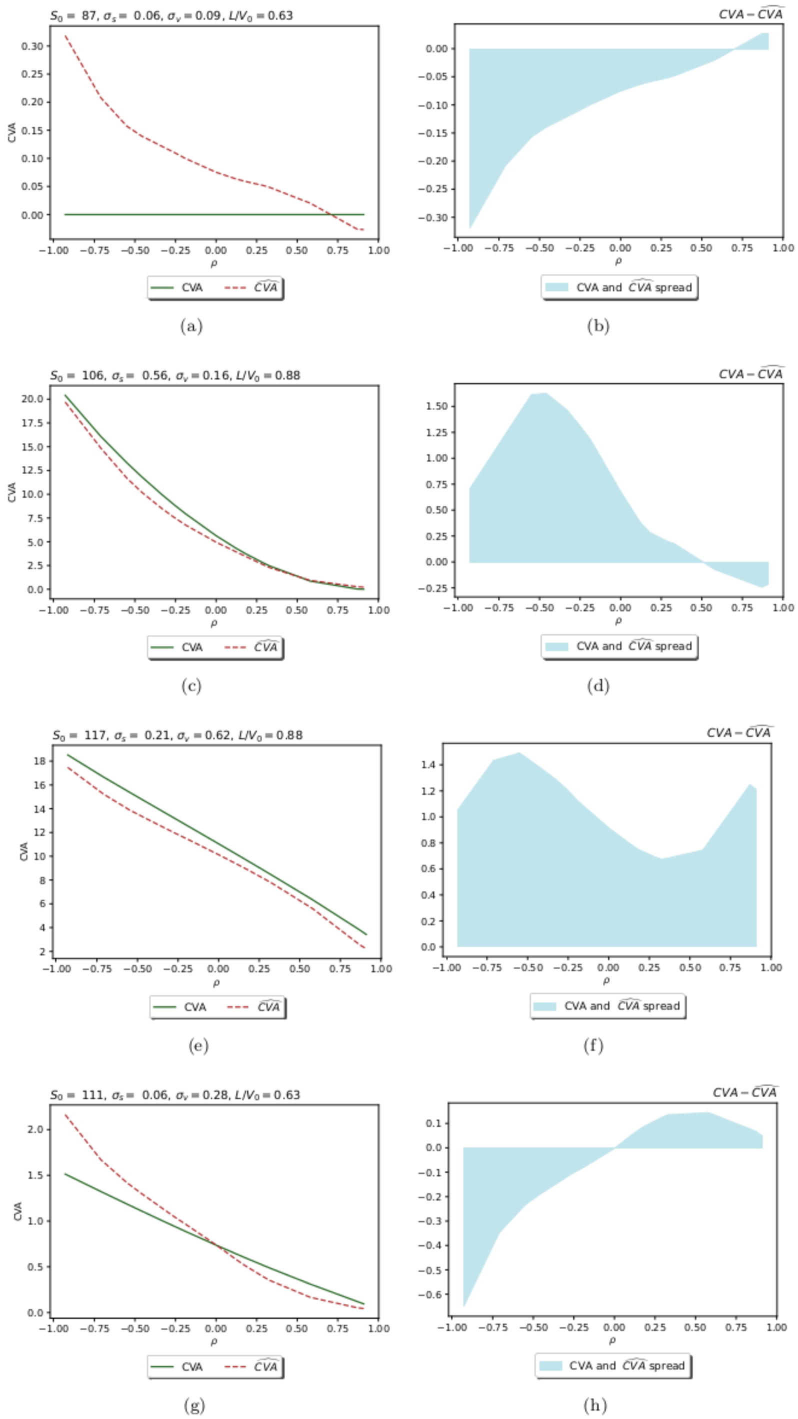


Figure 4.19: Comparative plots of the CVA obtained by numerical integration vs the deep CVA obtained by the neural network under the Design 7.

