ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

DOCTORAL THESIS

Discrete, continuous and machine learning models with applications in credit risk

Author:
Kyriakos GEORGIOU

Supervisor:
Professor Athanasios N. YANNACOPOULOS

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Stochastic Modeling and Applications Laboratory
Department of Statistics

September 13, 2023

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Abstract

Athens University of Economics and Business
Department of Statistics

Doctor of Philosophy

Discrete, continuous and machine learning models with applications in credit risk

by Kyriakos GEORGIOU

Credit risk modelling is a versatile and dynamic area of financial mathematics, with important practical implications, as has been historically established. In particular, the last financial crisis made it clear that credit risk models had to become more rigorous. For this reason, the recent International Financial Reporting Standards (IFRS) 9 have introduced a framework making credit risk modelling forward-looking, thereby increasing the need for robust mathematical tools. The goal of this thesis is to develop and explore such mathematical tools and models, motivated by specific open problems that arise due to these regulations, and develop frameworks that are both mathematically-sound and can be efficiently applied by practitioners.

We begin with discrete models, specifically Markov chains which are well established in the field of credit risk, and develop a framework that can be implemented by financial institutions for credit rating reporting and compliance purposes under IFRS 9. Subsequently, we consider continuous-time stochastic models and study how these can be used for probability of default estimation and provisioning calculations. Specifically, we use a general family of models to incorporate various latent variables on which the credit exposure may depend, and use approaches relying on Integral and Partial Integro-differential Equations to describe and prove important mathematical properties of the resulting probability of default process.

To show how these mathematical tools can be implemented by practitioners, we develop and study numerical methods that can be used for the estimation of default probabilities. We use the well-known Finite Difference methods, which we apply to the equations that arise under various types of stochastic models, to illustrate the variety of uses these approaches can find. Lastly, we draw from recent research to also consider how the modern field of machine learning models, and particularly the family of Deep Neural Networks, can be used to estimate the default probabilities, and discuss important theoretical and practical considerations that should be taken into account when comparing these to the Finite Difference methods.

Acknowledgements

I feel very lucky in that, if left uncontrolled, the things I would like to write about the people who helped and made this experience memorable to me is probably too long to fit in one page. First and foremost, I am grateful to my parents for their continuous and unconditional support, without which I would have never have been able to pursue the path I wanted. I am very lucky to be their son. I am also thankful to my brother and friends, who have been there for me throughout every phase of this pursuit, the good and difficult; the value of their encouragement, understanding and the discussions we had during these past years is immeasurable.

During these past years I was also lucky enough to be part of the Finloup founding team. I am extremely grateful to Antonis, Marios and Stelios for their trust in me and their support. They showed constant understanding and flexibility, which made it possible for me to manage my time, effort and, of course, stress levels, so that I was able to pursue our common goals along with this PhD. I feel grateful to the entire Finloup team, and proud to be working with such incredible people everyday.

Lastly, but definitely not least, I want to give special thanks to my supervisor Professor A.N. Yannacopoulos. From very early on, I understood that he is not only an extraordinary mathematician, but also an extraordinarily kind, helpful and encouraging person. I have learned so much from him in the past four years, that accompany me in my professional and personal life. I am honoured to call him a mentor and thank him for his guidance, which allowed me to extract the most knowledge and joy possible from this experience.

Contents

List of Figures

List of Tables

To my family and Uncle Savvas.

Chapter 1

Introduction

1.1 Credit risk management and recent developments under IFRS 9

One of the main issues currently concerning financial institutions is the implementation of the new International Financial Reporting Standards (IFRS) 9. Due to the financial crisis, the purpose of the updated standards is to introduce a framework under which institutions forecast credit losses (for loan provisioning purposes). Specifically, “under the impairment approach in IFRS 9 it is no longer necessary for a credit event to have occurred before credit losses are recognised. Instead, an entity always accounts for expected credit losses, and changes in those expected credit losses. The amount of expected credit losses is updated at each reporting date to reflect changes in credit risk since initial recognition and, consequently, more timely information is provided about expected credit losses”. Furthermore, "the objective of the impairment requirements is to recognise lifetime expected credit losses for all financial instruments for which there have been significant increases in credit risk since initial recognition — whether assessed on an individual or collective basis — considering all reasonable and supportable information" (IFRS 9 Red Book). Hence, loan provisioning regulations under IFRS 9, require financial institutions to consider expected losses based on the current credit state of each loan and depending on its change in credit risk. Therefore, all loans which have displayed an increase in credit risk must be aggregated by the institution when considering their provisions, which must also account for future expected losses.

These impositions create important and challenging credit modelling tasks. Estimating future losses for provisioning purposes requires forecasting key credit risk parameters such as the Probability of Default of a credit exposure. Furthermore, under IFRS 9 all internal models must now be compatible with the predetermined classifications imposed by the new regulations. This is true regardless of the institution’s internal rating model. Based on these requirements, it is of great importance to address various modelling tasks that arise using mathematically rigorous and robust approaches. Since loan exposures dynamics evolve over time, using stochastic processes in credit risk modelling has now become standard in the financial mathematics field, as well as in practice. We will consider various such settings, ranging from discrete to continuous stochastic processes and their applications to important modelling problems that arise as a consequence of the IFRS 9 implementation.

We first present some standard stochastic processes and related results that will form the basis of the models examined throughout.

1.2 Mathematical Preliminaries

1.2.1 Markov chains

We begin by recalling some fundamental concepts related to the framework of Markov chain modelling.

1.2.1.1 Discrete time Markov Chains

Definition 1.2.1 (Discrete state Markov Chain)**.** A discrete state Markov Chain M, defined on a state space S, of cardinality $|S| = n < \infty$, is a stochastic process $\{X_t : t \in \mathbb{N}\}$, such that

$$\mathbb{P}(X_t = s_t | X_{t-1} = s_{t-1}, \cdots, X_1 = s_1) = \mathbb{P}(X_t = s_t | X_{t-1} = s_{t-1}) \tag{1.1}$$

for any states $s_i \in S$.

For a general Markov Chain, we will use the notation $M = (S, P)$, with S defined as above and $P = (P_{i,j})_{i,j=1,\ldots n}$ is the row-stochastic transition matrix of the Markov Chain, whose entries $P_{i,j}$ reflect the conditional probability of transition from one state of the chain to another, i.e $P_{i,j} = \mathbb{P}(X_t = j | X_{t-1} = i)$, for a fixed $n \in \mathbb{N}$.

For the Markov chain $M = (S, P)$, we say that a state i is transient if, given that the chain starts at i, it is possible, but not certain, that the chain will return to i. Equivalently, there exists a non-zero probability that the chain will never return to i. We may define the random variable T_i, as $T_i = \inf\{t \geq 1 : X_t = i\}$ and say that state i of the Markov chain is transient if $P(T_i < \infty | X_0 = i) = \sum_{t=1}^{\infty} P(T_i = t | X_0 = i) < 1$. On the other hand, a state i is defined as absorbing if $P_{i,i} = 1$ i.e., the probability of transitioning from i to any other state is zero. Hence, the state space of a Markov chain can be written as $S = \mathcal{A} \cup \mathcal{T}$, where $\mathcal{A}$ and $\mathcal{T}$ denote the set of absorbing and transient states, respectively. We assume that $|\mathcal{A}| = r$ and hence, naturally, $|\mathcal{T}| = n - r$.

Definition 1.2.2 (Canonical form)**.** It is often convenient to use the canonical form of the transition matrix, which is constructed by relabelling the states so as to bring P into the following block form:

$$P = \begin{pmatrix} Q & R \\ \mathbf{0}_{r \times (n-r)} & I_{r \times r} \end{pmatrix},$$

where $Q \in \mathbb{R}^{(n-r) \times (n-r)}$, contains the transition probabilities between transient states, $R \in \mathbb{R}^{(n-r) \times r}$ contains the transition probabilities from the transient to the absorbing states of the chain, $\mathbf{0}_{r \times (n-r)}$ is the $r \times (n-r)$ zero matrix and $I_{r \times r}$ is the $(n-m) \times (n-m)$ identity matrix. The $\mathbf{0}$ block corresponds to the probabilities of transitioning from an absorbing to a transient state, which, by definition, have to be 0.

An important quantity in the study of Markov chains, with applications that will be explored in subsequent sections, is the fundamental matrix, defined below.

Definition 1.2.3 (Fundamental matrix)**.** The fundamental matrix N of a Markov Chain $M = (S, P)$, is defined as

$$N = I + Q + Q^2 + Q^3 + \cdots = (I - Q)^{-1},$$

where $N_{i,j}$ gives the expected time spent in transient state j, conditional on starting from transient state i.

1.2.1.2 Continuous Time Markov Chains

We continue by outlining the background and important results pertaining to Continuous Time Markov Chains (CTMC), which are used when considering continuous stochastic models in Chapter 3.

Definition 1.2.4. A continuous time Markov chain is a continuous stochastic process $X_t, t \geq 0$, with a discrete state space $\mathcal{R}$, of cardinality $|\mathcal{R}| < \infty$, satisfying the Markov property, and such that:

$$\mathbb{P}(X_{t+\delta} = j | X_t = i) = \begin{cases} q_{ij}\delta + o(\delta), & i \neq j \\ 1 + q_{ii}\delta + o(\delta), \end{cases} \tag{1.2}$$

as $\delta \downarrow 0$. In the above q_{ij} are known as the transition rates, for which we have $\sum_{i \in S} q_{ij} = 0, q_{ij} \geq 0$ for $i \neq j$.

The matrix Q with entries $(Q)_{ij} = q_{ij}$, for $i, j \in \mathcal{R}$ is known as the generator matrix of the Markov process (also referred to as the transition rate matrix). Similar to the discrete time Markov chains, we can define the transition matrix for a CTMS, $P(t), t \geq 0$, with entries:

$$p_{ij}(t) = \mathbb{P}(X_t = j | X_0 = i), \tag{1.3}$$

for $i, j \in \mathcal{R}$. The following result holds for the transition matrix, from which we are also able to obtain a connection between the transition and generator matrices.

Theorem 1.2.5. *The transition matrix $P(t)$ satisfies the Kolmogorov forward equation:*

$$P'(t) = P(t)Q,$$

and hence:

$$P(t) = e^{tQ}, \tag{1.4}$$

for $t \in [0, \infty]$, (naturally, $P(0)$ is equal to the corresponding identity matrix).

1.2.2 Additional matrix operations

1.2.2.1 The Moore-Penrose inverse

Now recall the notion of the Moore-Penrose inverse. Let $A \in \mathbb{R}^{n \times m}$ with full rank; it can be shown that there exists a unique matrix, the Moore-Penrose inverse which is denoted by $A^\dagger \in \mathbb{R}^{m \times n}$, satisfying the following four Penrose equations:

$$AA^\dagger A = A, \quad A^\dagger AA^\dagger = A^\dagger, \quad AA^\dagger = (AA^\dagger)^T, \quad A^\dagger A = (A^\dagger A)^T \tag{1.5}$$

One can easily verify that $AA^\dagger$ is the orthogonal projection of $\mathbb{R}^n$ onto the range $\mathcal{R}(A)$, denoted by Π_A, and that $A^\dagger A$ is the orthogonal projection of $\mathbb{R}^m$ onto $\mathcal{R}(A^T)$ denoted by Π_{A^T}, where by A^T we denote the transpose of A. It is also well known that $\mathcal{R}(A^\dagger) = \mathcal{R}(A^T)$.
Let us also recall the standard minimization property of the Moore-Penrose inverse:

Proposition 1.2.6. *Let $A \in \mathbb{R}^{n\times m}$ and $b \in \mathbb{R}^r$ with $b \in \mathcal{R}(A)$ and the equation $Ax = b$. Then, if $A^\dagger$ is the Moore-Penrose inverse of A, we have that $A^\dagger b = u$, where u is the minimal ℓ^2 norm solution of the equation.*

For more details on the Moore-Penrose inverse and pseudoinverses in general, standard reference books are e.g. Ben-Israel and Greville, 2003; Campbell and Meyer, 2009; Groetsch, 1977.

1.3 Matrix operations and vectorization

Definition 1.3.1. The Kronecker product of two matrices $A = (A_{i,j})_{i=1,\dots,n}^{j=1,\dots,m} \in \mathbb{R}^{n\times m}$ and $B = (B_{i,j})_{i=1,\dots,p}^{j=1,\dots,q} \in \mathbb{R}^{p\times q}$, denoted $A \otimes B$, is the $np \times mq$ dimensional matrix

$$A \otimes B := \begin{pmatrix} A_{1,1}B & \cdots & A_{1,m}B \\ A_{2,1}B & \cdots & A_{2,m}B \\ \vdots & \cdots & \vdots \\ A_{n,1}B & \cdots & A_{n,m}B \end{pmatrix},$$

where $A_{i,j}$ is the i^{th} row, j^{th} column entry of the matrix A.

A result which will prove to be of particular use in subsequent sections, is that the matrix equation $AXB = C$, can be re-written as a system of linear equations, using the Kronecker product. Let $X \in \mathbb{R}^{n\times m}$, then we use $x = vec(X)$ to denote the vectorizations of X, by stacking the rows of the matrix into column vectors. Then,

$$AXB = C \Rightarrow (A \otimes B^T)x = c.$$

Hereinafter, we will use the convention that upper-case letters denote matrices and lower-case denote their vectorized forms.

Even though the vectorization of a matrix is a well known procedure and functions that perform this operation are readily available in many programming packages, we include the following functions that perform this operation, for completeness. Moreover, it may be useful for practitioners to have their explicit forms, which can be applied regardless of the platform. To do this for an arbitrary n-dimensional square matrix P (resulting in a column vector $p \in \mathbb{R}^{n^2}$), define the function $\psi : \{1,2,\dots,n\} \times \{1,2,\dots,n\} \to \{1,2,\dots,n^2\}$, by $\psi(i,j) = (i-1)n + j$. Therefore, if we represent the entries of the square matrix P and the vector $p = vec(P)$ by $P_{i,j}$ and p_i, respectively, it holds that $p_{\psi(i,j)} = P_{i,j}$. For example, the first entry of a three-dimensional square matrix, $P_{1,1}$ will be mapped to p_1 since $\psi(1,1) = (1-1)\times 3 + 1 = 1$ and the last entry of the matrix $P_{3,3}$ will be mapped to p_9 in the vector, as required, since $\psi(3,3) = (3-1)\times 3 + 3 = 9$.

Given that the solutions of the problems we will be examining should be in matrix form, we must also define (with a slight abuse of notation) the inverse $\psi^{-1} : \{1,2,\dots,n^2\} \to \{1,2,\dots,n\} \times \{1,2,\dots,n\}$, which will relabel the solution back into matrix form. To define the inverse function we first extrapolate the column value j. From the definition of ψ we see that $j = \psi(i,j) \mod n$, for $j \in \{1,2,\dots,n-1\}$. When $\psi(i,j) \bmod n = 0$, this means that $j = n$. Hence, for $j \in \{1,2,\dots,n\}, j = \psi(i,j) \bmod n + n\mathbb{1}_{\{\psi(i,j) \mod n = 0\}}$. Using the definition, $\psi(i,j) = (i-1)n + j$ we can substitute in the expression for j and solve to obtain the row values i. For clarity, the

full output of this inverse function is given. For $k \in \{1, 2, \ldots, n^2\}$:

$$\psi^{-1}(k) = \left(\frac{k - k \bmod n - n\mathbb{1}_{\{k \bmod n = 0\}}}{n} + 1, k \bmod n + n\mathbb{1}_{\{k \bmod n = 0\}}\right).$$

One can easily check that by calculating $P_{\psi^{-1}(k)} = p_k$, for $k \in \{1, 2, \ldots, n^2\}$ we obtain original form of the square matrix.

1.3.1 Lévy processes

We move on to a general class of continuous stochastic processes, known as Lévy processes. Throughout this work, we have abopted the notation used in Øksendal and Sulem, 2007.

Definition 1.3.2 (Lévy process)**.** A Lévy process $\{L_t\}_{t \geq 0}$ is a stochastic process for which the following conditions hold:

- $L_0 = 0$.
- L has independent and stationary increments, i.e., if $t > s$ then $L_t - L_s$ is independent from L_s and $L_t - L_s \overset{D}{=} L_{t-s}$.
- L is stochastically continuous, i.e for all $\epsilon > 0$ and all $s > 0$ we have

$$\lim_{t \to s} \mathbb{P}(|X(t) - X(s)| > \epsilon) = 0.$$

A consequence of the above definition is the celebrated Itô - Lévy decomposition. First, we define the following required quantities:

Definition 1.3.3. Let L_t be a Lévy process, whose jump is defined as $\Delta L_t = L_t - L_{t\_}$. Furthermore, let $\mathbf{B}_0$ be the family of Borel sets $U \subset \mathbb{R}$, whose closure does not contain 0. Then, for $U \in \mathbf{B}_0$, define the Poisson random measure of the Lévy process L_t by:

$$N(t, U) = \sum_{0 < s \leq t} \mathbb{1}(\Delta L_s).$$

The Poisson random measure represents the number of jumps of size $\Delta L_s \in U$, which occur up to time t. We can therefore define the intensity of the jumps as follows:

Definition 1.3.4. The intensity of a Lévy jump process L_t, known as the Lévy measure of L_t is defined as:

$$\nu(U) = \mathbb{E}[N(1, U)],$$

where, as above, $U \in \mathbf{B}_0$.

A useful consequence of the above definitions is that if ν is the Lévy measure of a simple Compound Poisson Process with rate λ and jump size density $f(z)$, then we have that

$$\nu(U) = \lambda f(U).$$

To this end, we will employ the following result in the subsequent sections, due to Kyprianou, 2006:

Theorem 1.3.5. *Consider the Poisson random measure $N(t,U)$, with $U \in \boldsymbol{B}_0$, and corresponding Lévy measure $\nu(U)$. Then the process:*

$$X_t = \int_0^t \int_B zN(ds,dz),$$

where $B \in \mathcal{B}(\mathbb{R})$, is a Compound Poisson Process with rate $\nu(B)$ and jump distribution $\frac{\nu(dx)|_B}{\nu(B)}$.

Throughout Chapter 3, we will consider jump terms in the form above. We can now present the following celebrated theorem:

Theorem 1.3.6 (Itô - Lévy decomposition)**.** *Let $\{L_t\}_{t\geq 0}$ be a Lévy process. Then, we have*

$$L_t = bt + \sigma B_t + \int_{|z|<1} z\tilde{N}(t,dz) + \int_{|z|\geq 1} zN(t,dz), \tag{1.6}$$

for $t \geq 0$, where $b, \sigma \in \mathbb{R}$, B_t is a Brownian motion and $\tilde{N}(t,dz) := N(t,dz) - \nu(dz)t$ is the compensated Poisson measure.

More generally, we can define the stochastic process X_t, as:

$$dX_t = a(t)dt + \sigma(t)dB(t) + \int_{|z|<1} H(t,z)\tilde{N}(dt,dz) + \int_{|z|\geq 1} H(t,z)N(dt,dz), \tag{1.7}$$

known as Lévy - Itô processes. Moreover, by combining the compensator with the drift term the above can be written as:

$$dX_t = a(t)dt + \sigma(t)dB(t) + \int_{\mathbb{R}} H(t,z)N(dt,dz). \tag{1.8}$$

We will adopt this formulation throughout the remainder of this work. For such processes, we have the following results, which are extension of the standard, non-jump Itô-formula and generator operators.