Design 8: Three hidden layers with 200 neurons each, using ELU as activation functions and dropout to avoid over fitting

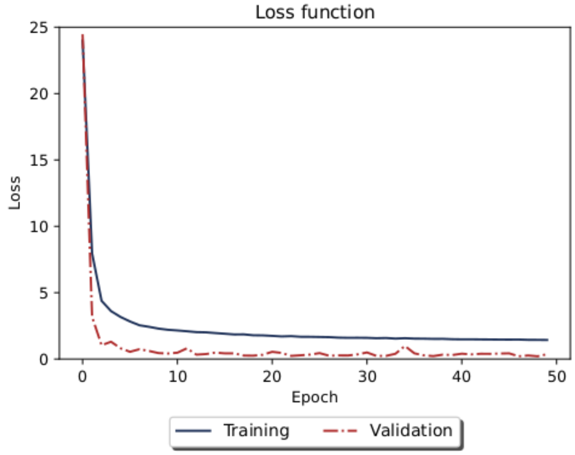


Figure 4.20: The loss function from the training of FNN with Design 8.

Characteristics of Design 8	
Structure of the NN	
Architecture	$\widehat{\text{CVA}} \in \mathcal{N}_4(5, 200, 200, 200, 1; ELU, ELU, ELU, Id)$
Parameters	81,801
The datasets	
Training	1,600,000 samples
Validation	96,000 samples
Optimizer and loss function	
Optimizer	Adam
Loss function	mean square error
Iterations	
Batch size	1,000
Number of epochs	50
Performance of datasets (loss of the last epoch)	
Training	1.434
Validation	0.36

Table 4.8: The characteristics of the Design 8.