Theorem 1.3.7 (Itô formula)**.** *Let $X_t \in \mathbb{R}$ be an Itô- Lévy process and consider a function $f(x,t)$, with $f \in C^2(\mathbb{R}\times[0,T])$. Then, the dynamics of the process $f(X_t,t)$ are given by the following version of the Itô formula:*

$$\begin{aligned} df(X_t,t) &= \frac{\partial f}{\partial t}(X_t,t)dt + \frac{\partial f}{\partial x}(X_t,t)\big(a(t)dt + \sigma(t)dB_t\big) + \frac{1}{2}\frac{\partial^2 f}{\partial x^2}(X_t,t)\sigma^2(t)dt \\ &+ \int_{\mathbb{R}} \big(f\big(X_{t-} + H(t,z),t\big) - f(X_{t-},t)\big)N(dt,dz) \end{aligned}$$

Definition 1.3.8 (Generator)**.** For a Lévy-Itô process, given by (1.8), and function $f : \mathbb{R}\times[0,T] \to \mathbb{R}$ we define the generator $\mathcal{L}$ by:

$$\mathcal{L}f(x,t) = \lim_{t\downarrow 0} \frac{\mathbb{E}[f(X_t,t)|X_0 = x] - f(x,t)}{t}. \tag{1.9}$$

It particular, it can be shown that the generator admits the following form:

$$\mathcal{L}f(x,t) = \frac{\partial f}{\partial t} + a(t)\frac{\partial f}{\partial x} + \frac{1}{2}\sigma^2(t)\frac{\partial^2 f}{\partial x^2} + \int_{\mathbb{R}} \big(f(x+z,t) - f(x,t)\big)\nu(dz), \tag{1.10}$$

1.3.1.1 The non-jump Ornstein-Uhlenbeck process

A specific Lévy process we will be studying in detail is the Ornstein-Uhlenbeck (OU) process, widely used in financial mathematics. In Chapter 3, we will use a Ornstein-Uhlenbeck process with a jump component, so we emphasize that the following is the non-jump, continuous version. In its simplest form, the OU process X_t is defined as the stochastic process satisfying the SDE:

$$dX_t = k(\theta - X_t)dt + \sigma dB_t, \; X_s = x, \tag{1.11}$$

for some known x, where, as above, B_t represents the standard Brownian motion and k, θ and σ are positive real constants. The OU process is a mean-reverting, Gaussian and Markov process, which is also temporally homogeneous. We can therefore equivalently write (1.11) as:

$$dX_u = k(\theta - X_u)du + \sigma dB_u, \; X_0 = x, \tag{1.12}$$

where $u = t - s$. For simplicity, we write X_t^x to indicate the OU process with $X_0 = x$. We adopt this convention for all stochastic processes in the remainder of this work. Employing Itô's formula we can obtain the solution to the above SDE:

$$X_t = xe^{-kt} + \theta(1 - e^{-kt}) + \sigma \int_0^t e^{-k(t-u)} dB_u,$$

from which is it easy to see that $X_t \sim N(\theta + (x - \theta)e^{-kt}, \sigma^2(1 - e^{-2kt})/2k)$. These properties are what make this particular family of processes widely used in many applications. We will also need the following regarding the transition density and hitting time for the OU process.

Theorem 1.3.9. *The transition density of the OU process, with initial condition $X_0 = x$, is given by:*

$$p(y, x, t) \equiv \mathbb{P}(X_t = y | X_0 = x) = \sqrt{\frac{k}{\pi\sigma^2(1 - e^{-2kt})}} \exp\Big(- \frac{k(y - \theta - (x - \theta)e^{-kt})^2}{\sigma^2(1 - e^{-2kt})} \Big), \tag{1.13}$$

for $x, y \geq 0$.

Furthermore, for the OU process as defined in (1.12), we define the corresponding survival probability distribution, given by $Q(x, t) := \mathbb{P}\Big(\inf_{r \leq t} G_r^x > 0 \Big)$. The distribution can be obtained via appropriate Volterra equations, for which we refer the reader to Lipton and Kaushansky, 2018.

Remark 1.3.10. An important result is the fact that the transition density (1.3.9) is uniformly continuous as a function of the initial position x. This result follows from the uniform continuity of the transition semigroup $P_t f(x) = \mathbb{E}[f(X_t^x)]$ for $f \in L^\infty(\mathbb{R})$ (i.e., essentially bounded and measurable functions), as established in Remark 2.2 of Priola and Zabczyk, 2009, and choosing $f(x) = \mathbb{1}_{\{x=y\}}$, for some fixed $y \geq 0$.

1.3.1.2 The Lévy-driven (jump) Ornstein-Uhlenbeck process

The Lévy-driven OU is given by:

$$dG_u = k(\theta - G_u)du + \sigma dB_u + \int_{\mathbb{R}} zN(du, dz), \quad G_0 = x. \tag{1.14}$$

The process has two sources of randomness: the continuous Brownian motion B_t and the discontinuous Lévy jump term L_t, determined by the Poisson random measure $N(\cdot, \cdot)$, defined by $N(t, U) = \sum_{0<s\leq t} \chi(\Delta L_s)$ for every Borel set $U \subset \mathbb{R}$, where $\Delta L_s = L_s - L_{s\_}$, that represents the number of jumps of size $\Delta L_s \in U$, which occur up to time t. It then follows (see Theorem 1.3.5 of the Introduction) that the jump term is a Compound Poisson process with arrival rate $\lambda = \nu(\mathbb{R})$ and jump distribution $f(dz) = \lambda^{-1}\nu(dz)$. Furthermore, this process is temporally homogeneous, as the the sum of two homogeneous processes (the continuous OU and Compound Poisson processes). For an in depth analysis of integrals with respect to Poisson measures and their properties see e.g., Kyprianou, 2006.

1.3.1.3 Infinitesimal generators and PDEs for the non-jump Ornstein - Uhlenbeck process

Finally, we recall the generator for the continuous OU process given by (1.12). This operator is related to the corresponding survival probability $Q(x,t) := \mathbb{P}\Big(\inf_{r\leq t} X_r^x > 0\Big)$, as well as the corresponding transition density $p(\cdot, x, t)$ via the equation:

$$\frac{\partial f}{\partial t}(x,t) = \mathcal{L}f(x,t), \tag{1.15}$$

where $\mathcal{L}$ represents the operator:

$$\mathcal{L}f(x,t) = k(\theta - x)\frac{\partial f}{\partial t}(x,t) + \frac{1}{2}\sigma^2\frac{\partial f^2}{\partial x^2}(x,t). \tag{1.16}$$

This is known as the Kolmogorov backward equation. In Chapter 3 we will be considering more complex models, such as regime switching and stochastic volatility OU processes. Under such models, analogous equations to (1.15) are produced, which are included in Appendix B.1. We will refer to these equations after presenting the models in Chapter 3.

Chapter 2

Markov chain lumpability and applications in credit risk

2.1 Background

As discussed, the implementation of IFRS 9 has lead to significant modelling challenges, which financial institutions now must address. One of the main considerations is loan classification. Specifically, under IFRS 9, it is now mandatory for financial institutions to classify loans into three distinct categories, known as the IFRS 9 Stages; Stage 1 loans are considered performing, Stage 2 contains loans which have displayed a significant increase in credit risk and Stage 3 contains all Non-Performing loans (NPLs), considered to have defaulted. Furthermore, under IFRS 9 all institutions must report credit exposures ratings in accordance to the Staging defined by the regulations. This create significant compatibility and reporting issues, since most institutions still have different loan classification and reporting standards, used for modelling.

It is standard in the industry to model the risk of a credit exposure as a discrete stochastic process, in the sense that, as time evolves, loans may migrate from one category to another. We will also abide by this standard, and also adopt the well-established assumption that the time-wise evolution of the state of a loan can be modelled by a Markov process. Hence, institutions now face the problem of finding an efficient way of aggregating existing classifications, corresponding to different state spaces of the internal Markov chains, into an IFRS 9 compliant state space, namely one comprised of the three IFRS 9 performances: Stage 1, Stage 2 and Stage 3, whilst preserving the defining Markov property. We therefore face three important challenges in the implementation of the new standards that will be addressed in this Chapter:

- Streamlining all detailed internal models into a format compatible with the IFRS 9 stage classification, while maitaining the Markov property.
- Estimating the total time a loan is considered to be Stage 2. This is due to the fact that financial institutions need to increase provisions for the loans categorized in Stage 2, by considering the expected total losses that will be incurred throughout the remaining lifetime of the loan, referred to as the Expected Lifetime Provisions (ECL).
- Modelling the expected lifetime for revolving credit facilities, for example credit cards. Such loans do not have a set maturity date and hence, if they was considered Stage 2, the institution needs to estimate the remaining lifetime, over which it must calculate the total provisions.

These last two quantities can be estimated using the fundamental matrix of the Markov chain on the fundamental state space consisting of the 3 states provisioned by IFRS 9, which in turn can calculated as an analytic function of the transition matrix. Therefore, we can see that the characterization of the loan classification through a Markov process is of paramount importance. However, the internal models used by companies contain different classifications, often using multiple categories for performing or non-performing loans, which are characterized by auxiliary quantities such as days past due, forborne flags etc. The loan classification and risk reporting, as well as data collection and evaluation procedures adopted by each institution are based on Markov models using these highly granular categories, producing a fine graining of the fundamental state space required by IFRS 9. For compliance reasons under IFRS 9, it is mandatory that these internal models are made compatible with the state space which consists of the three aforementioned Stages. This results in the need for a coarse graining procedure, which collapses the detailed state space of internal models to the fundamental state space of IFRS 9. However, for the reasons stated in the beginning of this paragraph, this coarse graining must be made while retaining the Markov property for the reduced process. We therefore need to find a method of reducing the state space of the corresponding Markov chains to the particular state space indicated by the IFRS 9, whilst preserving its important quantitative properties and, in particular, the Markov property. We address these problems using the so-called lumping of the Markov chain, which was first introduced in Kemeny and Snell, 1976, and whose properties and applications have been examined in Barr and Thomas, 1977; Buchholz, 1994; Gurvits and Ledoux, 2005; Thomas and Barr, 1977; Tian and Kannan, 2006. This method suggests a partition of the original state space into distinct subsets on which one can define a reduced Markov process, compatible with the original one. The conditions under which a Markov chain is lumpable, as well as the resulting lumped Markov matrix, are determined by a pair of row-stochastic matrices U, V, characterizing the partition of the original state space, through which we obtain a closed form representation of the lumpability condition.

A Markov process, whose transition matrix satisfies these conditions will be called an exactly lumpable Markov chain and the reduced process on the corresponding partition will remain a Markov process. As expected, exact lumpability is a very delicate property and often, Markov chains do not satisfy it for given partitions of the state space, provided by U and V. On the other hand, the need arises for obtaining approximations of Markov chains projected on a particular partition (whose transition matrices are probably incompatible with the full lumping condition for the given partition). In order to compromise these two competing goals, one can consider the problem of approximate lumpability, i.e., modifying the original Markov chain in such a way as not to disrupt important observable quantities of the chain, while at the same time choosing a pertubation so that a lumped approximation is feasible. This will be called the problem of approximate lumpability. The idea of allowing a pertubation of the original Markov process is acceptable if one considers that, in actual applications, the true transition matrix, which is unknown, is inferred by statistical techniques and therefore, is subject to uncertainty, which generates pertubations. As a result, the estimator of an exactly lumpable Markov chain may not enjoy the same property. Naive grouping of the states will result in the loss of the Markov property, which, for all reasons described above, is crucial in the analysis. We examine the problem of lumping Markov chains, as well as an appropriate approximate version under the framework of the Moore-Penrose inverse,

which provides equivalent characterizations of lumpability, as well as an algorithmic approach to approximate lumpability.

A way to solve the problem of approximate lumpability is to find the exactly lumpable matrix, which is closest to the original Markov chain transition matrix, with respect to a notion of an appropriate norm. For a wide range of ℓ^2-related norms, we consider this task as an ℓ^2 minimization problem, under constraints related to the stochasticity and lumpability of the resulting matrix. As the Moore-Penrose inverse provides a least norm solution to the equation $Ax = b$, we are able to employ this, in combination with an appropriate iterative scheme, to obtain a minimizer in the setting defined above, or in other words the closest lumpable approximation to a non-lumpable transition matrix. Moreover, by combining this methodology with results pertaining to exact lumpability, we are able to pose and solve certain variations of the concept of approximate lumpability, while considering analytic functions of the original transition matrix, which appear to be meaningful in several applications and in the description of various observable quantities of the process. We will employ such methods to address the modelling tasks under IFRS 9, posed above (for more work on IFRS 9 modelling, we refer the interested reader to Beerbaum and Ahmad, 2015; Berglund, 2016; Gornjak et al., 2017).

It is worth emphasizing that the need for lumping in the present situation does not arise from the need of reducing the dimensionality of Markov chains, but rather from the need of compatibility of the given Markov chain (representing the internal model of the firm) with a mandatory pre-prescribed Markov chain model required by the supervising authority (as in the case of the IFRS 9 classifications) or, more generally, the need for comparisons of credit (or other) ratings across different institutions or possibly even different countries, as seen in Hill, Brooks, and Faff, 2009. We explore such examples in this Chapter. Despite the fact that we do not focus on high-dimensional Markov chains in this work, it is worth noting that Markov chain lumpability can potentially have important applications in problems leading to higher-dimensional processes, such as in Takada and Sumita, 2011. The proposed methods could therefore still be applied for dimensionality reduction, but this is not the focus of the present paper. Problems where dimensionality reduction whilst preserving the fundamental properties of the processes might be of interest may be related to scorecard development, where internal ratings must be mapped to corresponding credit scores, and machine learning techniques (e.g., Li et al., 2020). The application of lumpable Markov chains in such cases may increase the robustness of the model, whilst also decreasing the risk of overfitting. Finally, even though motivated by credit risk modelling, the results presented are applicable to more general problems related to Markov chain modelling.

2.2 Lumpability of a Markov Chain

The notion of lumpability, introduced for the first time in Kemeny and Snell, 1976 is a generic method of reducing the dimension of a continuous or discrete-time Markov chain, whilst maintaining Markov property of the original chain. Restricting ourselves only to the discrete state spaces, we consider a Markov Chain $M = (S, P)$. Assume now that we wish to reduce the dimension of S by aggregating several states into new compound states, thus creating a partition of the original state space S'. One could try to calculate the total probability of reaching the new compound states, however in general, there is no guarantee that by doing so the resulting reduced stochastic process retains the Markov property in the new state space. Whether this

fundamental property is preserved or not depends on the original Markov process and the chosen partition. Under this consideration, we recall the definition of a *lumpable Markov chain.*

Definition 2.2.1 (Lumpable Markov chain)**.** Let $\{X_t : t \in \mathbb{N}\}$ and $M = (S, P)$ be a Markov chain and $S' = \{A_1, A_2, \dots, A_m\}$, where $m < n$, be a partition of $S = \{1, 2, \dots, n\}$. The chain M is called lumpable with respect to S' if, for any initial distribution, it holds that:

$$\mathbb{P}(X_t \in A_j | X_{t-1} \in A_{i_1}, \cdots, X_{t-k} \in A_{i_k}) = \mathbb{P}(X_t \in A_j | X_{t-1} \in A_{i_1}) \tag{2.1}$$

for any t, k, j and any $A_{i_1}, \cdots, A_{i_k} \in S'$, whenever these conditional probabilities are well-defined, i.e. these conditions occur with positive probability.

The above definition can equivalently be written as follows: Let $S' = \{A_1, ..., A_m\}$ be the partition of the original state space S. Then, the Markov Chain $M = (S, P)$ is lumpable if:

$$\sum_{k \in A_\eta} P_{i,k} = \sum_{k \in A_\eta} P_{j,k}, \text{ for all } i, j \in A_\xi \text{ and for all } (A_\eta, A_\xi). \tag{2.2}$$

These definitions imply that any Markov chain is lumpable if the lumped process maintains the Markov property, a fundamental qualitative property (see Kemeny and Snell, 1976 for counterexamples) and therefore, in the cases where the lumpability condition is met, we can consider the reduced Markov chain, with a lower dimensional state space S'.

Let us consider a Markov chain $M = (S, P)$ on a finite n-dimensional state space S, with corresponding transition matrix $P \in \mathbb{R}^{n \times n}$. We assume we want to study the lumpability of the chain M with respect to a known m-dimensional partition of the state space, S' where $m < n$. Therefore, $S' = \{A_1, A_2, \dots, A_m\} \simeq \{s'_1, s'_2, \dots, s'_m\}$, where by this notation we imply that we consider each element of the partitioned state space $A_i \subset S'$ as a new compound state of the aggregate Markov chain, defined on the state space S'. Therefore, it makes sense to define the function $\phi : \{1, 2, \dots, n\} \to \{1, 2, \dots, m\}$, which maps any element of S to the element of S' it will lumped to, according to the fixed partition. We can write, with an abuse of notation, that $S = \{A_1, A_2, \dots, A_m\} \simeq \{s'_1, s'_2, \dots, s'_m\} \simeq \{1, 2, \dots m\}$ and we have that $\phi(i) = s'_j$ when j is such that $i \in A_j$. We can now also define the cardinality of the aggregate state $s'_j \in S'$, as $|s'_j| = |\{i \in S : \phi(i) = s'_j\}|$.

In order to study the lumpability of M with respect to S', we need an efficient way of defining the aggregate states of the new Markov chain. Consider the two matrices $U \in \mathbb{R}^{m \times n}$ and $V \in \mathbb{R}^{n \times m}$, as follows:

$$(V)_{i,j} = \begin{cases} 1, & \text{if } \phi(i) = s'_j \\ 0, & \text{otherwise} \end{cases}$$

$$(U)_{i,j} = \begin{cases} 1/|s'_i|, & \text{if } \phi(j) = s'_i \\ 0, & \text{otherwise} \end{cases}$$

Therefore, an entry of V is 1 if the state indicated by its row is lumped to the aggregate state indicated by its column and 0 otherwise. For an entry U, if the state indicated by its column is lumped to the aggregate state indicated by its row (say s'_i) the entry is $1/|s'_i|$, and 0 otherwise.

Example 2.2.2. For example, let $S = \{1,2,3,4,5,6\}$ and $S' = \{\{1\},\{2,3\},\{4,5,6\}\} \simeq \{s'_1, s'_2, s'_3\}$ so that the matrices $U \in \mathbb{R}^{3\times 6}, V \in \mathbb{R}^{6\times 3}$ we obtain are:

$$V = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}, \quad U = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix}.$$

◁

These matrices can be used to define a partition S' of S. Before proceeding to the alternative definition of lumpability using matrices U and V, it is useful to examine the general structure of these matrices. We can write V and U as a collection of row and column vectors, respectively, as below:

$$V = \begin{pmatrix} \leftarrow & r_1 & \rightarrow \\ \leftarrow & r_2 & \rightarrow \\ & \vdots & \\ \leftarrow & r_n & \rightarrow \end{pmatrix}, \quad U = \begin{pmatrix} \uparrow & \uparrow & & \uparrow \\ c_1 & c_2 & \cdots & c_n \\ \downarrow & \downarrow & & \downarrow \end{pmatrix}, \tag{2.3}$$

where $r_i \in \mathbb{R}^{1\times m}$ and $c_i \in \mathbb{R}^{m\times 1}$. From the description above we can write the rows of V as $r_i = e^{S'}_{\phi(i)}$, where and $e^{S'}_k$ is an element of the standard basis with dimension m (the cardinality of S'), i.e., the m-dimensional row vector with 1 in the k^{th} column and 0's in all others. Similarly, we can write the columns of U as $c_i = \frac{1}{|\phi(i)|}\left(e^{S'}_{\phi(i)}\right)^T$, for $i = 1,2,\ldots,n$.

There seems to be a close connection between lumpability and the Moore-Penrose inverse. The following result provides an interesting characterization of the matrices V, U, which, to the best of our knowledge, has not been reported elsewhere.

Lemma 2.2.3. *For the matrices U, V presented above, it holds that $U^\dagger = V$.*

Proof. It is enough to show that the matrix V satisfies the four Penrose conditions. We can see that the matrix V satisfies the first three conditions of the Moore-Penrose inverse for U:

$$UVU = U, \quad VUV = V, \quad UV = (UV)^T = I_{m\times m}$$

Moreover, it is easy to see that the matrix VU is a projection matrix:

$$(VU)^2 = VUVU = VI_{m\times m}U = VU.$$

It remains to show that $(VU)^T = VU$. To do this, we will use the general form of the matrices U and V, in terms of their rows and columns, respectively, as defined above. The resulting matrix VU is of the form:

$$VU = \begin{pmatrix} r_1\cdot c_1 & r_1\cdot c_2 & \cdots & r_1\cdot c_n \\ r_2\cdot c_1 & r_2\cdot c_2 & \cdots & r_2\cdot c_n \\ \vdots & \vdots & \ddots & \vdots \\ r_n\cdot c_1 & r_n\cdot c_2 & \cdots & r_n\cdot c_n \end{pmatrix}. \tag{2.4}$$

We will now prove that this matrix is symmetric. Consider any two arbitrary states i, j. To analyse their contribution to the matrix VU we distinguish the following cases:

(*i*) $i \neq j$ and states i and j will not be lumped together, i.e. $\phi(i) \neq \phi(j)$. Hence, $r_i \cdot c_j = e^{S'}_{\phi(i)} \frac{1}{|\phi(j)|} \left(e^{S'}_{\phi(j)}\right)^T = 0 = r_j \cdot c_i$.