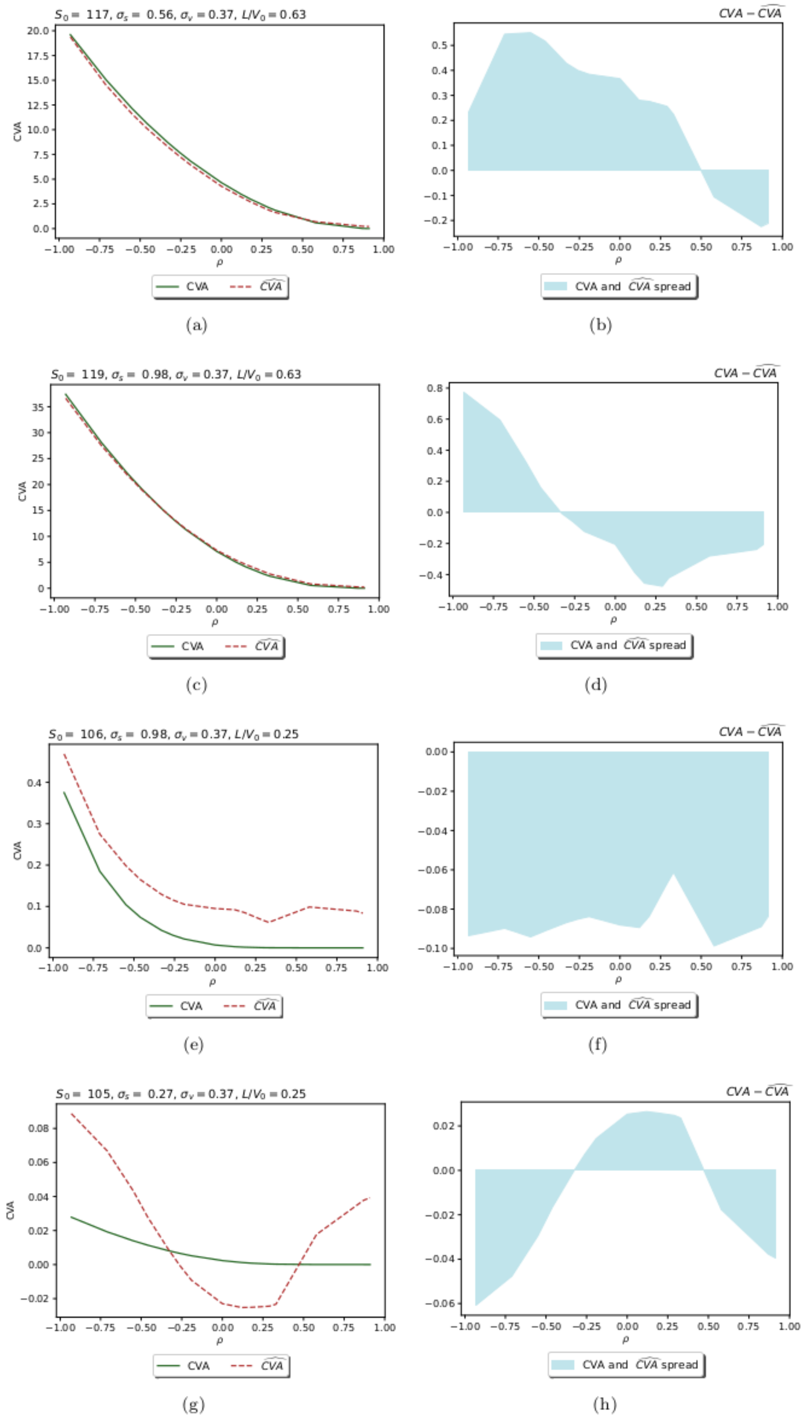


Figure 4.21: Comparative plots of the CVA obtained by numerical integration vs the deep CVA obtained by the neural network under the Design 8.

4.2 Results: choosing the best architecture

The best performance was achieved with the neural network with the design 5 (Figure 4.15). The FNN with this architecture contains 2,901 trainable parameters. The activation function for the two hidden layers is ELU, and the output layer's activation function is ReLU. The performance measured by the loss function (the mean square error) is 0.57 in the training dataset and 0.10 in the validation dataset. This FNN was trained with Adam optimizer and was optimized over 3,200 minibatches of size 500, over 50 epochs (see Table 4.5). This architecture has the characteristic that the output layer only spam the real positive numbers, a good aspect of this design because the CVA is always positive.

The second best formulation in therms of loss is the design 1 (Figure 4.7), which has a 0.13 loss in the validation dataset, and 0.53 in the training dataset. Although the training dataset performs better that the FNN with design 5, the validation dataset is not outperforming the desing 5. Another characteristic of the design 1 is that contains many more parameters than the number of parameters in design 1 (2,901 against 81,801 parameters), this difference does not impact the performance of the design 1. The architecture in the design 1 contains in the output layer a linear function, that means that any real number can be an output.

The worst design is the number 2 (Figure 4.9). This neural network provides a loss of 8.19 for training, and 6.89 for validation. This is the simplest design with just 154 trainable parameters and two hidden layers with ReLU activation function. Despite being a very simple architecture, the loss function is stuck above 8 in the learning process.

An improvement to the design 2 is the design 3 (Figure 4.11). This architecture is constructed with ReLU activation functions in the two hidden layers. The difference relies in the number of nodes within the hidden layers. With 50 nodes each hidden layer makes a total of 2,901 parameters, and the performance is considerably better, with a loss below 0.75. This architecture is good enough for some sets of parameters of the CVA as in Subfigures 4.11(a) and 4.11(h). However, the estimated CVA is erratic (Subfigure 4.11(e)) or has a wide spread against the true CVA (Subfigure 4.11(d)).

The Leaky-ReLU is a function that improves the difficulties that ReLU sometimes presents. In design 4 (see Table 4.4), the activation functions of the hidden layers are Leaky-RLU. With 2,901 parameters, this neural network outperfomes the results of design 3 with a loss below 0.65. The disadvantage of using this architecture is that the estimated CVA from this FNN is not a smooth function for some parameters (see for example Subfigures 4.13(c), 4.11(e), and 4.11(g)).

The design 6 (Figure 4.17) is the FNN with more parameters, with a total of 504,501. This function is composed by three hidden layers of 500 units each. The activation functions for the first two hidden layers is ELU and for the third is ReLU. This NN was trained with dropout layers[2] (probability 0.3), to avoid overfitting. The performance of the FNN was not significatly better than the other designs, and the time spent to train this function was higher. For these and more reasons, the design 6 is not the best option.