(*ii*) $i \neq j$ and states i and j will be lumped into the same aggregate state, i.e. $\phi(i) = \phi(j)$. Then, $r_i \cdot c_j = e^{S'}_{\phi(i)} \frac{1}{|\phi(j)|} \left(e^{(S')}_{\phi(j)}\right)^T = \frac{1}{|\phi(j)|} = e^{S'}_{\phi(j)} \frac{1}{|\phi(i)|} \left(e^{(S')}_{\phi(i)}\right)^T = r_j \cdot c_i$.

(*iii*) $i = j$ and trivially, $r_i \cdot c_i = e^{S'}_{\phi(i)} \frac{1}{|\phi(i)|} \left(e^{S'}_{\phi(i)}\right)^T = \frac{1}{|\phi(i)|} = \frac{1}{|\phi(j)|} = r_j \cdot c_j$. Of course, if state i is not lumped into an aggregate state of S' then $|\phi(i)| = 1$ so we will observe 1's in the diagonal for such states.

We have thus shown that $(VU)_{ij} = (VU)_{ji}$, for all i, j and we conclude that VU is a symmetric matrix, proving the last Penrose equation. Therefore, $V^\dagger = U$ and vice versa. Our calculations provide a clear view of the general form of the matrix VU:

$$VU = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & 0 & S_1 & 0 & 0 & 0 \\ \vdots & \vdots & 0 & S_k & 0 & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{pmatrix}, \tag{2.5}$$

where S_i are the symmetric matrices that are created from the multiplication of the rows and columns in U and V, corresponding to the lumped states. □

We can return to reformulating the definition of lumpability, using the matrices U, V.

Definition 2.2.4. We say that the transition matrix P is $U - V$-lumpable if

$$VUPV = PV. \tag{2.6}$$

In the case where (2.6) holds, the reduced stochastic process on S' retains the Markov property and the matrix $P^* := UPV$, which is stochastic, is called the transition matrix of the lumped system.

Condition (2.6) guarantees that if we define a new stochastic process on an m-dimensional state space S' rather than the original n-dimensional state space S, with the state space S' consisting of a partition of S, created by lumping together the states indicated by the 1's in the matrix V, the new process is a Markov process.

Moreover, since the matrix VU is a projection matrix, we have that $VU = VV^\dagger = \Pi_V$ and so, condition (2.6) implies that $\mathcal{R}(PV)$ is invariant under the orthogonal projection VU, giving us the following relation between the ranges:

$$\mathcal{R}(PV) \subseteq \mathcal{R}(VU) = \mathcal{R}(V) \tag{2.7}$$

Below we provide an example of a lumpable Markov chain and the matrices U and V corresponding to a possible partition.

Example 2.2.5. As an example of a lumpable system, consider the Markov chain $\{X_t : t \in \mathbb{N}\}$ on the four-dimensional state space $S = \{1,2,3,4\}$ with transition matrix

$$P = \begin{pmatrix} 0.225 & 0.325 & 0.175 & 0.275 \\ 0.175 & 0.375 & 0.125 & 0.325 \\ 0.420 & 0.000 & 0.240 & 0.340 \\ 0.060 & 0.360 & 0.340 & 0.240 \end{pmatrix}. \tag{2.8}$$

Consider the reduced state space $S' = \{\{1,2\},\{3,4\}\} \simeq \{s'_1, s'_2\}$, which is 2 dimensional, and aggregates states 1, 2 and 3, 4 of the original system as two compound states. Matrices U and V corresponding to this partition are given by:

$$U = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}, \quad V = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}. \tag{2.9}$$

One can easily check that condition (2.6) is satisfied for this choice of P, U and V, which implies that one can consider the original Markov chain as restricted on the new state space S', with the resulting transition matrix $P^* := UPV$:

$$P^* = \begin{pmatrix} 0.55 & 0.45 \\ 0.42 & 0.58 \end{pmatrix}.$$

To illustrate the frailty of the lumpability condition, note that if we perturb P to the matrix:

$$P^\epsilon = \begin{pmatrix} 0.215 & 0.335 & 0.175 & 0.275 \\ 0.175 & 0.375 & 0.125 & 0.325 \\ 0.420 & 0.000 & 0.240 & 0.340 \\ 0.060 & 0.360 & 0.340 & 0.240 \end{pmatrix},$$

the Markov process is no longer lumpable, as $VUP^\epsilon V \neq P^\epsilon V$.

◁

Furthermore, it is known that the $k-$th step transition matrix of a lumpable chain is also lumpable. The following lemma is due to Tian and Kannan, 2006.

Lemma 2.2.6. *If P is a stochastic matrix that is $U - V$ lumpable, then P^k, is also $U - V$ lumpable, for $k \in \mathbb{Z}$.*

This result shows that the k-step transition matrices of the Markov Chain are also lumpable. Moreover, the lumped k-step transition matrix is simply the k^{th} power of the original lumped matrix P. Indeed, one can easily check that $(P^*)^2 = (UPV)^2 = (UP)(VUPV) = UP(PV) = UP^2V = (P^2)^*$.

2.3 Approximate lumpability

Exact lumpability, in the sense previously defined, may not hold for a given Markov chain and moreover, is a rather delicate propery, as illustrated in Example 2.2.5. In the need to aggregate the original state space to a required form (e.g. aggregating an institution's internal rating to an IFRS 9-compliant model), one can consider a

perturbation of the original Markov chain in such a way so that the resulting process is lumpable (hence leading to a Markov process for the reduced system), while still being a reasonable approximation, in terms of observables, for the original process. We will formulate this problem in terms of an appropriate constrained optimization problem. However, we first introduce the following notation, which will be used throughout the section:

Notation 2.3.1. We denote by $\mathbf{1_n}$ the $n-$dimensional column vector, whose entries are all 1 and by the $\mathbf{0_n}$ the $n-$dimensional column vector, whose entries are all 0.

Consider a Markov chain with a transition matrix $P \in \mathbb{R}^{n\times n}$, which is *not* exactly lumpable with respect to a fixed given partition S', with dimension $m < n$, defined by the corresponding matrices V, U. Our goal, is to find a new Markov chain, which has the same dimension as the original one, with a transition matrix $P_L \in \mathbb{R}^{n\times n}$ as close as possible to the original chain under some norm related to a given observable, so that the new chain P_L is exactly lumpable with respect to S'. Under this assumption, we may apply the exact lumpability techniques and approximate the quantitative properties of the original transition matrix P, with the lumpable chain with transition matrix P_L. Since there is no free lunch, this procedure has an approximation error, which can be minimized under an appropriate norm of the difference $P_L - P$.

Based on these facts, the problem can be formulated as the following minimization under constraints:

$$\begin{aligned} \min_{P_L\in\mathbb{R}^{n\times n}} \quad & \|P_L - P\|_2 \\ \text{subject to} \quad & VUP_LV = P_LV, \quad P_L\mathbf{1}_n = \mathbf{1}_n, \quad (P)_{i,j} \geq 0, \end{aligned} \tag{2.10}$$

where $\|\cdot\|_2$ denotes the ℓ^2 norm (note that the choice of the ℓ^2 norm is indicative, as one may consider different weighted ℓ^2 norms). The second and third conditions of (2.10) correspond to the requirement that P_L is a row stochastic matrix, a necessary condition for P_L to be interpreted as a transition matrix.

An interesting observation comes from the fact that due to the Cauchy-Schwarz inequality and since the minimization described in Problem (2.10) stands with respect to ℓ^2 norm, we can obtain estimates of solution using ℓ^1 norm. In addition, (2.10) is a classical quadratic optimization problem, a family of well studied problems.

We reformulate the problem in vectorized form, using the Kronecker product. In the lumpability condition, the left hand side becomes $(VU \otimes V^T)p_L$ and the right hand side $(I_{n\times n} \otimes V^T)p_L$. Using the Kronecker product properties, the condition can be expressed as:

$$\Big((VU - I_{n\times n}) \otimes V^T\Big)p_L = \mathbf{0}_{mn},$$

where $p_L = vec(P_L)$. Similarly, noting that $\mathbf{1}$ is already a column vector, the unit row-sum condition can be written as:

$$(I_{n\times n} \otimes \mathbf{1}_n^T)p_L = \mathbf{1}_n.$$

These calculations motivate the following definition.

Definition 2.3.2 (Lumpability condition matrix)**.** Consider the problem of approximate lumpability (2.10). We define the lumpability condition matrix associated to

this problem as the matrix $A \in \mathbb{R}^{(mn+n)\times n^2}$, given by the following block form:

$$A = \begin{pmatrix} (VU - I_n) \otimes V^T \\ I_{n\times n} \otimes \mathbf{1}_n^T \end{pmatrix}.$$

Using the condition matrix we can now reformulate Problem (2.10) as:

$$\begin{aligned} \min_{P_L \in \mathbb{R}^{n\times n}} \quad & \|p_L - p\|_2 \\ \text{subject to} \quad & Ap_L = b, \quad (P)_{i,j} \geq 0, \end{aligned} \tag{2.11}$$

where the column vector b is given by $b = [\mathbf{0}_{mn}\ \mathbf{1}_n]^T$.

When using this form of A, we obtain duplicate, as well as $\mathbf{0}$ rows in the condition matrix, which, ideally, should be eliminated. However, this is only one way of calculating A. In Section 2.4 we propose a more tailor made algorithm that can be used to obtain an equivalent lumpability condition matrix, where the unecessary rows have been removed. This algorithm will be of particular use in the applications related to credit risk, where we make use of the fact that only consucutive ratings (i.e. states) will be lumped together. This is, of course, the case under the classification required by the IFRS 9, as well. Stage 2 contains all loans which have exhibited a significant increase in credit risk. When using the transition matrix representation of the migration between credit ratings, it is standard practice to write all riskier assets consecutively. For example, BBB, BBB-, CCC+, CCC would be consecutive states in the transition matrix and we would want to aggregate these four ratings to Stage 2, when applying the lumping for IFRS 9 compliance.

The reformulation of the problem as a simple vector equation will allow us to solve part of Problem (2.11) without the non-negativity contraint, using the Moore-Penrose inverse. However, in general, the Moore-Penrose solution will not satisfy the this additional contraint, which is essential in order for P_L to be interpreted as a transition matrix. Hence, in order to take into account both constraints, we consider the two conditions we have separately; $Ap_L = b$ captures the lumpability and unit row-sum condition for P_L and $P_{i,j} \geq 0$ captures the non-negative condition that must be satisfied for P_L to be an acceptable transition matrix. It makes sense now to define the following:

(a) The set of all $U - V$ lumpable, $n-$dimensional square matrices with unit row-sums, denoted by $\mathcal{L}(U, V)$ (in subsequent results, we omit the explicit dependence on U and V to simplify notation)

(b) The set of all non-negative $n-$dimensional square matrices, denoted by $\mathcal{M}_n^+$.

Therefore, the solution to our problem will be the orthogonal projection of P onto the intersection $\mathcal{L} \cap \mathcal{M}_n^+$. To calculate this, we employ an alternating projection technique (see e.g. Escalante and Raydan, 2011) and in particular, Dykstra's algorithm (presented in full in the following Theorem). The splitting of the projections is particularly convenient, as, seperately, they can be calculated very easily.

Theorem 2.3.3. *For a non exactly lumpable transition matrix $P \notin \mathcal{L} \cap \mathcal{M}_n^+$, the solution to Problem (2.11) is given by $\hat{P}_L = \lim_{k\to\infty} P_k$, where P_k is the k-th iteration of Dykstra's*

iterative scheme (i.e. relabelling $p_k = vec(P_k)$ back into matrix form):

$$\begin{aligned} y_k &= \Pi_{\mathcal{S}}(x_k + p_k) \\ x_{k+1} &= p_k + x_k - y_k \\ p_{k+1} &= \Pi_{\mathcal{L}}(y_k + q_k) = A^{\dagger}\Big(b - A(y_k + q_k)\Big) + (y_k + q_k) \\ q_{k+1} &= y_k + q_k - p_{k+1}, \end{aligned} \tag{2.12}$$

with $x_0 = q_0 = 0$ and $p_0 = p$. If, at any iteration, $P_k \in \mathcal{L} \cap \mathcal{M}_n^+$, the procedure stops and we obtain $\hat{P}_L = P_k$.

Proof. Firstly, we recognize that both $\mathcal{L}$ and $\mathcal{M}_n^+$ are closed and convex sets, under the standard topology of $\mathbb{R}^{n^2}$ and therefore, the algorithm converges. To apply Dykstra's algorithm to reach a solution, we need to know how to project onto each of the sets $\mathcal{L}$ and $\mathcal{M}_n^+$. To project onto the set of non-negative matrices, we set all negative values that occur from the projection onto the lumpable matrices equal to zero. It remains to find the orthogonal projection onto the set of lumpable matrices with unit row-sums. We recognize that this is equivalent to finding the solution to the simplified approximate lumpability problem:

$$\begin{aligned} \min_{P_L \in \mathbb{R}^{n \times n}} \quad & \|p_L - p\|_2 \\ \text{subject to} \quad & Ap_L = b. \end{aligned} \tag{2.13}$$

Since the only constraint is a vector equation, this minimization problem can be solved using the Moore-Penrose inverse. We aim to minimize the norm of the quantity $e_L := p_L - p$. Knowing that the Moore-Penrose inverse gives a minimal norm solution, we rearrange the problem to get:

$$Ap_L = b \Rightarrow A(e_L + p) = b \Rightarrow Ae_L = b - Ap \Rightarrow e_L = A^{\dagger}(b - Ap).$$

By definition, the solution will be that of minimal norm of $\|e_L\| = \|p_L - p\|$, as required and we can solve for p_L to obtain $p_L = A^{\dagger}(b - Ap) + p$. But, as the solution to the minimization problem, P_L is the orthogonal projection of P onto $\mathcal{L}$, hence $\Pi_{\mathcal{L}}(p) = A^{\dagger}(b - Ap) + p$. We can recover the matrix form of P_L from p_L, trivially. Hence, we now also have a closed form expression for the orthogonal projection onto set $\mathcal{L}$. We substitute this into the third step of Dykstra's algorithm and obtain (2.12).

Further details on Dykstra' algorithm and convergence can be found in Bauschke and Borwein, 1994; Bauschke et al., 2011. □

Remark 2.3.4. We should note that Dykstra's algorithm is only necessary when the first iteration $p_L = A^{\dagger}(b - Ap) + p$ returns one or more negative entries. We apply the algorithm to converge to the correct solution in these cases. If the first p_L, however, satisfies all the conditions there is no need for alternating projection schemes.

Theorem 2.3.3 provides the solution to the approximate lumpability problem where we consider the error as the difference between the lumpable and non-lumpable versions of the transition matrix. This provides the best possible Markovian approximation and the quantity $\|P_L - P\|$, is the least error we have to undertake if we need to conform to the reduced model. Whether this error is acceptable depends on the application and the expert opinion of the practitioner.

In many cases, it may be meaningful to consider the error of observable quantities that arises due to the use of the exactly lumpable approximation P_L. Therefore,

we wish to examine a larger family of minimization problems, where the error can be considered as the difference between observable processes, when replacing the original Markov chain with the the lumpable approximation. These observables correspond to functions of the transition matrix. To build these ideas we will need the following results, which are applicable when P is invertible. In such cases, we may consider the lumpability of functions depending on the inverse of the transition matrix, as for example, the fundamental matrix, which will be used for the IFRS 9 modelling (and must therefore be in terms of the three IFRS 9 Stages). Of course, as we have already seen, invertibility of P is not a necessary condition for any of the results up to this point. However, when applicable, this condition allows us to solve an extended family of minimization problems, which is much wider than the quadratic minimization problems considered above.

The extension of Theorem 2.3.3 to the approximation of observables of the corresponding Markov chains requires some auxiliary observations, that we prepare as seperate Lemmata below.

Lemma 2.3.5 (CB lemma)**.** *Assume that P is an invertible stochastic transition matrix which is $U-V$ lumpable (in the sense of Definition 2.2.4). Then,*

$$VUP^{-1}V = P^{-1}V.$$

Proof. The proof consists of 5 steps.

1. We note that if P is $U-V$ lumpable then $(UP^{-1}V)(UPV) = I_{m\times m}$, i.e. $UP^{-1}V$ is a right inverse of the lumped transition matrix $P^* = UPV$.

Indeed, by multiplying (2.6) by U from the right and by P^{-1} from the left we obtain that $P^{-1}VUPVU = VU$, and multiplying by U from the left, and V from the right leads to

$$(UP^{-1})V)(UPV)(UV) = (UV)(UV)$$

which upon recalling that $UV = I_{m\times m}$ yields the required result.

2. We have from (2.7) that $\mathcal{R}(PV) \subset \mathcal{R}(V)$.

3. Since P is invertible, $\mathcal{R}(PV) = P\mathcal{R}(V) \subset \mathcal{R}(V)$ (from step 2) which leads to $\mathcal{R}(V) \subset P^{-1}\mathcal{R}(V)$.

4. We claim that $\mathcal{R}(V) \subset \mathcal{R}(PV)$.

To prove the claim it is enough to prove that for every $x \in \mathbb{R}^n$ there exists $x' \in \mathbb{R}^n$ such that, $VxPVx'$. We must identify such an x'. This requires solving the equation $Vx = PVx'$ in terms of x'. Multiplying from the left by U, and recalling that $UV = I_{m\times m}$ this reduces to $x = UPVx'$, and multiplying from the left by $UP^{-1}V$, and using the result of step 1, we have that $x' = UP^{-1}Vx$. Hence the claim is proved.

5. From steps 3 and 4, $\mathcal{R}(V) = \mathcal{R}(PV)$ and by the invertibility of P this yields that $P^{-1}\mathcal{R}(V) = \mathcal{R}(V)$, which is the required condition for P^{-1}. □

Note that, even though P^{-1} is not necessarily a transition matrix, one can still say, abusing notation, that P^{-1} is $U-V$ lumpable. The resulting lumped matrix $(P^{-1})^* = UP^{-1}V$ cannot necessarily be interpreted as a lower-dimensional version of the process, as it is no longer a stochastic matrix. However, its role is very important in the calculation of analytic functions of P.

Moreover, we obtain a relation between the lumped inverse and the inverse of the lumped matrix, in the same fashion as the k-th step transition matrices above, as the following lemma indicates:

Lemma 2.3.6. *Assume P is $U-V$ lumpable with lumped and invertible transition matrix P^*. Then, it holds that $(P^*)^{-1} = (P^{-1})^*$*

Proof. From Step 1 in Lemma 2.3.5, we have already shown that $(UP^{-1}V)(UPV) = I_{m\times m}$. We must also show that $(UPV)(UP^{-1}V) = I_{m\times m}$. Starting now with the lumpability condition for P^{-1} multiplying by U from the right and by P from the left we obtain $PVUP^{-1}PVU = VU$, and multiplying by U from the left, and V from the right leads to

$$(UPV)(UP^{-1}V)(UV) = (UV)(UV) \implies (UPV)(UP^{-1}V) = I_{m\times m}. \tag{2.14}$$

By definition, $P^* = UPV$ and $(P^{-1})^* = UP^{-1}V$, which completes the proof. □

Due to linearity of matrix addition, we can easily deduce that the sum of lumpable matrices is also a lumpable matrix. In view of the previous lemmata, we can then conclude that analytic function of a lumpable transition matrix P, of the form $f(P) = \sum_{k=0}^{\infty} a_k P^k, a_k \in \mathbb{R}$, is also lumpable. As a result, we may consider problems where we wish to minimize the ℓ^2 distance between matrices depending on P. As stated above, these functions of P can be interpreted as observables, based on the dynamics of the Markov chain. Hence, we wish to examine how we can solve the problem of approximate lumpability, whilst minimizing the error in these important quantities. This method will prove useful in subsequent applications under the IFRS 9 framework, where we will need to calculate the fundamental matrix, under the dynamics of the lumped process. The following proposition outlines the methodology to solve such minimization problems.

Proposition 2.3.7.

Consider the partition, defined by U, V, fixed and an invertible transition matrix $P \notin \mathcal{L}$. For any analytic and invertible function $f(\cdot) : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$, of the form $f(P) = \sum_{k=0}^{\infty} a_k P^k, a_k \in \mathbb{R}$, the approximate lumpability problem

$$\begin{aligned} \min_{P_L \in \mathbb{R}^{n\times n}} \quad & \|f(P_L) - f(P)\|_2 \\ \text{subject to} \quad & VUP_LV = P_LV, \quad P_L\mathbf{1}_n = \mathbf{1}_n, \quad (P)_{i,j} \geq 0 \end{aligned} \tag{2.15}$$

admits the solution $\hat{P}_L = f^{-1}(\lim_{k\to\infty} P_k)$, where P_k is the $k-th$ iteration of Dykstra's algorithm (i.e. relabelling $p_k = vec(P_k)$ back into matrix form):

$$\begin{aligned} y_k &= \Pi_{\mathcal{S}}(x_k + p_k) \\ x_{k+1} &= p_k + x_k - y_k \\ p_{k+1} &= \Pi_{\mathcal{L}}(y_k + q_k) = f^{-1}\Big(A'^{\dagger}\big(b - A'f(y_k + q_k)\big) + f(y_k + q_k)\Big) \\ q_{k+1} &= y_k + q_k - p_{k+1}, \end{aligned} \tag{2.16}$$

with $x_0 = q_0 = 0$ and $p_0 = p$ and for appropriate lumpability condition matrix A' and vector b'. If, at any iteration, $P_k \in \mathcal{L} \cap \mathcal{M}_n^+$, the procedure stops and we obtain $\hat{P}_L = P_k$.

Proof. Given the properties proven for a lumpable matrix P, we can deduce that $f(P)$ will also be lumpable. Therefore, $VUf(P)V = f(P)V$. Moreover, the unit row-sum condition $P_L\mathbf{1}_n = \mathbf{1}_n$ can be transformed into an equation corresponding to the row-sums of the resulting matrix $f(P_L)$, i.e., we can rewrite the condition in the form $f(P_L)X = Y$ for some appropriate n-dimensional column vectors X, Y. Hence, we can write the problem in terms of $f(P_L)$ as:

$$\begin{aligned} \min_{P_L \in \mathbb{R}^{n\times n}} \quad & \|f(P_L) - f(P)\|_2 \\ \text{subject to} \quad & VUf(P_L)V = f(P_L)V, \quad f(P_L)X = Y, \quad (P_L)_{i,j} \geq 0. \end{aligned} \tag{2.17}$$

Using the Kronecker product we can again write the two first conditions as a vector equation. With the new row-sum condition, the second part of the condition matrix will become:

$$f(P_L)X = Y \Rightarrow (I_{n\times n} \otimes X^T) f(p_L) = Y,$$

where we need not write y for the vectorized form of Y, as it is already a column vector. We can now define the lumpability condition matrix, A' associated to this minimization problem by:

$$A' = \begin{pmatrix} (VU - I_{n\times n}) \otimes V^T \\ I_{n\times n} \otimes X^T \end{pmatrix},$$

as well as the vector $b' = [\mathbf{0}_{mn} \;\; Y]^T$.

This allows us to write a simplified version of the problem (as in (2.11), omitting the non-negative condition):

$$\begin{aligned} \min_{P_L \in \mathbb{R}^{n\times n}} \quad & \|f(p_L) - f(p)\|_2 \\ \text{subject to} \quad & A' f(p_L) = b', \end{aligned} \tag{2.18}$$

where by $f(p_L)$ we mean the vectorized form of the entries of the matrix $f(P_L)$. As in Theorem 2.3.3 we can define $e_L := f(p_L) - f(p)$, to obtain:

$$A' f(p_L) = b' \Rightarrow A'(e_L + f(p)) = b' \Rightarrow e_L = A'^{\dagger}(b' - A' f(p)).$$

Hence, $f(p_L) = A'^{\dagger}\big(b' - A' f(p)\big) + f(p)$ and $p_L = f^{-1}\Big(A'^{\dagger}\big(b' - A' f(p)\big) + f(p)\Big)$ is the transition matrix that satisfies (2.18). This expression is used to find projection of P onto the set $\mathcal{L}$, such that $\|f(P_L) - f(P)\|_2$ is minimized. Hence, we use this expression for the third step of Dykstra's algorithm, obtaining (2.16), which will converge to the solution belonging to $\mathcal{L} \cap \mathcal{M}_n^+$, □

Remark 2.3.8.

(*a*) Results (2.12) and (2.16) can also be written as $p_L = A^{\dagger}b + P_{\mathcal{N}(A)}p$ and $p_L = f^{-1}\big((A')^{\dagger}b' + P_{\mathcal{N}(A')}f(p)\big)$. Therefore, the solutions can also be understood in terms of the minimal norm solution $A^{\dagger}b$ plus a component which arises from the projection of the original transition matrix onto the null space of A.

(*b*) The case of minimizing under weighted norms is captured by Proposition 2.3.7. We have also assumed that f is an invertible function, which is a logical assumption when considering applications in IFRS 9, as well as in other fields. This condition is necessary when we want to also retrieve the lumpable transition matrix P_L, when solving Problem (2.15). In other cases, where we only be interested in the lumpable version of $f(P)$, the invertibility condition may be relaxed.

(*c*) The error induced by replacing the exact transition matrix with the closest lumpable version for more general observables, e.g. of the form $f(P) = EP^k, k \in \mathbb{N}$, where E is a matrix of appropriate dimension (found e.g., in pricing applications), can be estimated by standard inequalities pertaining to matrix norms or triangle inequalities.

Moreover, when studying applications of Markov chains, transition matrices are often produced through statistical methods, which incur an error. The following

proposition establishes how this error is carried through when solving the problem of approximate lumpability, a result which can be used for statistical analysis of observables. For more work on the analysis of transition probability errors we refer the interested reader to e.g., Kim and So, 2008.

Proposition 2.3.9. *Let p and $\tilde{p}$ be the vector forms of the true transition matrix and the approximate matrix, respectively, such that $p = \tilde{p} + \epsilon$, where ϵ is a n^2-dimensional random variable, with mutlivaritate normal distribution, $\epsilon \sim \mathcal{N}_{n^2}(\mathbf{0}, \Sigma)$, where $\mathbf{0}$ is the n^2-dimensional 0 vector and Σ is an $n^2 \times n^2$ covariance matrix. Then, it holds that*

$$p_L - \tilde{p_L} \sim \mathcal{N}_{n^2}\big(0, (I_{n^2\times n^2} - A^\dagger A)^T \Sigma (I_{n^2\times n^2} - A^\dagger A)\big)$$

Proof. By Theorem 2.3.3 we know that $p_L = A^\dagger(b - Ap) + p$, where we have written the matrices in vector form, as above. Then by replacing we have

$$\begin{aligned} p_L &= A^\dagger\big(b - A(\tilde{p} + \epsilon)\big) + \tilde{p} + \epsilon \\ &= A^\dagger\big(b - A\tilde{p}\big) - A^\dagger A\epsilon + \tilde{p} + \epsilon \\ &= A^\dagger\big(b - A\tilde{p}\big) + \tilde{p} + \epsilon\big(I_{n^2\times n^2} - A^\dagger A\big) \\ &= \tilde{p_L} + \epsilon\big(I_{n^2\times n^2} - A^\dagger A\big), \end{aligned}$$

where the last equality arises from the lumped version of P. Hence, we conclude that

$$p_L - \tilde{p_L} \sim \mathcal{N}_{n^2}\big(0, (I_{n^2\times n^2} - A^\dagger A)^T \Sigma (I_{n^2\times n^2} - A^\dagger A)\big),$$

which completes the proof. □

2.4 Construction of the lumpability condition matrix A

In this section we outline an alternative algorithm to construct the lumpability condition matrix A (hereinafter simply referred to as the condition matrix), for an arbitrary approximate lumpability problem. One can use the form of A given in Definition 2.3.2, however, this returns duplicate and zero rows in the condition matrix. The proposed algorithm eliminates these rows, and we therefore obtain a lower dimensional matrix, making the problem well-posed.

We now proceed to the construction of A. Initially, we need to calculate the dimensions of A. We know that, due to the relabelling of the problem, A will have n^2 columns (recall that n is the size of the original state space of the Markov chain). The row dimension will depend on the number of equations that are necessary to ensure lumpability.

Lemma 2.4.1. *Consider lumping the Markov chain $M = (S, P)$, where $|S| = n$ onto a partition S', where $|S'| = m$. The number of equations needed to ensure lumpability from S to S' is $m(n - m)$.*

Proof. Consider an arbitrary state in the partition S', $A_\xi = \{i_1, i_2, \ldots, i_{|A_\xi|}\}$, with cardinality $|A_\xi|$. Then, according to Definition (2.2) of lumpability we obtain the equations

$$\sum_{k\in A_\eta} P_{i_1,k} = \sum_{k\in A_\eta} P_{i_2,k} = \cdots = \sum_{k\in A_\eta} P_{i_{|A_\xi|},k}, \text{ for all } A_\eta.$$

For this relation to hold we need $|A_\xi| - 1$ equations. However, we obtain such a system of equations for all pairs (A_η, A_ξ). Therefore, in total, for lumpability to

hold, the number of equations required is:

$$\sum_{i=1}^{m}\sum_{j=1}^{m}\left(|A_j|-1\right)=\sum_{i=1}^{m}\left(\sum_{j=1}^{m}|A_j|-m\right)=m(|S|-m)=m(n-m).$$

□

Hence, from the result above, along with the n equations for the row-sums of the lumpable matrix, we deduce that A will be a $\left(m(n-m)+n\right)\times n^2$ matrix (compared to the $(mn+n)\times n^2$ dimensional version obtained using the Kronecker product formulation).

For the proposed method that follows, we consider a partition in which lumped states contain only consecutive states of the original chain, for example $S'=\{\{1,2,3\},\{4,5\},\{6\},\{7,8,9,10\}\}\simeq\{s'_1,s'_2,s'_3,s'_4\}$. Without loss of generality, all partitions can be written in such a form, by simply multiplying the original Markov transition matrix with a sequence of permutation matrices. Moreover, as mentioned, this is a fair assumption since, when applying Markov models to credit risk, the ratings (i.e., states) are ordered from best to worst. Hence, when lumping these states to create aggregate ratings, it will only be meaningful to aggregate the consecutive ratings.

Definition 2.2 of lumpability can be written as $\sum_{k\in A_\eta}P_{i,k}-P_{j,k}=0$, for all $i,j\in A_\xi$. From this form we see that, for a fixed column k, the coefficients of the probabilities $P_{i,k}$ will be either 1 or -1, for all i. Given that the lumping occurs in consecutive order, we can begin by considering the first and second rows, for all columns $k\in A_\eta$. This motivates the following definition.

Definition 2.4.2 (Partition matrix)**.** Consider a partition S', with cardinality $|S'|=m$ of an original state space S, with $|S|=n$, in consecutive order, meaning that all states to be lumped together, into a single aggregate state, are of the form $i,i+1,\ldots,i+k$, for appropriate i and k. Then, for all $A_\eta\in S'$ define the *partition matrix* $L_\eta=(L_\eta)_{i,j}\in\mathbb{R}^{n\times n}$ by:

$$(L_\eta)_{i,j}=\begin{cases}1 & \text{if } i=1, j\in A_\eta\\ -1 & \text{if } i=2, j\in A_\eta\\ 0 & \text{otherwise}\end{cases}$$

thus obtaining m total partition matrices $L_1,L_2,\ldots,L_m$, characterizing the partition.

In order to proceed with each partition matrix $L_i, i=1,2,\ldots,m$, we define a family of $n-1$ related matrices $L_i^{(k)}, k=1,2,\ldots,n-1$, where $L_i^{(1)}$ is the original matrix L_i. Then, $L_i^{(k)}$ is the matrix where the block of the first two lines has shifted $k-1$ positions downward with the empty positions filled by the **0** row. Therefore, we will obtain:

$$(L_\eta^{(k)})_{i,j}=\begin{cases}1 & \text{if } i=k, j\in A_\eta\\ -1 & \text{if } i=k+1, j\in A_\eta\\ 0 & \text{otherwise,}\end{cases}$$

for $k=1,2,\ldots,n-1$. This procedure repeats until the block reaches the final row.

Writing the definition of lumpability in the form $\sum_{k\in A_\eta}P_{i,k}-P_{j,k}=0$, for all $i,j\in A_\xi$, we can see that these equations can be written as $l_\eta^{(k)}\cdot p=0$, where $l_\eta^{(k)}=vec(L_\eta^{(k)})$ and $p=vec(P)$. This holds since, for a fixed A_η, with corresponding partition matrix L_η, we obtain the equation for all pairs $i,j\in A_\xi$. Hence, each time

we shift and obtain $L_\eta^{(k)}$, we account for a new pair of states in $A_{\tilde{\xi}}$ (since only consequtive states are lumped together). Of course, for each $L_\eta^{(k)}$, we must check that the rows k and $k+1$ belong to the same aggregate state $A_{\tilde{\xi}}$ (if they do not, then we would not obtain this equation for lumpability). We do this until we have shifted down all the permutation matrices $n-1$ times. This algorithm will result in the first $m(n-m)$ rows of A.

We still have to consider the final n rows, which correspond to the necessary row sum conditions of a lumpable matrix. Using a similar approach to the above, we define the family of *row condition matrices* $C^{(k)} = (C^{(k)})_{i,j} \in \mathbb{R}^{n\times n}$ by:

$$(C^{(k)})_{i,j} = \begin{cases} 1 & \text{if } i = k, \text{ and for all } j, \\ 0 & \text{otherwise.} \end{cases}$$

We easily see that each $C^{(k)}$ is simply $C^{(k-1)}$, where the row of 1's have been shifted down once, and the remaining rows are all 0's (in the same manner as we shift the matrices $L_\eta^{(k)}$). After relabelling the row condition matrices into their corresponding vector forms, $c^{(k)} = vec(C^{(k)}) \in \mathbb{R}^{n^2}$, with $c^{(k)}_{\psi(i,j)} = C^{(k)}_{i,j}$, for $i,j,k = 1,2,\ldots,n$, we can see that $c^{(k)} \cdot p = \sum_{j=1}^{n} P_{k,j} = 1$, for $k = 1,2,\ldots,n$, resulting in the unit row-sum condition for a lumpable matrix. With the row condition matrices, we can complete the construction of A by filling in the remaining n rows with the vectors $c^{(k)}$, with $k \in \{1,2,\ldots,n\}$.

Finally, we also need to construct the column vector $b \in \mathbb{R}^{m(n-m)+n}$. From the definition of lumpability used for the construction of A, we can easily see that the first $m(n-m)$ entries of b will be 0's and the remaining n entries will be 1's (similarly defined in Theorem 2.3.3).

It is important to note that the above form of $C^{(k)}$ corresponds to the standard case of the approximate lumpability Problem (2.10). In the cases where we consider analytic functions of the lumpable transition matrix, as presented in Proposition 2.3.7, we define the row condition matrix by:

$$(C^{(k)})_{i,j} = \begin{cases} X_j & \text{if } i = k \text{ and for all } j, \\ 0 & \text{otherwise,} \end{cases}$$

where X_j is the i-th entry of the vector X is as defined in Proposition 2.3.7. Moreover, the resulting rows-sums $c^{(k)} \cdot f(p)$, will differ from 1. Specifically, we will have $c^{(k)} \cdot f(p) = Y_k$, where Y_k is the k-th entry of the vector Y also as defined in 2.3.7. Therefore, in this problem, the first $m(n-m)$ entries of b will remain 0's and the remaining n entries will be those of Y.

We outline the complete algorithm for solving the problem of approximate lumpability, using the methodology described:

Example 2.4.3. To illustrate the algorithm, consider the partition $S' = \{\{1,2\},\{3\}\}$ of the state space $S = \{1,2,3\}$. We have the two elements $A_1 = \{1,2\}$ and $A_2 = \{3\}$ of S', whose partition matrices are given by:

$$L_1 = \begin{pmatrix} 1 & 1 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad L_2 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{pmatrix}.$$

Algorithm 1 Construction of the condition matrix

Given an initial and aggregate state space S and S', respectively, with $|S| = n$ and $|S'| = m$, with mapping $\phi : S \to S'$ and the partition matrices L_i, for $i \in S'$ as defined above:

1. Construct the condition matrix A as follows:

(a) For $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n-1$, construct all matrices $L_i^{(j)}$.

(b) For each $L_i^{(j)}$, if $\phi(j) = \phi(j+1)$ calculate $l_{\psi(i,j)}$, and consequtively fill in the rows of A with vectors $l_{\psi(i,j)}$.

(c) Starting with $C^{(1)}$, construct row condition vectors $C^{(k)}$ and obtain vectors $c^{(k)}$. Fill in the final n rows of A with $c^{(k)}$, for $k = 1, 2, \ldots, n$.

2. Construct the reparameterized transition matrix p.
3. Construct vector b.
4. Use iterative scheme (2.12) (or (2.16) depending on the problem at hand) along with an appropriate stopping criterion, to find the entries of the lumpable matrix in vector form p_L.
5. Use relabelling $(P_L)_{i,j} = (p_L)_{\psi^{-1}(i,j)}$ to obtain the solution in matrix form.

We perform the iterations for A_1 first. We check that the condition $\phi(1) = \phi(2) = s_1'$ is satisfied and we therefore accept the version of the partition matrix $l_1^{(1)}$, which will be the first row of A. The next iteration will be:

$$L_1^{(2)} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ -1 & -1 & 0 \end{pmatrix}.$$

For this iteration, $\phi(2) = s_1' \neq \phi(3) = s_2'$, and therefore, this iteration will not be accepted as part of A.

We now turn to A_2. The condition has been checked and so we accept $l_2^{(2)}$ as the second row of A. The second iteration of this partition matrix will be:

$$L_2^{(2)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{pmatrix},$$

which is not accepted. Finally, we need to include the conditions related to the row-sums of P_L. As explained above, this is done by using the vector forms $c^{(1)}, k = 1, 2, 3$, of the row condition matrices:

$$C^{(1)} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad C^{(2)} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \quad C^{(3)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}. \tag{2.19}$$

We have now collected all the rows necessary to construct A, which consists of $l_1^{(1)}, l_2^{(2)}, c^{(1)}, c^{(2)}, c^{(3)}$. Notice that the row dimension of A is indeed $2(3-2)+3 = 5$, as expected.The column vector b will consist of 0's corresponding to the lumpability condition, and 1's corresponding to the stochasticity condition. We complete the

example with the full form of A and b displayed below.

$$A = \begin{pmatrix} 1 & 1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}. \tag{2.20}$$

Employing the method described in (2.12) and relabelling the solution into a 3×3 matrix solves the problem of approximate lumpability. ◁

It is also important to compare the results obtained from solving the problem of approximate lumpability to those that would occur from standard statistical methods, such as a Monte Carlo simulation applied to calculate the transition matrix of the aggregated Markov chain. To this end, we present a typical example, where we will compare the bootstrap estimate of the aggregated Markov chain transition probabilities with those obtained by solving the problem of approximate lumpability.