Finally, the designs 7 and 8 (Figures 4.19 and 4.21) show a weak performance for the training dataset, with a 2.64 loss in architecture 7 and 1.43 in architecture 8. The first function frequently overestimates the CVA (for example, for the parameters in Subfigure 4.19(a)), and also underestimates the CVA (see Subfigure 4.19(e)). I some cases the CVA output is negative, specially when the true CVA is low (see Subfigure 4.21(g)). Both designs where created with dropout technique but this feature does not helped to improve the leaning.

4.2.1 Computation time for training

In training, computational time depends in the number of iterations and the structure of the NN: The design 1, the time spent to train the FNN was 6.5 seconds on average for each epoch, resulting in 325 seconds. The design 2 last one second for epoch, in total 50 seconds. The design 3, on average 3.5 seconds for epoch, 175 seconds. The design 4, 3 seconds for epoch, 150 seconds. The design 5, 4 seconds for epoch, 200 seconds. The design 6, 22 seconds for epoch, 440 seconds. The

[2]The dropout layer is introduced between each layer in the training process. Each input gets randomly replaced by zero with a probability previously fixed. See the full detail in [39].

design 7, 5 seconds for epoch, 125 seconds in total. The design 8, 8.5 seconds for epoch, 425 seconds in total. Once the FNN is trained, the prediction for each design is almost instantaneously.

Remark. All the computational time presented in this document was extracted from the same computer. We consider all the time measures comparable.

Chapter 5

Conclusions and further work

This thesis provides an alternative method to compute the unilateral CVA under a WWR correlation using machine learning. We investigated the impact of five variables in the CVA within the Merton's model for a portfolio with a Call option under a stock. The variables are: the volatilities of the stock and firm value processes, the initial value of the stock, the debt ratio of the company and the correlation between the stock price and the firm value. We further investigate the influence of such correlation in the CVA by calibrating a feedforward neural network to learn from this.

We found that a deep neural network can learn the calculation of CVA WWR for options in the context of the Merton Model effectively. However, we identified some difficulties increasing the precision of the approximation, hinting at a baseline error in the model. Despite the efficiency of the neural network to compute an estimated CVA, a small-scale error was always present. The best architecture is presented in section 4.1.4.

The most significant advantage of using a deep learning approach was the efficiency. As we pointed out in sections 3.2 and 4.1.2, we made a numerical integration study to build the training and validation datasets, and it took a long time to compute all the samples. In contrast, the neural network took some minutes to build and train, and the expected CVA calculated with this FNN is obtained almost instantaneously. Deep learning could reduce the CVA's computation time by more than 140 times (from 8 hours that took integrate numerically 96,000 CVA's, to 200 seconds that took run the neural network to compute all those CVA's).

Beyond the scope of the thesis

This work has more variants that could be interesting to explore in further work:

- Consider a portfolio of different derivatives. We pointed out that this work was done under a simple portfolio with a Call option under a stock whose price dynamics is a GBM. Something interesting could be to to try with other derivatives.
- Change the firm value model with some of its variants or improvements. For instance, the AT1P model, a more sophisticated model that accounts for default not only at maturity.
- Use instead an intensity model. This type of models correlates the intensity itself with the stock to model WWR. The level of dependence that an intensity model can achieve to the correlation could be limited due to the source of randomness. However, there is a recent work from El Mouden (2021) [32] that concludes that a NN could learn from this correlation in an intensity-based framework.
- Choose a different architecture. We trained a feedforward neural network that could reduce considerably the time of computing the CVA with WWR. This computation involved a baseline error that could not be reduced by any of the proposed designs. However, that does not mean that there aren't more architectures. On the contrary, the only limit only is in the imagination.

- Analyse the impact of each variable in the network. We could make an interpretability analysis of the variables in the FNN. There is a recent paper that applies interpretability techniques in deep learning from Brigo *et al.* (2021) [12].

Appendix A

Probability and Stochastic Calculus

From [5], [8], and [37].