Example 2.4.4. Let $M = (S, P)$ be a Markov chain with $S = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ and:

$$P = \begin{pmatrix} 0.11 & 0.08 & 0.07 & 0.02 & 0.18 & 0.07 & 0.10 & 0.15 & 0.09 & 0.13 \\ 0.17 & 0.04 & 0.02 & 0.15 & 0.21 & 0.07 & 0.01 & 0.08 & 0.12 & 0.13 \\ 0.13 & 0.09 & 0.07 & 0.02 & 0.15 & 0.13 & 0.07 & 0.25 & 0.01 & 0.08 \\ 0.04 & 0.08 & 0.16 & 0.06 & 0.05 & 0.13 & 0.15 & 0.08 & 0.15 & 0.1 \\ 0.14 & 0.00 & 0.00 & 0.24 & 0.19 & 0.02 & 0.03 & 0.05 & 0.16 & 0.17 \\ 0.19 & 0.11 & 0.03 & 0.13 & 0.16 & 0.02 & 0.05 & 0.03 & 0.19 & 0.09 \\ 0.11 & 0.13 & 0.11 & 0.13 & 0.10 & 0.10 & 0.16 & 0.04 & 0.06 & 0.06 \\ 0.08 & 0.01 & 0.13 & 0.05 & 0.13 & 0.18 & 0.06 & 0.19 & 0.01 & 0.16 \\ 0.06 & 0.20 & 0.03 & 0.08 & 0.05 & 0.13 & 0.12 & 0.08 & 0.02 & 0.23 \\ 0.10 & 0.06 & 0.05 & 0.10 & 0.07 & 0.13 & 0.05 & 0.09 & 0.15 & 0.20 \end{pmatrix}.$$

We want to lump M on the partition $S' = \{\{1, 2, 3\}, \{4, 5\}, \{6, 7, 8\}, \{9, 10\}\}$. Using the algorithm to find the condition matrix A, we have $Ap_L = b$ (A will be a 34×10 matrix) and obtain the lumpable approximation to P (given the large dimensions of the condition matrix, for brevity we only present the resulting transition matrix; the intermediate matrices and results are available upon request).:

$$P_L = \begin{pmatrix} 0.11000 & 0.08000 & 0.07000 & 0.04167 & 0.20167 & 0.06667 & 0.09667 & 0.14667 & 0.07333 & 0.11333 \\ 0.18000 & 0.05000 & 0.03000 & 0.09167 & 0.15167 & 0.12000 & 0.06000 & 0.13000 & 0.08833 & 0.09833 \\ 0.12000 & 0.08000 & 0.06000 & 0.05667 & 0.18667 & 0.08333 & 0.02333 & 0.20333 & 0.05833 & 0.12833 \\ 0.01667 & 0.05667 & 0.13667 & 0.14000 & 0.13000 & 0.07833 & 0.10833 & 0.03833 & 0.16750 & 0.12750 \\ 0.16333 & 0.02333 & 0.02333 & 0.16000 & 0.11000 & 0.06167 & 0.07167 & 0.09167 & 0.14250 & 0.15250 \\ 0.18000 & 0.10000 & 0.02000 & 0.10167 & 0.13167 & 0.07889 & 0.10889 & 0.08889 & 0.14500 & 0.04500 \\ 0.09333 & 0.11333 & 0.09333 & 0.13167 & 0.10167 & 0.09222 & 0.15222 & 0.03222 & 0.09500 & 0.09500 \\ 0.10667 & 0.03667 & 0.15667 & 0.07667 & 0.15667 & 0.12889 & 0.00889 & 0.13889 & 0.02000 & 0.17000 \\ 0.04667 & 0.18667 & 0.01667 & 0.09000 & 0.06000 & 0.12000 & 0.11000 & 0.07000 & 0.04500 & 0.25500 \\ 0.11333 & 0.07333 & 0.06333 & 0.09000 & 0.06000 & 0.14000 & 0.06000 & 0.10000 & 0.12500 & 0.17500 \end{pmatrix}.$$

We can now find the transition matrix of the lumped process P_L^*:

$$P_L^* = \begin{pmatrix} 0.2600 & 0.2433 & 0.3100 & 0.1867 \\ 0.2085 & 0.2690 & 0.2285 & 0.2940 \\ 0.3000 & 0.2333 & 0.2767 & 0.1900 \\ 0.2500 & 0.1500 & 0.3000 & 0.3000 \end{pmatrix}.$$

We now wish to compare the above, which is based on the theory developed in previous Sections, with the aggregated transition matrix calculated from standard statistical estimators, based on observables. To this end we simulate 10,000 observations of the original 10-state Markov chain. We then map each state in the trajectory to its corresponding lumped state defined by the partition S'. In order to obtain the dynamics of the aggregated chain we apply a bootsrap maximum likelihood estimation with 100 paths to calculated the transition probabilities using the standard maximum likelihood estimator

$$\hat{p}_{i,j} = \frac{n_{i,j}}{N_i},$$

where $n_{i,j}$ is the number of transitions from state i to j and N_i is the total number of times the chain was in state i. The resulting transition matrix is given below:

$$P^*_{MLE} = \begin{pmatrix} 0.2590 & 0.2402 & 0.3156 & 0.1852 \\ 0.2118 & 0.2731 & 0.2163 & 0.2988 \\ 0.2848 & 0.2480 & 0.2661 & 0.2011 \\ 0.2397 & 0.1576 & 0.3077 & 0.2950 \end{pmatrix}.$$

We observe that P^*_{MLE} is an accurate approximation to the lumped version P^*_L calculated above, with an error of merely $\|P^*_{MLE} - P^*_L\|_2 = 0.0012$.

◁

2.5 Markov chain lumpability in credit risk

For the application of the techniques developed in the previous sections, we focus mainly on the problem of internal rating comparability and compatibility, which is prevalent as institutions develop their own internal rating system. In particular, we now have seen that institutions have to find a way to report such results in terms of the Stage 1, 2 and 3 classifications, introduced by IFRS 9. Therefore, we will need to solve approximate lumpability problems in order to aggregate original ratings (i.e., states of the original state space). It is fair to assume that, in practice, the states that must be lumped will be known. For example, institutions will be aware of the ratings in the pre-existing classifications which indicate a significant increase in credit risk. These ratings will be aggregated into Stage 2, according to the IFRS 9. Moreover, the methods of approximate lumpability can be used to assess which states to lump together, by selecting different partitions, which will naturally result in different P_L matrices, and comparing the errors $\|P_L - P\|_2$. However, the theory of lumpability can be applied to other problems pertaining to credit risk, as examined the first subsection below.

2.5.1 Applications in IFRS 9 modelling

2.5.1.1 Compatibility of internal models with the required IFRS 9 classification

We are now in the position to consider the application of Markov chain lumpability under the IFRS 9 framework. We first consider the following example, where we solve the problem of approximate lumpability in order to calculated the IFRS 9 - compliant transition matrix between Stages. As previously mentioned, this aggregate Markov chain is necessary for regulatory purposes, as well as for provisioning and pricing calculations.

For illustrative purposes, we start with a simple example, which allows us to further demonstrate in detail the construction of the partition and condition matrices that arise in the problem of approximate lumpability.

Example 2.5.1. Consider a Markov chain used to model loan performance, with a five-dimensional state space $S = \{s_1, s_2, s_3, s_4, s_5\}$ above and the transition matrix is given by:

$$P = \left(\begin{array}{c|ccccc} \text{Rating} & s_1 & s_2 & s_3 & s_4 & s_5 \\ \hline s_1 & 0.60 & 0.20 & 0.10 & 0.05 & 0.05 \\ s_2 & 0.20 & 0.40 & 0.20 & 0.10 & 0.10 \\ s_3 & 0.15 & 0.10 & 0.30 & 0.30 & 0.15 \\ s_4 & 0.00 & 0.00 & 0.00 & 1.00 & 0.00 \\ s_5 & 0.00 & 0.00 & 0.00 & 0.00 & 1.00 \end{array}\right).$$

Under IFRS 9, the aggregate state space we consider is defined by the partition $S' = \{s_1, \{s_2, s_3\}, \{s_4, s_5\}\} \simeq \{\text{Stage 1}, \text{Stage 2}, \text{Stage 3}\}$. We can check that P is not lumpable on this partition and hence we apply the approximate lumpability method with partition matrices:

$$L_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad L_2 = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad L_3 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

and condition matrices:

$$C^{(1)} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad C^{(2)} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad C^{(3)} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$C^{(4)} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad C^{(5)} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

Then, applying Algorithm 1, we solve the problem of approximate lumpability to obtain:

$$P_L = \left(\begin{array}{c|ccccc} \text{Rating} & s_1 & s_2 & s_3 & s_4 & s_5 \\ \hline s_1 & 0.6000 & 0.2000 & 0.1000 & 0.0500 & 0.0500 \\ s_2 & 0.1850 & 0.3600 & 0.1600 & 0.1475 & 0.1475 \\ s_3 & 0.1850 & 0.1600 & 0.43600 & 0.2725 & 0.0225 \\ s_4 & 0.0000 & 0.0000 & 0.0000 & 1.0000 & 0.0000 \\ s_5 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 1.0000 \end{array}\right).$$

We can now calculate the transition matrix of the lumped Markov chain, which describes the model under IFRS 9:

$$P^* = UP_LV = \left(\begin{array}{c|ccc} \text{IFRS 9 Staging} & \text{Stage 1} & \text{Stage 2} & \text{Stage 3} \\ \hline \text{Stage 1} & 0.600 & 0.300 & 0.100 \\ \text{Stage 2} & 0.185 & 0.520 & 0.295 \\ \text{Stage 3} & 0.000 & 0.000 & 1.000 \end{array}\right).$$

◁

We now present a more realistic example, based on data provided by Nickell and Perraudin, 2000.

Example 2.5.2. Consider the estimated rating transition matrix (see for example Nickell and Perraudin, 2000), given by (in percentages):

$$P = \left(\begin{array}{c|cccccccc} \text{Rating} & \text{Aaa} & \text{Aa} & \text{A} & \text{Baa} & \text{Ba} & \text{B} & \text{Caa} & \text{C/Ca} & \text{D} \\ \hline \text{Aaa} & 89.6 & 10 & 0.4 & 0 & 0 & 0 & 0 & 0 & 0 \\ \text{Aa} & 0.9 & 88.3 & 10.7 & 0.1 & 0 & 0 & 0 & 0 & 0 \\ \text{A} & 0.2 & 2.7 & 91.1 & 5.6 & 0.4 & 0 & 0 & 0 & 0 \\ \text{Baa} & 0 & 0.3 & 6.6 & 86.8 & 5.6 & 0.4 & 0.2 & 0 & 0.1 \\ \text{Ba} & 0 & 0.1 & 0.5 & 5.9 & 83.1 & 8.4 & 0.3 & 0 & 1.7 \\ \text{B} & 0 & 0.2 & 0.2 & 0.8 & 6.6 & 79.6 & 2.2 & 1.0 & 9.4 \\ \text{Caa} & 0 & 0 & 0 & 0.8 & 1.9 & 9.3 & 63.0 & 1.9 & 23.1 \\ \text{C/Ca} & 0 & 0 & 0 & 0 & 0 & 5.9 & 5.9 & 64.7 & 23.5 \\ \text{D} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 100 \end{array}\right).$$

We consider the partition $S' = \{\{Aaa, Aa, A\}, \{Baa, Ba, B, Caa, C/Ca\}, \{D\}\} \simeq \{\text{Stage 1}, \text{Stage 2}, \text{Stage 3}\}$, which will produce the IFRS 9 compliant transition matrix. It has been shown (see Loizides and Yannacopoulos, 2012) that the matrix is not lumpable with respect to this partition. Hence, we apply the approximate lumpability projection to find P_L, which produces the lumpable, yet non-stochastic matrix (in percentages):

$$\left(\begin{array}{c|cccccccc} \text{Rating} & \text{Aaa} & \text{Aa} & \text{A} & \text{Baa} & \text{Ba} & \text{B} & \text{Caa} & \text{C/Ca} & \text{D} \\ \hline \text{Aaa} & 88.91 & 9.32 & -0.28 & 0.41 & 0.41 & 0.41 & 0.41 & 0.41 & 0.00 \\ \text{Aa} & 0.25 & 87.65 & 10.05 & 0.49 & 0.39 & 0.39 & 0.39 & 0.39 & 0.00 \\ \text{A} & 1.52 & 4.02 & 92.41 & 4.81 & -0.39 & -0.79 & -0.79 & -0.79 & 0.00 \\ \text{Baa} & -1.77 & -1.47 & 4.37 & 85.57 & 4.37 & -0.83 & 1.03 & 1.23 & 11.56 \\ \text{Ba} & 0.33 & 0.43 & 0.83 & 3.73 & 80.93 & 6.23 & -1.87 & -2.17 & 11.56 \\ \text{B} & 0.39 & 0.59 & 0.59 & 0.13 & 5.94 & 78.94 & 1.53 & 0.33 & 11.56 \\ \text{Caa} & 0.53 & 0.53 & 0.53 & 2.79 & 3.89 & 11.29 & 64.99 & 3.89 & 11.56 \\ \text{C/Ca} & 0.53 & 0.53 & 0.53 & 2.07 & 2.07 & 7.97 & 7.97 & 66.77 & 11.56 \\ \text{D} & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 100.0 \end{array}\right).$$

Therefore, to obtain the required lumpable and stochastic matrix, we must apply Dykstra's algorithm. After only 60 iterations, we can see that the negative values have been eliminated (to within an error of 10^{-11}) and the resulting matrix is (as in Example 2.4.4, we omit the partition and condition matrices for brevity):

$$P_L = \left(\begin{array}{c|cccccccc} \text{Rating} & \text{Aaa} & \text{Aa} & \text{A} & \text{Baa} & \text{Ba} & \text{B} & \text{Caa} & \text{C/Ca} & \text{D} \\ \hline \text{Aaa} & 88.2780 & 8.6780 & 0.0000 & 0.6088 & 0.6088 & 0.6088 & 0.6088 & 0.6088 & 0.0000 \\ \text{Aa} & 0.0000 & 87.2780 & 9.6780 & 0.6888 & 0.5888 & 0.5888 & 0.5888 & 0.5888 & 0.0000 \\ \text{A} & 1.1854 & 3.6854 & 92.0851 & 3.0439 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ \text{Baa} & 0.0000 & 0.0000 & 1.8708 & 84.1414 & 2.9414 & 0.0000 & 0.0000 & 0.0000 & 11.0464 \\ \text{Ba} & 0.4236 & 0.5236 & 0.9236 & 2.4609 & 79.6610 & 4.9609 & 0.0000 & 0.0000 & 11.0464 \\ \text{B} & 0.4903 & 0.6902 & 0.6903 & 0.1766 & 5.9766 & 78.9764 & 1.5766 & 0.3766 & 11.0464 \\ \text{Caa} & 0.6236 & 0.6236 & 0.6236 & 2.8366 & 3.9365 & 11.3365 & 65.0366 & 3.9366 & 11.0464 \\ \text{C/Ca} & 0.6236 & 0.6236 & 0.6236 & 2.1165 & 2.1165 & 8.0166 & 8.0166 & 66.8166 & 11.0464 \\ \text{D} & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 100.00 \end{array}\right).$$

We can finally calculate the dynamics of the lumped, obtaining the IFRS 9-compliant transition matrix:

$$P^* = UP_LV = \left(\begin{array}{c|ccc} \text{IFRS 9 Rating} & \text{Stage 1} & \text{Stage 2} & \text{Stage 3} \\ \hline \text{Stage 1} & 96.956 & 3.044 & 0.000 \\ \text{Stage 2} & 1.871 & 87.083 & 11.046 \\ \text{Stage 3} & 0.000 & 0.000 & 100.0 \end{array}\right),$$

with matrices U, V, corresponding to the partition, given by:

$$U = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad V = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

◁

2.5.1.2 Calculation of expected time in Stage 2 and expected lifetime

As mentioned, under the IFRS 9 framework, we are interested in the calculation of the expected time a loan will remain in Stage 2 (as the institution will be required to consider lifetime provisions for this time period), as well as the expected lifetime for revolving loans. To motivate the theory necessary for addressing the modelling tasks related to IFRS 9, we first consider a simplified case, in which an institution has five possible loan classifications (in reality most banks have considerably more) and hence, the state space of the original Markov Chain is $S =$ {performing loans, 5 % increase in Probability of Default, 30 or more days past due, 90 or more days past due, NPL}. We will use the notation $S = \{s_1, s_2, s_3, s_4, s_5\}$. Under this model, s_1, s_2 and s_3 are transient states of the chain, while s_4 and s_5 are absorbing states (since we consider that the loan is has and therefore cannot return to any other state). The canonical form of this generic transition matrix would be:

$$P = \begin{pmatrix} Q_{1,1} & Q_{1,2} & Q_{1,3} & R_{1,1} & R_{1,2} \\ Q_{2,1} & Q_{2,2} & Q_{2,3} & R_{2,1} & R_{2,2} \\ Q_{3,1} & Q_{3,2} & Q_{3,3} & R_{3,1} & R_{3,2} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

with:

$$Q = \begin{pmatrix} Q_{1,1} & Q_{1,2} & Q_{1,3} \\ Q_{2,1} & Q_{2,2} & Q_{2,3} \\ Q_{3,1} & Q_{3,2} & Q_{3,3} \end{pmatrix}, \quad R = \begin{pmatrix} R_{1,1} & R_{1,2} \\ R_{2,1} & R_{2,2} \\ R_{3,1} & R_{3,2} \end{pmatrix}. \tag{2.21}$$

We need a way to perform our analysis under the state space $\{\text{Stage 1}, \text{Stage 2}, \text{Stage 3}\}$, in order to comply with the IFRS 9 framework. We can achieve this by lumping the original transition matrix. The resulting lumped

matrix will be of the form:

$$P^* = \begin{pmatrix} Q^*_{1,1} & Q^*_{1,2} & R^*_{1,3} \\ Q^*_{2,1} & Q^*_{2,2} & R^*_{2,3} \\ 0 & 0 & 1 \end{pmatrix}.$$

The form of the last row is due to the fact that Stage 3 is an absorbing state of the Markov chain (we consider that the loan does not return to another Stage once it has defaulted). Consider lumping the original transition matrix above, where we have a five-state Markov Chain. In this case, $\{s_1\}$ would correspond to Stage 1 (performing loans), $\{s_2, s_3\}$ would be lumped to Stage 2 (loans which have displayed a significant increase in credit risk) and $\{s_4, s_5\}$ would be lumped to Stage 3 (defaulted loans). To check the lumpability condition, we would need to use U, V:

$$U = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}, \quad V = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}, \tag{2.22}$$

Generally, we expect that the original matrices will not be lumpable, since they are usually based on historical transitional data. Hence, we will have to return to the notion of approximate lumpability, as described in (2.10). We can now find the entries of P_L. This is relatively straightforward using the methods described in the previous sections. For illustrative purposes we present the example below.

For the two quantities of interest, namely, the estimation of total time spent in Stage 2 and of the lifetime for revolving loan, of great importance is the fundamental matrix, $N \in \mathbb{R}^{m\times m}$, of the Markov chain. In this setting, if T_i is the random variable representing the absorption time of the chain, given that the chain started from state i, then $\mathbb{E}[T_i] = (N\mathbf{1}_m)_i$. This quantity can be used to address both modelling tasks mentioned.

These calculations can be done by using the lumped transition matrix, corresponding to the simple approximate lumpability problem, as done is the example above. However, we propose another method, using the fundamental matrix. Due to the importance of the fundamental matrix under the IFRS 9 framework, it makes sense to approximate the matrix in a way that minimizes the quantity $\|N_L - N\|_2$. Recall that the fundamental matrix gives the absorption time of the process, which is equivalent to the time taken for a loan to enter any default state. Therefore, since the fundamental matrix only depends on the block Q of the canonical form of $P \in \mathbb{R}^{n\times n}$, the approximation P_l will be of the form:

$$P_l = \begin{pmatrix} Q_L & R \\ \mathbf{0}_{r\times(n-r)} & I_{r\times r} \end{pmatrix},$$

where we recall that Q_L is an $(n-r)\times(n-r)$ matrix containing all transition probabilities between transient states and R is an $(n-r)\times r$ matrix containing the transition probabilities from transient states to the absorbing state (defaulted state). Of course, the lumpability of Q_L does not imply lumpability of P (hence the notation P_l, to avoid confusion with the lumpable P_L); however, in this application we are only concerned with Q, so it suffices to only consider the lumpable Q_L. Depending on the application and minimization problem, one would have to examine which part of the canonical form is to be lumped, as well as the effect on P.

Remark 2.5.3. It is important to note that, since we are only interested in the lumping of Q, we do not need to consider the lumpability of the whole matrix P and we write the problem in terms of Q directly. To see this, consider a 4×4 transition matrix, in canonical form (for simplicity, we consider only one absorbing state, but we can easily see that the calculations below work for any dimensions of R):

$$P = \begin{pmatrix} Q_{1,1} & Q_{1,2} & Q_{1,3} & R_1 \\ Q_{2,1} & Q_{2,2} & Q_{2,3} & R_2 \\ Q_{3,1} & Q_{3,2} & Q_{3,3} & R_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

We consider the lumping of states $\{2, 3\}$. Then:

$$U = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad V = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{2.23}$$

After elementary matrix operations, the lumpability condition $VUPV = PV$ results in the system of equations:

$$\begin{aligned} Q_{2,1} &= Q_{3,1} \\ Q_{2,2} + Q_{2,3} &= Q_{3,2} + Q_{3,3} \end{aligned}$$

We arrive at the same conditions if we were to simply consider the lumping of Q and the simpler versions of the above V and U corresponding only to the states of Q, i.e,

$$U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}, \quad V = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}. \tag{2.24}$$

This implication follows directly from the form of the matrices V, U, which characterize the lumping. When performing the matrix multiplication, the entries of R will not contribute to the condition arising for the entries of Q (and vice versa). Therefore, in what follows we can focus our attention on the lumpability of Q. ◁

Now, we wish to lump Q_L into a two-state transition matrix with states $\{\text{Stage 1}, \text{Stage 2}\}$. Hence, $VUQ_LV = Q_LV$, with U, V as in (2.24). In addition, if $Q_{i,j}$ and $R_{i,j}$ represent the entries of Q_L and R respectively, then

$$\sum_{j=1}^{n-r} Q_{i,j} = 1 - \sum_{j=1}^{r} R_{i,j}, \text{ for } i = 1, 2, \ldots, n-r$$

Equivalently, in matrix form, the constraints can be written as:

$$\begin{aligned} VUQ_LV &= Q_LV \\ Q_L\mathbf{1}_{(n-r)} &= \mathbf{1}_{(n-r)} - R\mathbf{1}_r. \end{aligned} \tag{2.25}$$

Therefore, the full approximate lumpability problem is:

$$\begin{aligned} \min_{Q_L \in \mathbb{R}^{(n-r)\times(n-r)}} \quad & \|(I_{(n-r)\times(n-r)} - Q_L)^{-1} - (I_{(n-r)\times(n-r)} - Q)^{-1}\|_2 \\ \text{subject to} \quad & VUQ_LV = Q_LV, \quad Q_L\mathbf{1}_{(n-r)} = \mathbf{1}_{(n-r)} - R\mathbf{1}_r, \quad (Q_L)_{i,j} \geq 0. \end{aligned} \tag{2.26}$$

This problem is in the form addressed in Proposition 2.3.7. For clarity, we outline the steps in detail, following the same structure as the proof of the Proposition. In what follows, we suppress the dimensions of the identity matrix $I_{(n-r)\times(n-r)}$, when possible, for notational ease.

By assumption, we have that $VUI_{(n-r)\times(n-r)}V = I_{(n-r)\times(n-r)}V$ and $VUQ_LV = Q_LV$. By linearity of matrix multiplication we get $VU(I - Q_L)V = (I - Q_L)V$, and by directly applying Lemma 2.3.5, we obtain $VU(I - Q_L)^{-1}V = (I - Q_L)^{-1}V$. We can also change the form of the second condition, to obtain the condition for the row-sums of the fundamental matrix:

$$Q_L\mathbf{1}_{(n-r)} = \mathbf{1}_{(n-r)} - R\mathbf{1}_r \implies (I_{(n-r)\times(n-r)} - Q_L)^{-1}R\mathbf{1}_r = \mathbf{1}_{(n-r)}.$$

Using Proposition 2.3.7, we obtain the condition matrix for this problem:

$$A' = \begin{pmatrix} (VU - I_{(n-r)\times(n-r)}) \otimes V^T \\ I_{(n-r)\times(n-r)} \otimes (R\mathbf{1}_r)^T \end{pmatrix},$$

as well as the vector $b' = [\mathbf{0} \;\; \mathbf{1}_{(n-r)}]^T$.

Hence, the first two conditions can be written as $A'n_L = b'$, where n_L is the relabelling of the entries of $(I - Q_L)^{-1}$ as a vector. The minimization problem can now be written as:

$$\begin{aligned} \min_{Q_L \in \mathbb{R}^{m\times m}} \quad & \|N_L - N\|_2 \\ \text{subject to} \quad & A'n_L = b', \quad (Q_L)_{i,j} \geq 0. \end{aligned} \tag{2.27}$$

This now is of the same form as the general minimization problem considered in Proposition (2.3.7), which we can solve using a combination of Dyksta's algorithm and the orthogonal projection calculated by the Moore-Penrose inverse. Sepecifically, let n be the relabelling of the entries of $N = (I - Q)^{-1}$ as a vector. By following the same steps, we obtain:

$$n_L = (A')^\dagger(b' - A'n) + n.$$

Relabelling the entries of n_L back into matrix form N_L, we can solve to find the lumpable matrix Q_L. Therefore,

$$Q_L = I_{(n-r)\times(n-r)} - (N_L)^{-1} = I_{(n-r)\times(n-r)} - ((A')^\dagger(b' - A'n) + n)^{-1}, \tag{2.28}$$

where the inverse has been taken on the relabelled matrix form of n_L. Therefore, in this case, we use (2.28) to project onto the set $\mathcal{L}'$, where $\mathcal{L}'$ is the set of all matrices Q_L satisfying $A'n_L = b'$ ($\mathcal{L}'$ is a different set than $\mathcal{L}$ in this case, since we are considering row-sum conditions which differ from the standard case), followed by iterations of Dykstra's algorithm (if necessary) to converge to solution on $\mathcal{L} \cap \mathcal{M}_n^+$, as required.

We now have a solution to the approximate lumpability problem. Using this solution we can easily calculate any quantities of interest, such as the absorption time under the IFRS 9 framework. As explained, institutions can use these quantities in

order to address the problem of expected lifetime and also estimate the total time a loan remains in State 2, during which the bank has to increase its provisions. Therefore, this has an impact not only on the reporting, but also on the actual calculation of the provisions.

Example 2.5.4 (Calculation of expected time in Stage 2 and expected lifetime)**.** Consider a loan portfolio, where loans are modelled as a discrete Markov chain, with state space S ={Performing, 5% increase in PD, 30-89 days past due, NPL}, and note that the same methodology can be applied for larger state spaces. Using historical transition data, the canonical form of the transition matrix is given by:

$$P = \left(\begin{array}{c|cccc} \text{Internal Rating} & \text{Performing} & \text{5\% PD increase} & \text{30-89 DPD} & \text{NPL} \\ \hline \text{Performing} & 0.75 & 0.10 & 0.10 & 0.05 \\ \text{5\% PD increase} & 0.35 & 0.50 & 0.05 & 0.10 \\ \text{30-89 DPD} & 0.10 & 0.30 & 0.50 & 0.10 \\ \text{Default} & 0.00 & 0.00 & 0.00 & 1.00 \end{array}\right).$$

From P, we see that:

$$Q = \begin{pmatrix} 0.75 & 0.10 & 0.10 \\ 0.35 & 0.50 & 0.05 \\ 0.10 & 0.30 & 0.50 \end{pmatrix}, \quad R = \begin{pmatrix} 0.05 \\ 0.10 \\ 0.10 \end{pmatrix}. \tag{2.29}$$

In line with the theory previously described, we wish to lump Q in such a way that it describes the transition of the transient states under IFRS 9. To this end, we want to lump states s_2 (5% increase in PD) and s_3 (30-89 days past due) together, as they constitute criteria for classification of a loan as Stage 2. The new state space is therefore $S' = \{\{s_1\}, \{s_1, s_2\}, \{s_3\}\}$, with:

$$U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}, \quad V = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}.$$

A direct computation shows the that condition $VUQV = QV$ does not hold, and therefore, Q is not lumpable, as expected. We, therefore, resort to the problem of approximate lumpability. Moreover, we may want to solve this problem by minimizing the error in the fundamental matrix, thus returning to problem (2.26). We obtain the following conditions for the entries of the lumpable fundamental matrix $N = (I_{3\times 3} - Q_L)^{-1}$.

$$\begin{aligned} n_4 - n_7 &= 0 \\ n_5 + n_6 - n_8 - n_9 &= 0 \\ 0.05n_1 + 0.1n_2 + 0.1n_3 &= 1 \\ 0.05n_4 + 0.1n_5 + 0.1n_6 &= 1 \\ 0.05n_7 + 0.1n_8 + 0.1n_9 &= 1. \end{aligned}$$

We write these equations in matrix form to match (2.27). If we were to apply algorithm 1 to construct A, the partition matrices and condition matrix are given by:

$$L_1 = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad L_2 = \begin{pmatrix} 0 & 1 & 1 \\ 0 & -1 & -1 \\ 0 & 0 & 0 \end{pmatrix}, \quad C^{(1)} = \begin{pmatrix} 0.05 & 0.1 & 0.1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Therefore, $A'n_L = b'$, where:

$$A' = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & -1 & -1 \\ 0.05 & 0.1 & 0.1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.05 & 0.1 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.05 & 0.1 & 0.1 \end{pmatrix}, \quad n_L = \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \\ n_6 \\ n_7 \\ n_8 \\ n_9 \end{pmatrix}, \quad b' = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

By Lemma 5.2, the best approximation to the fundamental matrix is $n_L = (A')^{\dagger}(b' - A'n) + n$, which gives:

$$n_L = \begin{pmatrix} 9.31 \\ 3.17 \\ 2.18 \\ 6.63 \\ 4.68 \\ 2.00 \\ 6.63 \\ 3.24 \\ 3.44 \end{pmatrix}, \quad N_L = \begin{pmatrix} 9.31 & 3.17 & 2.18 \\ 6.63 & 4.68 & 2.00 \\ 6.63 & 3.24 & 3.44 \end{pmatrix},$$

and hence, solving for the lumpable version of Q, we get $Q_L = I_{3\times 3} - (N_L)^{-1}$ and

$$Q_L = \begin{pmatrix} 0.75 & 0.10 & 0.10 \\ 0.25 & 0.54 & 0.11 \\ 0.25 & 0.24 & 0.41 \end{pmatrix}.$$

We have obtained Q_L which satisfies the non-negativity condition, which means there is no need to apply Dysktra's algorithm. This choice of Q_L solves the minimization problem and one can easily check that it indeed satisfies the necessary conditions posed in the problem. We can now lump Q_L and N_L to obtain the dynamics of the new Markov chain:

$$Q_L^* = UQ_LV = \left(\begin{array}{c|cc} \text{IFRS 9 Rating} & \text{Stage 1} & \text{Stage 2} \\ \hline \text{Stage 1} & 0.75 & 0.20 \\ \text{Stage 2} & 0.25 & 0.65 \end{array}\right), \quad N_L^* = UN_LV = \begin{pmatrix} 9.31 & 5.34 \\ 6.63 & 6.68 \end{pmatrix}.$$

A straightforward computation shows that N_L^* is in fact the fundamental matrix produced from Q_L^*, as expected. One could now use N_L^* to deduce that if a loan starts in Stage 1, it is expected to be classified as Stage 2 for 5.34 years (if we assume the analysis for the transition matrix has been done on a annual basis) and the total time to absorption is the first entry of $N_L^*\mathbf{1}_2$, which would be 14.65 years. These results provide the estimates for the quantities we need, under IFRS 9. ◁

2.5.2 Further applications in credit risk

In this final section, we consider various problems in credit risk where the lumpability of the underlying Markov chains can play an important role. We highlight that

the common requirement across these, as in the previous cases regarding the IFRS 9 framework, is that of comparability and consistency, which can now be directly addressed using the techniques of lumpable Markov chains that have been developed in the previous chapters.

2.5.2.1 Comparability of credit ratings for the same issuer provided by different rating agencies

Example 2.5.5 (Compatibility of sovereign ratings using lumpability)**.** In this example, we consider a case based on Hill, Brooks, and Faff, 2009, where differences in ratings from the three main credit agancies and their effects are examined at a sovereign level (of course, other work has been done on a corporate level as well, as in e.g., Moon and Stotsky, 1993 and Cantor and Packer, 1997 and the methods that follow could be applied in the same way to these cases). We will consider the credit ratings applied by the agencies Standar and Poor's (S&P), Moody's and Fitch and their respective transition matrices. Despite having the same cardinality, the explicit ratings differ between agencies. For this reason a consolidated broad rating is often considered for comparison purposes. However, it now becomes necessary to calculate the dynamics of the transition matrix under the new aggregated state space defined by the consolidated ratings, whilst preserving the Markov property. To do this, we can apply the method of approximate lumpability and lump the resulting matrices to the new state space, defined by an appropriate partition of the original ratings (see Appendix A.1 for detailed transition matrices and the mapping table between individual and consolidated ratings which defines the required partition). For illustration purposes, we perform these operations for the transition matrices as given in Hill, Brooks, and Faff, 2009, however, an identical process can be followed for any ratings transition matrices, over any aggregate state space. The resulting transition matrices on the aggregated state space for each of the three credit rating agencies are given below (note that, in the resulting matrices, default is not considered an absorbing state, as sovereigns may recover from defaults):

Moody's* =

Broad Rating	8	7	6	5	4	3	2	1	Default
8	0.9282	0.0718	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7	0.0850	0.8837	0.0313	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0634	0.9002	0.0364	0.0000	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0951	0.8636	0.0315	0.0000	0.0000	0.0000	0.0098
4	0.0000	0.0000	0.0003	0.0810	0.8095	0.1087	0.0005	0.0000	0.0000
3	0.0000	0.0000	0.0000	0.0000	0.0537	0.8196	0.1043	0.0034	0.0190
2	0.0000	0.0000	0.0000	0.0000	0.0000	0.1352	0.7439	0.1209	0.0000
1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0012	0.1965	0.8023	0.0000
Default	0.0000	0.0009	0.0688	0.0077	0.0632	0.0603	0.0091	0.0003	0.7897

.

Fitch* =

Broad Rating	8	7	6	5	4	3	2	1	Default
8	0.9924	0.0076	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7	0.0505	0.9283	0.0212	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0695	0.9129	0.0176	0.0000	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.1759	0.7882	0.0359	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	0.0000	0.0790	0.8471	0.0716	0.0000	0.0000	0.0023
3	0.0000	0.0000	0.0000	0.0000	0.1404	0.8005	0.0579	0.0000	0.0012
2	0.0000	0.0000	0.0000	0.0000	0.0000	0.1957	0.7215	0.0295	0.0533
1	0.0000	0.0000	0.0000	0.0000	0.0009	0.2685	0.0136	0.0901	0.6269
Default	0.0000	0.0000	0.0000	0.0004	0.0072	0.3476	0.0333	0.0013	0.6102

.

$$\text{S\&P}^* = \left(\begin{array}{c|ccccccccc} \text{Broad Rating} & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & \text{Default} \\ \hline 8 & 0.9789 & 0.0211 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 7 & 0.0267 & 0.9524 & 0.0209 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 6 & 0.0000 & 0.0581 & 0.9274 & 0.0145 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 5 & 0.0000 & 0.0000 & 0.1556 & 0.8269 & 0.0175 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 4 & 0.0000 & 0.0000 & 0.0000 & 0.0798 & 0.8394 & 0.0808 & 0.0000 & 0.0000 & 0.0000 \\ 3 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0917 & 0.8215 & 0.0577 & 0.0037 & 0.0254 \\ 2 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.2662 & 0.3795 & 0.0664 & 0.2879 \\ 1 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.1659 & 0.0108 & 0.6369 & 0.1864 \\ \text{Default} & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0014 & 0.5202 & 0.1324 & 0.0075 & 0.3385 \end{array}\right).$$

◁

Differences in internal ratings systems and inconsistency in risk classification is of great importance in the financial industry and can affect many aspects of financial modelling, such as the pricing and optimization of diversified credit portfolios. A detailed analysis of such implications is given in e.g., Jacobson, Lindé, and Roszbach, 2006, where the "results reveal, for a portfolio with identical counterparts, substantial differences in the implied riskiness between banks. Such differences could translate into different amounts of required economic capital and create (new) incentives to securitize part of their loan portfolios or increase the riskiness of loans in certain rating classes". Therefore, modelling tasks may require a consistent rating scheme for credit instruments originating from different institutions (or even different countries), from which the necessity of defining robust risk classification criteria and finding the subsequent quantitative dynamics of these classifications, is evident. These issues can be efficiently tackled using the IFRS 9 framework for Staging and the theory of Markov chain lumpability, respectively. To give an example of a pricing model where the differences in internal rating would play an important role, we refer the interested reader to e.g., Acharya, Das, and Sundaram, 2019, where a pricing mechanism based on spreads across rating classes is developed. When applying such methods, consistency and comparability in ratings across all instruments is imperative for the pricing calculations, and therefore methods such as those applied in the example above can be directly employed to overcome discrepancies when pricing and creating a portfolio of assets whose original institutional ratings differ. We consider such a case in the example that follows.

2.5.2.2 Applications in credit portfolio construction and securitization

Example 2.5.6 (Optimal porfolio selection using lumpability)**.** Suppose a securitization agency creates a portfolio consisting of loans (or credit derivatives) originating from three different institutions. The dynamics of the instruments will be different depending on the institution and we suppose they are described by the transition matrices $P_i, i = 1, 2, 3$. The agency aims to select the investment allocated to each of the types of instuments. In order to perform the optimization, the transisition matrices must be defined on the same state space. We therefore need to solve the problem of approximate lumpability to obtain the lumped versions $P_i^*, i = 1, 2, 3$, which are now all defined on the same state space and hence directly comparable.

With the common state space the agency then poses the following portfolio optimization problem: suppose $w_i, i = 1, 2, 3$ and $r_i, i = 1, 2, 3$ represent the weight of total investment allocated to each institution's set of loans and their average return, respectively (in principal, the rate of returns might depend on the characteristics of the transition matrices). Consider, furthermore, that the required rate of return is

set to be R. Let $Y_i(t), i = 1,2,3$ be the stochastic processes representing the state of an instrument originating from institution i at time t. The total loss function for the agency is then given by $L(t) = \sum_1^3 a_i w_i \mathbf{1}_{\{Y_i(t)=3\}}$, where $a_i, i = 1,2,3$ is the average loss rate of the instruments from each institution. In order to rebalance the portfolio at each period, the securitization agency is then interested in solving a portfolio optimization problem (we present a very simple such problem, which can be solved analytically to illustrate the use of the method). The optimization we consider is the following:

minimize $\mathbb{E}[U(L(t))]$, subject to

$$\begin{aligned} w_1 r_1 + w_2 r_2 + w_3 r_3 &= R, \\ w_1 + w_2 + w_3 &= 1, \end{aligned}$$

for an appropriate loss function U. While the following analysis can be extended to any convex loss function, for the sake of simplicity we illustrate the calculation selecting the quadratic loss function $U(L) = bL^2 - L$ (in the sense of a negative utility function). The expected loss is then:

$$\sum_{i=1}^{3} U(a_i w_i) p_3^i,$$

where $p_3^i = p_{1,3}^i + p_{2,3}^i$ is the total probability that instruments from institution i (originally in either Stage 1 or Stage 2) will default. The problem thus becomes:

minimize $f(w_1, w_2, w_3) := \sum_{i=1}^3 b(a_i w_i)^2 - a_i w_i$, subject to

$$\begin{aligned} w_1 r_1 + w_2 r_2 + w_3 r_3 &= R, \\ w_1 + w_2 + w_3 &= 1, \end{aligned}$$

This simple quadratic optimization problem can now be solved either analytically or numerically. One can easily obtain the expression for one of the weights, e.g. we find that w_3^* is given by:

$$w_3^* = \frac{p_3^1(2ba_1^2\delta\epsilon - \epsilon) + p_3^2(\gamma - 2ba_2^2\beta\gamma) - p_3^3}{2(p_3^1 ba_1^2\epsilon^2 + p_3^2 ba_2^2\gamma^2 + p_3^3 ba_3^2)},$$

where $\beta = \frac{R-r_1}{r_2-r_1}, \gamma = \frac{r_3-r_1}{r_2-r_1}, \delta = \frac{r_2-R}{r_2-r_1}$ and $\epsilon = \frac{r_3-r_2}{r_2-r_1}$. A straightforward substitution using the two conditions will result in the corresponding values w_1^* and w_2^*.

To illustrate the results of such an optimization, we consider the above setting with average returns from each institutions instruments $r = (r_1, r_2, r_3)^T$, loss rates $a = (a_1, a_2, a_3)^T$ and probability of of default $p_3 = (p_3^1, p_3^2, p_3^3)^T$, given by:

$$r = \begin{pmatrix} 0.1 \\ 0.4 \\ 0.1 \end{pmatrix}, \quad a = \begin{pmatrix} 0.7 \\ 0.7 \\ 0.7 \end{pmatrix}, \quad p_3 = \begin{pmatrix} 0.15 \\ 0.40 \\ 0.15 \end{pmatrix}.$$

Fixing the expected total return to be $R = 0.15$, the resulting optimal weights $w^* = (w_1^*, w_2^*, w_3^*)^T$ are:

$$w^* = \begin{pmatrix} 0.42 \\ 0.16 \\ 0.42 \end{pmatrix}.$$

◁

Further portfolio optimization problems can be found in Markov and semi-Markov portfolio literature, e.g., Pasricha et al., 2020b, where the authors consider a portfolio of risky bonds originating from the Industry and Service sector, by explicitly stating that it is assumed these sectors share the same ratings, which might not be the case in practice. By mapping the corresponding sector ratings we can aggregate the two transition matrices and proceed with the portfolio optimization schemes, under the aggregate state space.