A.1 Wiener process

Definition A.1.1 (Wiener process (also called Brownian motion))**.** Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space. For each $\omega \in \Omega$, supose there is a continuous function $W(t)$ of $t \geq 0$ that satisfies $W(0) = 0$ and that depends on ω. Then $W(t), t \geq 0$, is a Brownian motion if for all $0 = t_0 < t_1 < \cdots < t_m$ the increments

$$W(t_1) = W(t_1) - W(t_0),\ W(t_2) - W(t_1), \ldots,\ W(t_m) - W(t_{m-1})$$

are the independent and each of these increments is normally distributed with

$$\mathbb{E}[W(t_{i+1}) - W(t_i)] = 0,$$
$$\mathrm{Var}[W(t_{i+1}) - W(t_i)] = t_{i+1} - t_i.$$

A.2 Stochastic Differential Equations

Definition A.2.1 (Stochastic Differential equation (SDE))**.** Let $\mathcal{M}(n, d)$ denote the class of $n \times d$ matrices. Given W a d-dimensional Wiener process, a function $\mu : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$, a function $\sigma : \mathbb{R}^+ \times \mathbb{R}^n \to \mathcal{M}(n, d)$, and a real vector $x_0 \in \mathbb{R}^n$. The SDE is

$$X_t = \mu(t, X_t)dt + \sigma(t, X_t)dW_t, \tag{A.2.1}$$
$$X_0 = x_0, \tag{A.2.2}$$

with a solution, when exists, the process X that satisfies the equation

$$X_t = x_0 + \int_0^t \mu(t, X_t)dt + \int_0^t \sigma(t, X_t)dW_t, \quad \text{for all } t \geq 0. \tag{A.2.3}$$

The standard method that provides the existence of X, the solution to the SDE, is constructing an iteration Scheme of Couchy-Picard type to obtain a sequence converging to a limiting process if the conditions in [5, Proposition 5.1, pp. 68] are satisfied.

A.3 Geometric Brownian Motion

Proposition A.3.1 (Geometric Brownian motion). *The solution to the equation*

$$dX_t = \alpha X_t dt + \sigma X_t dW_t, \tag{A.3.1}$$
$$X_0 = x_0, \tag{A.3.2}$$

is given by

$$X(t) = x_0 \cdot \exp\left\{\left(\alpha - \frac{1}{2}\sigma^2\right) t + \sigma W(t)\right\} \tag{A.3.3}$$

Proof. We can write the equation A.3.1 as

$$\dot{X}_t = \left(\alpha + \sigma \dot{W}_t\right) X_t$$

where W is a Wiener process. The solution to the corresponding deterministic linear equation is an exponential function of time. We define Z such that $Z_t = \log X_t$, and assume X the solution, being strictly positive. Applying the Itô's formula (see A.4) we have

$$\begin{aligned} dZ &= \frac{1}{X}dX + \frac{1}{2}\left\{-\frac{1}{X^2}\right\}[dX]^2 \\ &= \frac{1}{X}\{\alpha X dt + \sigma X dW\} + \frac{1}{2}\left\{-\frac{1}{X^2}\right\}\sigma^2 X^2 dt \\ &= \{\alpha dt + \sigma dW\} - \frac{1}{2}\sigma^2 dt. \end{aligned}$$

Hence,

$$\begin{aligned} dZ_t &= \left(\alpha - \frac{1}{2}\sigma^2\right) dt + \sigma dW_t \\ Z_0 &= \log x_0. \end{aligned}$$

Integrating in both sides and using the initial condition,

$$Z_t = \log x_0 + \left(\alpha - \frac{1}{2}\sigma^2\right) + \sigma W_t$$

that in terms of X is

$$X_t = x_0 \cdot \exp\left\{\left(\alpha - \frac{1}{2}\sigma^2\right) + \sigma W_t\right\}$$

□

A.4 Itô's formula

Definition A.4.1 (Itô's formula). Let be the stochastic differential equation $dX_t = f(X_t)dt + \sigma(X_t)dW_t$ and $\varphi(\cdot, x)$ a smooth function. the Itô's formula says

$$d\varphi(t, X_t) = \frac{\partial \varphi}{\partial t}dt + \frac{\partial \varphi}{\partial X}dX_t + \frac{1}{2}\frac{\partial^2 \varphi}{\partial X^2}dX_t dX_t$$

where $dX_t dX_t$ is the quadratic variation.