2.6 Conclusion

The extensive tools available in Markov chain modelling, in combination with the vast range of fields in which these methods find applications, indicate that the notion of lumpability can be particularly useful in many problems. Compartemental models in biology (e.g., Gibson and Renshaw, 1998), applications in finance (e.g., Meko and Lyn, 2011), as well as in decision theory and data science (e.g., Fourneau, Lecoz, and Quessette, 2004) are only a few examples of fields in which it would be meaningful to consider the theory of lumpability.

In this Chapter we have focused on the use of Markov chain lumpability for comparison and compatiblity purposes, an issue of importance in credit risk. This is directly related to the new IFRS 9 regulations, which aim to introduce a universal framework for defining risky credit assets. In many such applications, such as loan staging under IFRS 9, one would expect the true transition matrix $\tilde{P}$ to be lumpable, with respect to an appropriate partition, since the theory dictates that the individual states have common characteristics and should be aggregated (for example, an increase in default probability and a forborne flag indication are both considered a significant increase in credit risk event, and the loans that exhibit such behavior should belong to the same state when considering the IFRS 9 framework i.e., the aggregated Stage 2). In this setting, one may even consider that the lumpable verion P_L of the historical transition matrix P, is closer (in the sense of an appropriate norm), to the true transition matrix, $\tilde{P}$. Hence, these techniques could also be used to assess the existing models and determine further statistical estimators.

Lastly, we have seen in section 2.5.2 that there exist many problems that can be addressed or simplified by applying the techniques of approximate lumpability. As shown in Proposition 2.3.7, there exists a large family of approximate lumpability problems that can be posed and solved, depending on the application. These provide many options, as it is possible to consider different approximate lumpability problems, depending on the observable in question (i.e. the function $f(P)$). When applying these methods in cases where f is invertible, we can obtain a complete Markov chain framework, by retrieving the lumpable transition matrix P_L. Finally, the numerical estimates and their error bounds in such applications can addressed using Proposition 2.3.9 and related results.

Chapter 3

Probability of Default estimation under stochastic models

3.1 Background

Perhaps one of the most influential changes due to the IFRS 9 is the requirement for financial institutions to consider Expected Lifetime Provisions (ECL), whereby future losses must be forecast using mathematically robust and rigorous methods, for all Stage 2 credit exposures, which are considered to have displayed a significant increase in risk. This estimation requires knowledge of the lifetime Probability of Default (PD) for all loan exposures, as well as additional risk parameters such as the Loss Given Default (LGD) and the Exposure at Default (EAD), and finally being able to update these quantities dynamically under changing market conditions. There exist recent papers detailing and studying the ECL calculation, such as Beerbaum, 2015 and Xu, 2016. We extend this modelling framework by employing the theory of stochastic processes and their dynamics for the estimation of lifetime PDs and future losses (i.e., forward looking provisions). In this Chapter, we aim to develop a stochastic modelling framework under which the PD process can be considered mathematically and practically. Using this approach we can address in a robust and efficient manner the challenging provisioning, forecasting and pricing tasks under IFRS 9.

In general, calculating default probabilities both analytically and numerically is of paramount importance in risk management and a broad range of financial applications. However, particularly under IFRS 9, credit loss forecasting has introduced the need for robust structural models, that can be used for pricing and provisioning purposes. To this end, we will consider stochastic models for the evolution the underlying asset value process, whose default will be studied as an appropriate firs-time-hitting problem. Therefore, it can be assumed that we are working mainly within portfolios of corporate and small business loans, where it is common practice to consider the company's assets to be governed by stochastic process (see e.g., Barndorff-Nielsen and Shephard, 2001b and Barndorff-Nielsen and Shephard, 2001a). Under this assumption, the PD associated with each loan depends on the underlying asset process and we can define the PD as the probability that the asset process falls below a fixed threshold. More specifically, we will assume that the asset process is governed by an Ornstein - Uhlenbeck (OU) process with a jump component, a member of the family of jump-diffusion processes. We note that practitioners may consider the evolution of asset-dependent processes instead, e.g., returns; such processes can still be described by similar stochastic models, rendering the methods proposed in this Chapter applicable in these cases, as well. For brevity, hereinafter we will refer to this underlying process as the asset process, with the understanding

that is can be replaced with any related dynamics considered appropriate by practitioners. Important theoretical background of such processes and their properties are given in Applebaum, 2009 and Øksendal and Sulem, 2007. The jump process will account for abrupt changes in the asset processes, which are very common in practice and are closely related to loan defaults. Obtaining the evolution of the PD values, based on the stochastic asset model, will allow us to then tackle various modelling tasks which are currently open problems for financial institutions under the IFRS 9 framework.

It is common in the literature to consider two separate cases for default probabilities:

- A variable starting time and constant time horizon, defined by the process.
- A variable time horizon and constant starting time.

Our results in this Chapter constitute a generalization that combines these two cases (as analyzed in Mishura and Ragulina, 2016 and Møller, 1995). Specifically, we consider a generalized "Probability of Default function" and prove that, under certain homogeneity assumptions, we can obtain Integral Equations (IEs) and Partial Integro-Differential Equations (PIDEs) for both aforementioned PD cases. Finally, using the IE formulation we will prove the existence of the PD values and the solvability of the PIDEs in the viscosity sense, in order to obtain estimates that can be used for the aforementioned modelling tasks, without having to always assume and/or prove strict regularity conditions, and further consider the conditions under which these solutions can be considered strong, with the required smoothness.

The above methodology will be used to consider real-life examples of loan provisioning calculations, scenario analysis and pricing, exemplifying the wide range of credit risk modelling tasks the proposed methodology can address. Finally, we note that, even though motivated by credit risk, the approaches detailed in this Chapter can find applications in other areas of financial mathematics, such as derivatives pricing, where the use of stochastic modelling remains prevalent, e.g., in the pricing of barrier options.

3.2 Aims and modelling framework

We start by discussing the PD process which, in its most general form, can be written as a function of both the starting time and maturity, as well as of the initial position of the corresponding asset value process. Furthermore, to accurately model real-life dynamics, it is necessary to account for the dependence on latent variables which affect the PD. Incorporating such processes, which in practice are e.g., macroeconomic variables or different market regimes, is of paramount importance as it largely affects PD estimation and subsequent modelling results. Rigorously accounting for these exogenous variables is therefore necessary, and will allow us to consider a large family of stochastic processes that can be used in practice. To begin, consider compact and bounded sets $\mathcal{D}, \mathcal{D}_i \subset \mathbb{R}$, for $i = 1, \dots, d$. Then, define the PD function:

Definition 3.2.1. Consider $x \in \mathcal{D}$ and the vector of (discrete or continuous) stochastic processes $(X_t^i)_{t\geq 0}$ with corresponding state spaces $\mathcal{D}_i$, for $i = 1, \dots, d$. Furthermore, consider the stochastic asset value process $(G_t)_{t\geq s}$, with initial value $G_s = x$ and which depends on $(X_t^i)_{t\geq 0}$, for $i = 1, \dots, d$. Then, we define the Probability of Default function $\Psi : \mathcal{D} \times \mathcal{D}_1 \times \cdots \times \mathcal{D}_d \times [0, T] \times [0, T] \to [0, 1]$, for some fixed

$T > 0$, by:

$$\Psi(x, x_s^1, x_s^2, \ldots, x_s^d, s, t) = \mathbb{P}\Big(\inf_{s \le r \le t} G_r \le 0 | G_s = x, X_s^1 = x_s^1, X_s^2 = x_s^2, \ldots, X_s^d = x_s^d \Big), \tag{3.1}$$

and the corresponding survival probability $\Phi : \mathcal{D} \times \mathcal{D}_1 \times \cdots \times \mathcal{D}_d \times [0, T] \times [0, T] \to [0, 1]$ by:

$$\Phi(x, x_s^1, x_s^2, \ldots, x_s^d, s, t) = 1 - \Psi(x, x_s^1, x_s^2, \ldots, x_s^d, s, t). \tag{3.2}$$

To motivate this definition and its usefulness, notice that by fixing s we obtain the standard finite-horizon ruin probability (see e.g., Mishura and Ragulina, 2016), whereas by fixing t we obtain the ruin probability with variable starting time, as defined in Møller, 1995, which can be used to define a martingale. Finally, allowing $t \to \infty$ we obtain the infinite-horizon ruin probability.

Modelling the evolution of PD functions has become even more important under IFRS 9, due to the increased complexity of provision calculations and Staging criteria. In general, all aforementioned PD functions, corresponding to variable maturity or starting times find many applications and have been considered in the field of credit risk, such as in Ballotta and Fusai, 2015, Schoutens and Cariboni, 2010 and Zhou, 1997. For example, the case of a variable maturity is often referred to as the Lifetime Probability of Default and is used extensively for provisioning and pricing purposes. Particularly in the context of IFRS 9 modelling, the Lifetime Probability of Default is used to assess credit risk at origination, as well as for Expected Lifetime Provisions for Stage 2 loans. We give detailed examples of such calculations in Section 4.5.

As mentioned, it is standard in the field of financial mathematics to consider the evolution of a debtor's assets to be governed by a stochastic process. A well-documented process that is used in various such applications is the Ornstein-Uhlenbeck (OU) process. In Section 1.3.1.1 of the Introduction we outline important properties of the OU process, a generalized version of which we will consider in this paper. OU models have been considered in past research and many applications. For example, a well-known special case is the Vasicek model Vasicek, 1977. Further work has explored the Merton model for default with underlying dynamics given by the continuous OU process, and has been extended to cases incorporating jumps. These find important applications particularly in credit risk modelling and pricing; see e.g., Hull, Nelken, and White, 2004 and Barndorff-Nielsen and Shephard, 2001b, respectively.

By including a jump process to the continuous OU asset process, we obtain the Lévy-driven Ornstein-Uhlenbeck process, defined in (1.14). This is a natural generalization, as significant changes in credit events are often abrupt and unpredictable (particularly a deterioration in creditworthiness), corresponding to a discontinuous component in the driving stochastic process. Indeed, the goal of credit risk requirements under IFRS 9 is to ensure that financial institutions and their customers are protected against such rare and unexpected events and the subsequent losses. It is therefore important to capture the effect of such events mathematically, which is why this model will form the basis of our analysis and will be used to construct more sophisticated models in the next section. We will employ the fact that the jump OU process is time homogeneous so that, rather than considering a starting time s and initial position G_s, we can define the time until maturity by $u := t - s$ and consider

$(G_u)_{u\geq 0}$, as above, equivalently. This is an important property that we will take advantage of to simplify the PD estimation.

To conclude, we note that the use of Lévy processes for financial modeling is well-documented and established. Schoutens and Cariboni, 2010 gives an extensive analysis of Lévy processes and their use for asset process modelling, credit derivatives pricing and more. In Luciano and Schoutens, 2006 and Onalan, 2009 the authors consider a Lévy-driven OU process, and Lévy multivariate models for assets processes. The former fits the model parameters to the General Motors stock price, while the latter considers many different indices, obtaining suprisingly accurate results. We also refer the interested reader to Ballotta and Bonfiglioli, 2016 for a detailed analysis of the properties of the multivariate model. Seminal work has also been done in the study of Lévy-driven OU processes in Barndorff-Nielsen and Shephard, 2001b. Finally, well-documented numerical methods exist for the calibration of stochastic models with jumps, such as the Yuima framework for stochastic differential equations in R statistical language Brouste et al., 2014.

3.3 The generalized asset value model and PD function

In this section we develop a stochastic model that incorporates the exogenous variables required when considering asset value processes. To incorporate such effects, we build upon the family of regime switching and stochastic volatility models, as described below. We combine these to produce a generalized model, which we will use to construct a framework that encapsulates a large family of stochastic processes that can be used for asset value modelling and subsequent credit risk calculations. In addition to the mathematical results presented in this Chapter, we highlight that the framework developed using the generalized model addresses the strict requirements under IFRS 9, whereby credit risk modelling is required to incorporate multiple appropriate latent variables, whilst adhering to mathematical rigor.

3.3.1 Regime switching and stochastic volatility models

First recall that loan exposures under the IFRS 9 framework are now classified into three Stages. Each of these Stages correspond to a given level of risk, with the most noteworthy change being the introduction of Stage 2 loans, i.e., credit exposures which have exhibited a significant increase in credit risk (SICR event, which can be defined by the institution, e.g., as a statistically significant increase in PD, a delinquency warning flag etc.). By definition, changes in the risk profile of an exposure will correspond to changes in the dynamics of the underlying asset process. For example, a debtor may request restructuring, or may be 30 days delinquent. This will trigger a SICR event, which can then affect the underlying asset value process. To capture this dependency we consider a regime switching model for the asset process, whereby the parameters of the stochastic process vary according to the underlying rating (Stage) of the exposure. We can do this by considering the Continuous Time Markov Chain (CTMC) $(R_t)_{t\geq 0}$ describing the rating at time t, where the set of all loan ratings is denoted by $\mathcal{R}$, with cardinality $|\mathcal{R}| = R$. Therefore, we obtain the following jump diffusion with Markov switching model:

$$dG_u = k(R_u)\big(\theta(R_u) - G_u\big)dt + \sigma(R_u)dB_u + \int_{\mathbb{R}} zN(du, dz), \quad G_0 = x, R_0 = \rho, \quad (3.3)$$

with $\rho \in \mathcal{R}$. Note that in subsequent sections we adopt the notation $k_\rho, \theta_\rho, \sigma_\rho$, for brevity. For a reminder of CTMC processes and their properties see Section 1.2.1.2 of the Introduction.

In the sequel, to develop a realistic model we want to capture the effects of macroeconomic variables, which naturally affect the evolution of the asset process; this is necessary for the modelling tasks we will consider under IFRS 9, as previously discussed. Typically, such latent variables are incorporated by considering stochastic volatility models, whereby the diffusion term of the asset process also evolves according to a stochastic process, as described by the coupled process:

$$\begin{cases} dG_t = \mu_x(G_t, Y_t)dt + \sigma_x(G_t, Y_t)dB_t + \int_{\mathbb{R}} zN(dt, dz), & G_s = x, \\ dY_t = \mu_y(Y_t)dt + \sigma_y(Y_t)dW_t, & Y_s = y, \end{cases} \tag{3.4}$$

for $y \in \mathcal{V}$ and where B_t and W_t are independent Brownian motions. Standard cases are the Bates' model, introduced in Bates, 1996, as well as the Heston model (see Benhamou, Gobet, and Miri, 2010), a version of which we consider below. In particular, letting $\mu_x(Y_t) = k(\theta - Y_t)$ and $\sigma_x(Y_t) = \sqrt{Y_t}$, we obtain the asset process driven by a stochastic volatility process, which follows the well-established Cox–Ingersoll–Ross (CIR) model, developed in Cox, Ingersoll Jr, and Ross, 2005 (note that both processes are time-homogeneous):

$$\begin{cases} dG_u = k(\theta - G_u)dt + \sqrt{Y_t}dB_t + \int_{\mathbb{R}} zN(dt, dz), & G_0 = x, \\ dY_u = \kappa(\mu - Y_u)dt + \xi\sqrt{Y_u}dW_t, & Y_0 = y. \end{cases} \tag{3.5}$$

The above models are widely used in mathematical finance and stochastic modelling. Regime switching is a well-documented approach in financial modelling (see Hamilton, 2010), with applications ranging from macroeconomics (e.g., Aristidou, 2018) to option pricing (e.g., Duan, Popova, and Ritchken, 2002, Hainaut and Le Courtois, 2014) and interest rate modelling (Goutte and Ngoupeyou, 2011). In the case of credit risk, the underlying Markov chain is considered as an indication of the market conditions, which significantly impacts credit exposures and ratings. In subsequent sections we add to the multitude of applications by using the PD function that arises from the regime switching model to estimate Lifetime provisions and scenario analysis under IFRS 9. When considering regime switching in asset processes it is important to note that financial institutions may currently have various credit rating systems, which are not compatible with the IFRS 9 staging (which requires three district Stages for exposure ratings). However, recent work has shown that this is not restrictive and IFRS 9 - compatible transition matrices can be obtained from the existing internal ratings (see Georgiou et al., 2021). Finally, we refer the reader to Zhu, Yin, and Baran, 2015 for a detailed analysis of more general regime switching jump diffusion processes, where the authors also consider the dynamics of the underlying Markov process to be a function of the initial position of the jump diffusion.

The stochastic volatility model is a natural extension, as it can be seen as the limit process of the regime switching model, as $\mathcal{R} = \mathbb{R}_+$. Such models, in the case of both continuous and jump processes, have also been considered for numerous applications in mathematical finance, particularly in pricing and hedging, such as in Toivanen, 2010 and Goutte, 2013.

3.3.2 The generalized model

Both the aforementioned models have important applications in credit risk modelling under the IFRS 9 framework. Combining the two, we obtain a generalized model that captures all the observable or latent variables required to estimate the PD evolution and subsequently tackle the IFRS 9 modelling tasks. This generalized model is of the form:

$$\begin{cases} dG_u = k(Y_u, R_u)\big(\theta(Y_u, R_u) - G_u\big)dt + \sigma(Y_u, R_u)dB_u + \int_{\mathbb{R}} zN(du, dz), \quad G_0 = x, \\ dY_u = \kappa(\mu - Y_u)dt + \xi\sqrt{Y_u}dW_t. \end{cases} \tag{3.6}$$

Specifically, we will be considering a combination of (3.3) and (3.5), which gives rise to the following.

Definition 3.3.1. Under the generalized model, the asset value process is defined by the triple $(G_t, R_t, Y_t)_{t\geq 0}$, capturing both the switching and volatility processes, and is given by:

$$\begin{cases} dG_u = k(R_u)\big(\theta(R_u) - G_u\big)dt + \sigma(R_u)\sqrt{Y_t}dB_u + \int_{\mathbb{R}} zN(du, dz), \quad G_0 = x, \\ dY_u = \kappa(\mu - Y_u)dt + \xi\sqrt{Y_u}dW_t, \end{cases} \tag{3.7}$$

with $G_0 = x, R_0 = \rho$ and $Y_0 = y$.

Before moving on to define the appropriate Probability of Default functions, it will be useful to define some notation.

Remark 3.3.2. An important note is that the transition probability of the generalized OU process is also uniformly continuous. This follows from the fact that its transition probability, for given $(R_0, Y_0) = (\rho, y)$, $p(x', x, t; \rho, y)$, is simply the coupling of the transition densities of the corresponding stochastic volatility models, which are continuous functions (see e.g., Ackerer, Filipović, and Pulido, 2018), and is therefore uniformly continuous on all closed and bounded intervals $\mathcal{D}$ we will consider.

Notation 3.3.3.

1. Throughout the remainder of this Chapter, we employ the notation Z_u^x to represent the stochastic process $(Z_u)_{u\geq 0}$, with $Z_0 = x$, where appropriate. We also generalize this notation to incorporate cases with additional underlying variables $X_t^1, X_t^2, \ldots, X_t^n$, by writing $Z_u^{(x_1, x_2, \ldots, x_n)}$ to represent $(Z_u)_{u\geq 0}$ with $X_0^i = x_i$ for $i = 1, 2, \ldots, n$, (the superscripts are to be understood as indices, i.e., the $i-$th underlying variable is $(X_t^i)_{t\geq 0}$).

2. In the following sections, when referring to the transition densities of the regime switching, stochastic volatility and generalized OU models, we will omit the dependence on the latent variables for brevity, as it will be obvious from the context.

3.3.3 The Probability of Default function

Following the definition of the PD function, as given in (3.1), under the generalized model (3.7) we will condition on the initial state of the regime switching and stochastic volatility processes, i.e., ρ and y, to obtain:

$$\Psi(x, \rho, y, s, t) := \mathbb{P}\Big(\inf_{s\leq r\leq t} G_r \leq 0 | G_s = x, R_s = \rho, Y_s = y\Big). \tag{3.8}$$

Under this assumption, we can utilize the time homogeneity property to write the PD function more succinctly, whilst still being able to obtain the evolution of the PD, both in the case of a variable maturity and variable starting time. We describe this in the lemma below.