Bibliography

[1] E. Altman, A. Resti, and A. Sironi. Default recovery rates in credit risk modelling: A review of the literature and empirical evidence. *Economic Notes by Banca Monte dei Paschi di Siena SpA*, 33(2):183–208, 2004.

[2] M. Ammann. *Credit Risk Valuation: Methods, Models, and Applications*. Springer Finance. Springer Berlin Heidelberg, 2002.

[3] Nurshazlyn Mohd Aszemi and PDD Dominic. Hyperparameter optimization in convolutional neural network using genetic algorithms. *Int. J. Adv. Comput. Sci. Appl*, 10(6):269–278, 2019.

[4] T.R. Bielecki and M. Rutkowski. *Credit Risk: Modeling, Valuation and Hedging*. Springer Finance. Springer Berlin Heidelberg, 2013.

[5] Tomas. Björck. *Arbitrage theory in continuous time*. Oxford Finance Series. Oxford University Press, Oxford, 3rd ed. edition, 2009.

[6] F. Black and M. Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 1973.

[7] Fischer Black and John C Cox. Valuing corporate securities: Some effects of bond indenture provisions. *Journal of Finance*, 31(2):351–67, 1976.

[8] Damiano Brigo. Interest rate models with credit risk, collateral, funding liquidity risk and multiple curves. Lecture notes (MATH97114), MSc Mathematics and Finance, IC, 2020-2021.

[9] Damiano Brigo and Agostino Capponi. Bilateral counterparty risk valuation with stochastic dynamical models and application to credit default swaps, 2009.

[10] Damiano Brigo, Agostino Capponi, Andrea Pallavicini, and Vasileios Papatheodorou. Collateral margining in arbitrage-free counterparty valuation adjustment including re-hypotecation and netting, 2011.

[11] Damiano Brigo, Marco Francischello, and Andrea Pallavicini. An indifference approach to the cost of capital constraints: Kva and beyond, 2017.

[12] Damiano Brigo, Xiaoshan Huang, Andrea Pallavicini, and Haitz Sáez de Ocáriz Borde. Interpretability in deep learning for finance: a case study for the heston model. *Available at SSRN 3829947*, 2021.

[13] Damiano Brigo and Fabio Mercurio. *Interest Rate Models Theory and Practice*. Springer Finance. Springer Berlin Heidelberg, 2013.

[14] Damiano Brigo, Massimo Morini, and Andrea Pallavicini. *Counterparty credit risk, collateral and funding: with pricing cases for all asset classes*. Wiley finance series. Wiley, Chichester Hoboken, N.J, 1st ed. edition, 2013.

[15] Damiano Brigo and Marco Tarenghi. Credit default swap calibration and equity swap valuation under counterparty risk with a tractable structural model. *Available at SSRN 581302*, Mar 2004.

[16] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus), 2016.

[17] Michel Crouhy, Dan Galai, and Robert Mark. A comparative analysis of current credit risk models. *Journal of Banking Finance*, 24(1):59–117, 2000.

[18] C. J. Maddison A. Guez L. Sifre G. van den Driessche J. Schrittwieser I. Antonoglou V. Panneershelvam M. Lanctot S. Dieleman D. Grewe J. Nham N. Kalchbrenner I. Sutskever T. Lillicrap M. Leach K. Kavukcuoglu T. Graepel D. Silver, A. Huang and D. Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529:484–489, Jan 2016.

[19] Darrell Duffie and Ming Huang. Swaps rates and credit quality. *The Journal of Finance*, 51(3):921–949, 1996.

[20] Damir Filipovic. *Term-structure models : a graduate course*. Springer Finance. Springer, Berlin ;, 2009.

[21] A. Green. *XVA: Credit, Funding and Capital Valuation Adjustments*. The Wiley Finance Series. Wiley, 2015.

[22] Martin Hellwig. Twelve Years after the Financial Crisis—Too-big-to-fail is still with us. *Journal of Financial Regulation*, 7(1):175–187, 01 2021.

[23] John C. Hull. *Options, futures, and other derivatives*. Pearson Prentice Hall, 11. ed., pearson internat. ed edition, 2021.

[24] C. Kenyon and R. Stamm. *Discounting, LIBOR, CVA and Funding: Interest Rate and Credit Pricing*. Applied Quantitative Finance. Palgrave Macmillan, 2012.

[25] M. Kijima and T. Suzuki. A jump-diffusion model for pricing corporate debt securities in a complex capital structure. *Quantitative Finance*, 1(6):611–620, 2001.

[26] Y.K. Kwok. *Mathematical Models of Financial Derivatives*. Springer Finance. Springer Berlin Heidelberg, 2008.

[27] D. Lando. *Credit Risk Modeling: Theory and Applications*. Princeton University Press. Princeton University Press, 2004.

[28] Moshe Leshno, Vladimir Ya. Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, 6(6):861–867, 1993.

[29] Y. Mahindroo, E. Barron, and M. Codling. Xva explained. valuation adjustments and their impact on the banking sector. *pwc (https://www.pwc.com.au/)*, 1, December 2015.

[30] Robert C Merton. On the Pricing of Corporate Debt: The Risk Structure of Interest Rates. *Journal of Finance*, 29(2):449–470, May 1974.

[31] M. Morini and A. Prampolini. Risky funding: A unified framework for counterparty and liquidity charges. *Banca IMI and Bocconi University*, 2010.

[32] Wiam EL Mouden. Deep intensity-based cva with wrong way risk. Master thesis, Imperial College London, 2020-2021.

[33] M. Pakkanen. Deep learning. Lecture notes (MATH97231), Department of Mathematics, Imperial College London, 2020-2021.

[34] Andrea Pallavicini, Daniele Perini, and Damiano Brigo. Funding valuation adjustment: a consistent framework including cva, dva, collateral,netting rules and re-hypothecation, 2011.

[35] M. Pykhtin and S. H. Zhu. The crash sonata in d major. *BASEL II HANDBOOK, Michael Ong, ed., RISK Books*, (1), 2006.

[36] Jacob T. Schwartz. The new connectionism: Developing relationships between neuroscience and artificial intelligence. *Daedalus*, 117(1):123–141, 1988.

[37] Steven E. Shreve. *Stochastic calculus for finance. II, Continuous-time models*. Springer finance. Springer, New York, 2004.

[38] A.R. Sorkin. *Too Big to Fail: The Inside Story of How Wall Street and Washington Fought to Save the Financial System–and Themselves*. Penguin Publishing Group, 2010.

[39] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[40] Giorgio Szegö. The crash sonata in d major. *Journal of Risk Management in Financial Institutions*, 3(1):31–45, 2009.

[41] H. Thorin. Artificial neural networks for sabr model calibration hedging. Master thesis, Imperial College London, 2019-2020.

[42] K. Webster. Methods for data science: Deep learning. Video lectures (MATH97097), Department of Mathematics, Imperial College London, 2020-2021.

[43] Jia Wu, Xiu-Yun Chen, Hao Zhang, Li-Dong Xiong, Hang Lei, and Si-Hao Deng. Hyperparameter optimization for machine learning models based on bayesian optimizationb. *Journal of Electronic Science and Technology*, 17(1):26–40, 2019.

[44] Y. Zhang. *New Advances in Machine Learning*. IntechOpen, 2010.

TOSCANO_MARIANNE_01894118

GRADEMARK REPORT

FINAL GRADE

/0

GENERAL COMMENTS

Instructor

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

PAGE 33

PAGE 34

PAGE 35

PAGE 36

PAGE 37

PAGE 38

PAGE 39

PAGE 40

PAGE 41

PAGE 42

PAGE 43

PAGE 44

PAGE 45

PAGE 46

PAGE 47

PAGE 48

PAGE 49

PAGE 50

PAGE 51

PAGE 52

PAGE 53

PAGE 54

PAGE 55

PAGE 56

PAGE 57

PAGE 58

PAGE 59

PAGE 60

PAGE 61

PAGE 62

PAGE 63

PAGE 64

PAGE 65

PAGE 66

PAGE 67

PAGE 68

PAGE 69

PAGE 70

PAGE 71