Lemma 3.3.4. *Under the generalized model (3.7), the PD function with variable maturity $\tilde{\Psi}(x,\rho,y,s;t)$ and variable starting time $\tilde{\Psi}(x,\rho,y,s;t)$, can be retrieved from the generalized function $\Psi(x,\rho,y,u)$, where $u = t-s$ represents the remaining time until maturity.*

Proof. As mentioned, this observation follows immediately from the time homogeneity of the asset process, since:

$$\begin{aligned}\Psi(x,s,t) &= \mathbb{P}\Big(\inf_{s\le r\le t} G_r \le 0 | G_s = x, R_s = \rho, Y_s = s\Big)\\ &= \mathbb{P}\Big(\inf_{0\le r\le t-s} G_r \le 0 | G_0 = x, R_0 = \rho, Y_0 = y\Big) = \Psi(x,\rho,y,0,t-s). \quad (3.9)\end{aligned}$$

We can now write $\Phi(x,\rho,y,u)$ with $u := t-s$ representing the remaining time until maturity. □

The above shows that, by fixing the appropriate time and with a simple change of variables, we can obtain the evolution of both PD processes. It easily follows that this approach can be generalized to any time homogeneous stochastic processes. Throughout the remainder of this paper, we will therefore use the following formulation of the PD and corresponding survival process:

$$\Psi(x,\rho,y,u) := \mathbb{P}\Big(\inf_{r\le u} G_r \le 0 | G_0 = x, R_0 = \rho, Y_0 = y\Big) \equiv \mathbb{P}\Big(\inf_{r\le u} G_r^{(x,\rho,y)} \le 0\Big), \quad (3.10)$$

$$\Phi(x,\rho,y,u) := 1 - \Psi(x,\rho,y,u) = \mathbb{P}\Big(\inf_{r\le u} G_r^{(x,\rho,y)} > 0\Big). \quad (3.11)$$

Remark 3.3.5. It is worth emphasizing that we use the general term "time until maturity" purposefully, as it captures both PD cases, under the homogeneity assumption. We will continue to use this term throughout this and subsequent Chapters, to instill the importance of this generalization. Furthermore, the homogeneity assumption is strong, yet fair. Particularly in the case of corporate and/or small business loans, it is natural to consider such asset processes, since credit risk modelling is often done across complete financial/business cycles (e.g., years or quarters), over which the evolution of the asset process (or related return processes) will have similar dynamics, regardless of the exact point in time. However, even without this assumption, the approaches developed in this paper can be used by fixing the either the starting or maturity time in order to obtain whichever case of the PD process the modeller requires. Hence, this framework is useful for PD modelling under any asset value process.

Using the generalized model and the corresponding PD (or survival) process (3.10) (or (3.11)), we can prove certain mathematical properties that are required to ensure the existence of appropriate solutions for the Partial Integro-differential Equations (PIDEs) we will obtain for the PD functions. This creates a complete and robust mathematical framework which can be applied even without assuming or proving regularity, and is therefore applicable to a wide range of asset value models. At the same time, it is important to discuss the practical implications and applicability of the approaches described in this and subsequent sections. When considering real-life credit risk modelling tasks the state of the regime (e.g., the IFRS 9 Stage),

and/or the value of any underlying macroeconomic factors may be observable and can therefore be inserted explicitly into the generalized model (3.7), thereby obtaining the regime switching or stochastic volatility model, with PD functions given by:

$$\Psi(x,\rho,u) = \mathbb{P}\Big(\inf_{r\leq u} G_r^{(x,\rho)} > 0\Big), \tag{3.12}$$

$$\Psi(x,y,u) = \mathbb{P}\Big(\inf_{r\leq u} G_r^{(x,y)} > 0\Big), \tag{3.13}$$

respectively, and corresponding PD (or survival) functions $\Psi(x,\rho,u)$ and $\Psi(x,y,u)$ (or $\Phi(x,\rho,u)$ and $\Phi(x,y,u)$). These models can then be used for the tasks we consider under IFRS 9, and credit risk more generally. For this reason, in Chapter 4 we develop numerical schemes for such simplified versions of the generalized model, starting with the one-dimensional Lévy-driven OU asset and its corresponding PD function $\Psi(x,u) = \mathbb{P}\Big(\inf_{r\leq u} G_r^x \leq 0\Big)$ (and survival $\Phi(x,u)$), and continuing with the regime switching and stochastic volatility models, which will be used for related applications.

Remark 3.3.6. We refer the reader to section 1.3.1.3 of the Introduction for a brief overview of useful PDEs that the survival processes and transition densities of the non-jump versions of the above models satisfy (i.e., the versions which do not contain the Lévy jump process). These, known as Kolmogorov backward equations, can be written in terms of the infinitesimal generators of the stochastic processes, and will be referred to in the subsequent sections, where we will obtain similar equations for the Lévy-driven models. We also note that we adopt the notation for the generator operators used in these equations for the remainder of the Chapter, for convenience.

3.4 Integral characterization and properties of the PD function

As previously mentioned, our approach ultimately relies on deriving and solving (PIDEs) for the PD functions. To obtain these equations, we will first consider an integral equation (IE) characterization of the PD under the generalized model, from which we can obtain similar representations for the simplified models. We develop these IEs in this section, which will allow us to prove that the PD functions enjoy the properties required so as to be considered appropriate (either weak or strong) solutions to the PIDEs.

3.4.1 Required notation for the Integral Equations

To ease the calculations presented in this section, we first introduce some notation, which will be used for the integral equations.

Definition 3.4.1. Consider $x \in \mathcal{D}$ and the vector of stochastic processes $(X_t^i)_{t\geq 0}$, with corresponding state space $\mathcal{D}_i$, for $i = 0, 1, \ldots, d$. For a fixed time $u \in [0,T]$, we define the family of operators $\big(\mathcal{T}_s, s \in [0,u]\big)$, acting on a function $\phi : \mathbb{R}^{d+1} \times [0,T] \to [0,1]$, by:

$$\begin{aligned}\mathcal{T}_s\phi(x, x_0^1, x_0^2, \ldots, x_0^d, u)&\\ =\mathbb{E}[\phi(x, X_s^1, X_s^2, \ldots, X_s^d, u-s) | X_0^1 = x_0^1, X_0^2 = x_0^2, \ldots, X_0^d = x_0^d].&\end{aligned} \tag{3.14}$$

In our setting, x corresponds to the initial position of the asset value process, i.e., $G_0 = x$, and each X_t^i represents a latent variable, such as the CTMC in the regime switching model or the volatility in the stochastic volatility model. To give analytic forms that will be used in these two models, respectively, we specify the following cases:

(i) Let $(X_t^i)_{t\geq 0}$, for $i = 1, 2, \ldots, d$ be discrete space and independent stochastic processes, with state space $\mathcal{X}^i$, such that $|\mathcal{X}^i| < \infty$, and with transition probabilities $\pi_i(k, x_0^i, t) := \mathbb{P}(X_t^i = k | X_0^i = x_0^i)$ for $k \in \mathcal{X}^i$. Then:

$$\begin{aligned}&\mathcal{T}_s\phi(x, x_0^1, x_0^2, \ldots, x_0^d, u)\\ &\quad = \sum_{x_s^1 \in \mathcal{X}^1} \cdots \sum_{x_s^d \in \mathcal{X}^d} \phi(x, x_s^1, x_s^2, \ldots, x_s^d, u-s)\pi_1(x_s^1, x_0^1, s)\cdots\pi_n(x_s^d, x_0^d, s).\end{aligned} \tag{3.15}$$

(ii) Let $(X_t^i)_{t\geq 0}$, for $i = 1, 2, \ldots, d$, be continuous and independent random variables, with supports $\mathcal{D}^i$ and transition densities $q_i(k, x_0^i, t) := \mathbb{P}(X_t^i = k | X_0^i = x_0^i)$ for $k \in \mathcal{D}^i$. Then:

$$\begin{aligned}&\mathcal{T}_s\phi(x, x_0^1, x_0^2, \ldots, x_0^d, u)\\ &= \int_{x_s^1 \in \mathcal{D}^1} \cdots \int_{x_s^d \in \mathcal{D}^d} \phi(x, x_s^1, x_s^2, \ldots, x_s^d, u-s) q_1(x_s^1, x_0^1, s)\cdots q_n(x_s^d, x_0^d, s) dx_s^1 \cdots dx_s^d.\end{aligned} \tag{3.16}$$

It is easy to see that in the case where $(X_t^i)_{t\geq 0}$ contains both discrete and continuous stochastic processes the analytical expression will contains both summation and integral terms. Hereinafter, we will refer to $\mathcal{T}_s$, for a fixed $s \in [0, u]$ as the $s-$*operator*. In the calculations that follow we prefer to express the relevant integral equations in terms of the $s-$operator notation for convenience and simplicity.

3.4.2 Integral Equation formulations of the PD functions

The first step in our methodology entails deriving the IEs for the PD functions. As is convention in most of the literature, we perform our calculations with the survival probability and it is easy to see that the same steps can be used for the corresponding PDs. These equations prove to be very useful, as they will allow us to prove continuity and existence results for the PD function. We prove the result under the generalized model, from which analogous results for regime switching and stochastic volatility cases are easily obtained.

Proposition 3.4.2. *Consider the asset value process under the generalized model (3.7) with jump rate $\lambda = \nu(\mathbb{R})$ and jump size distribution $F(z)$. Furthermore, let $Q(x, \rho, y, u)$ and $p(x', x, s)$ be the survival probability and transition density, respectively, of the non-jump generalized OU process. Then, the survival probability $\Phi(x, \rho, y, u)$ satisfies the integral equation:*

$$\begin{aligned}&\Phi(x, \rho, y, u)\\ &\quad = \int_0^u \lambda e^{-\lambda s} \int_0^\infty \int_{\mathbb{R}} \mathcal{T}_s\Phi(x' + z, \rho, y, u) p(x', x, s) dF(z) dx' ds + e^{-\lambda u} Q(x, \rho, y, u).\end{aligned} \tag{3.17}$$

Proof. Consider the natural filtration $\mathcal{F}_t$ generated by the by the tri-variate process (G_t, R_t, Y_t). Recall that we use the notation $G_t^{(x,\rho,y)}$ to represent the asset value process depending on the regime CTMC and volatility process Y_t, with $G_0 = x$, $R_0 = \rho$ and $Y_0 = y$. By definition, $\Phi(x,\rho,y,u) = \mathbb{P}\Big(\inf_{r\leq u} G_r^{(x,\rho,y)} > 0\Big)$ and it will therefore be useful to define the martingale:

$$M_s = \mathbb{E}[\mathbb{1}\big(\inf_{r\leq u} G_r^{(x,\rho,y)} > 0\big)|\mathcal{F}_s], \tag{3.18}$$

for $s < u$. Furthermore, let τ be the time of the first jump and we will also define the stopping time:

$$\tau^* = \inf\{t < u : \{\Delta G_t^{(x,\rho,y)} \neq 0\} \cap \{G_s^{(x,\rho,y)} > 0,\ \forall s \in [0,t]\}\}, \tag{3.19}$$

i.e., the time the process first jumps, having not yet defaulted. It is easy to check that τ^* is indeed an $\mathcal{F}_t$-stopping time.

On the event $\{\tau > u\}$ no jumps occur within the examined time horizon and therefore $\Phi(x,\rho,y,u) = Q(x,\rho,y,u)$, where recall that $Q(x,\rho,y,u)$ is the survival probability of the non-jump generalized OU process. On $\{\tau \leq u\}$, we have:

$$\Phi(x,\rho,y,u) = \mathbb{E}[\mathbb{1}\big(\inf_{r\leq u} G_r^{(x,\rho,y)} > 0\big)|\mathcal{F}_0] = M_0 = \mathbb{E}[M_{\tau^*}], \tag{3.20}$$

where the last step follows from the Optional Stopping Theorem. Notice that $\mathbb{P}(\tau^* = \infty) > 0$ and therefore the above is to be understood in an almost sure sense. Now, by the strong Markov property and the time homogeneity of the OU process, it follows that:

$$\begin{aligned}\Phi(x,\rho,y,u) =& \mathbb{E}[\Phi(G_{\tau^*}^{(x,\rho,y)}, R_{\tau^*}, Y_{\tau^*}, u-\tau^*)] \\ =& \mathbb{E}[\Phi(G_{\tau^*}, R_{\tau^*}, Y_{\tau^*}, u-\tau^*)|G_0 = x, R_0 = \rho, Y_0 = y].\end{aligned} \tag{3.21}$$

Concerning the two cases for the time of the first jump, we can write:

$$\begin{aligned}\Phi(x,\rho,y,u) =& \mathbb{E}[\mathbb{1}\big(\inf_{r\leq u} G_r^{(x,\rho,y)} > 0\big)\mathbb{1}(\tau \leq u)] + \mathbb{E}[\mathbb{1}\big(\inf_{r\leq u} G_r^{(x,\rho,y)} > 0\big)\mathbb{1}(\tau > u)] \\ =& \mathbb{E}[\mathbb{1}\big(\inf_{r\leq u} G_r^{(x,\rho,y)} > 0\big)|\tau \leq u]\mathbb{P}(\tau \leq u) + e^{-\lambda u}Q(x,\rho,y,u).\end{aligned} \tag{3.22}$$

Then, using (3.21), the law of total probability and the definition of the $s-$operator the first term can be written as:

$$\begin{aligned}&\int_0^u \lambda e^{-\lambda s}\mathbb{E}[\Phi(G_s, R_s, Y_s, u-s)|G_0 = x, R_0 = \rho, Y_0 = y]ds \\ =& \int_0^u \lambda e^{-\lambda s}\int_0^\infty \int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z, R_s, Y_s, u-s)|R_0 = \rho, Y_0 = y]p(x', x, s)dF(z)dx'ds \\ =& \int_0^u \lambda e^{-\lambda s}\int_0^\infty \int_{\mathbb{R}} \mathcal{T}_s\Phi(x'+z, \rho, y, u)p(x', x, s)dF(z)dx'ds,\end{aligned} \tag{3.23}$$

where we have conditioned on the pre-jump asset value and the subsequent jump size. Substituting back into (3.22), we get the required result. □

It is now straightforward to obtain the analogous integral equations for the PD functions under the regime switching and stochastic volatility models. We present

both these results below, omitting the proofs for brevity, as they follow the proof of Proposition 3.4.2, almost identically.

Corollary 3.4.3.

(i) Consider the asset value process under the regime switching model (3.3) with jump rate and jump size distribution as in Proposition 3.4.2. Let $Q(x,\rho,u)$ and $p(x',x,s)$ be the survival probability and transition density, respectively, of the continuous regime switching OU process. Then, the survival probability $\Phi(x,\rho,u)$ satisfies the integral equation:

$$\Phi(x,\rho,u) = \int_0^u \lambda e^{-\lambda s} \int_0^\infty \int_{\mathbb{R}} \mathcal{T}_s \Phi(x'+z,\rho,u) p(x',x,s) dF(z) dx' ds + e^{-\lambda u} Q(x,\rho,u). \tag{3.24}$$

(ii) Consider the asset value process under the stochastic volatility model (3.5) with jump rate and jump size distribution as in Proposition 3.4.2. Let $Q(x,y,u)$ and $p(x',x,s)$ be the survival probability and transition density, respectively, of the continuous stochastic volatility OU process. Then, the survival probability $\Phi(x,y,u)$ satisfies the integral equation:

$$\Phi(x,y,u) = \int_0^u \lambda e^{-\lambda s} \int_0^\infty \int_{\mathbb{R}} \mathcal{T}_s \Phi(x'+z,y,u) p(x',x,s) dF(z) dx' ds + e^{-\lambda u} Q(x,y,u). \tag{3.25}$$

Remark 3.4.4. The proof above is based on the approach considered in Mishura and Ragulina, 2016 and constitutes an extension for processes where the diffusion term in non-zero. This results in having to consider appropriate stopping times, as well as the transition density of the OU process in the representation above. Furthermore, it is worth mentioning the simple case of the non-jump OU process, the survival probability $\Phi(x,u)$ satisfies the integral equation:

$$\Phi(x,u) = \int_0^u \lambda e^{-\lambda s} \int_0^\infty \int_{\mathbb{R}} \Phi(x'+z,u-s) p(x',x,s) dF(z) dx' ds + e^{-\lambda u} Q(x,u), \tag{3.26}$$

which can be easily derived from the regime switching model by considering a single regime. This can be directly compared to the corresponding integral equation in Mishura and Ragulina, 2016, where similar integral equations are used to study the ruin probability function for an analogous asset value process. On the other hand, in this work, the integral equations can lead to the existence and continuity results that suffice to obtain approximations of the PD functions, as we will see below.

Remark 3.4.5. In the following sections we will often interchange between using the $s-$operator formulation and more analytical expressions in terms of an appropriate expected value. In particular, applying the law of total probability, the integral equation for the stochastic volatility model can be written in more detail as:

$$\Phi(x,y,u) = \int_0^u \lambda e^{-\lambda s} \int_0^\infty \int_0^\infty \int_{\mathbb{R}} \Phi(x'+z,v,u-s) q(v,y,s) p(x',x,s) dF(z) dx' dv ds + e^{-\lambda u} Q(x,y,u), \tag{3.27}$$

where $q(\nu, y, s)$ is the transition probability of the CIR volatility process and we have used that, by definition:

$$\mathcal{T}_s\Phi(x'+z, y, u) = \int_0^\infty \Phi(x'+z, \nu, u-s)q(\nu, y, s)d\nu.$$

Similarly, for the generalized model, we can write:

$$\begin{aligned}&\Phi(x,\rho,y,u)\\ &\quad= \int_0^u \lambda e^{-\lambda s}\int_0^\infty\int_0^\infty\int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z, R_s^\rho, \nu, u-s)]q(\nu, y, s)p(x', x, s)dF(z)dx'd\nu ds\\ &\qquad + e^{-\lambda u}Q(x,\rho,y,u).\end{aligned} \tag{3.28}$$

These analytical versions will be useful when using the integral equations to derive PIDEs for the PD processes, whereas the more concise form will be used when proving the required continuity results that follow.

Remark 3.4.6. Finally, it is also worth noting that, in obtaining the integral equations in this section, we have included the initial values of the regime and volatility processes as additional variables on which the PD function depends. An alternative approach is writing these as a system of integral equations. For example, considering the regime switching model with states in accordance to the IFRS 9 framework (i.e., Stage 1, 2 and 3) we can rewrite $\Phi_i(x,u) := \Phi(x,\rho^i,u)$ (and similarly $Q_i(x,u) := Q(x,\rho^i,u)$), which results in the system of equations:

$$\begin{aligned}&\Phi_i(x,u)\\ &\quad= \int_0^u \lambda e^{-\lambda s}\int_0^\infty\int_{\mathbb{R}}\sum_{j=1}^3 \Phi_j(x, u-s)\pi(R_j,\rho^i,s)p(x', x, s)dF(z)dx'ds + e^{-\lambda u}Q_1(x,u),\end{aligned} \tag{3.29}$$

for $i = 1,2,3$, and where we have used that, in this case, $\mathcal{T}_s\Phi(x,\rho,u-s) = \sum_{j=1}^3 \Phi(x,R_j,u-s)\pi(R_j,\rho,s)$, by definition. In our approach, we prefer to account for these externals parameters explicitly, and generalize this representation in the cases of the additional variables.

3.4.3 Properties and existence of solutions

Using the IEs derived above, we can now prove that the PD functions enjoy certain mathematical properties required to obtain the viscosity solutions to the corresponding PIDEs.

To consider such solutions, we require Ψ (equivalently Φ) to be a continuous function of (x,u). To this end, we first note that Ψ (Φ) is a monotonically decreasing (increasing) function with respect to x and a monotonically increasing (decreasing) function with respect to maturity u. Moreover, as it is bounded, we can conclude that it is an integrable function. We can take advantage of the integral equation forms to prove that solutions for the survival probabilities do exist, and moreover, that they are continuous with respect to x and u. We prove this result for the generalized model, from which the other cases follow easily.

Remark 3.4.7. As mentioned, in what follows we will focus on the survival probability as a function of (x,u), as the additional complexity we are interested in arises from the jump component of the OU process. It is straightforward to reproduce

the proofs for the stochastic volatility variable, as well. Finally, the existence of the regime switching process simply creates a coupling that does not affect the results in this section.

Lemma 3.4.8. *The probability of survival function under the generalized model $\Phi(x,\rho,y,u)$, as defined in (3.11) is uniformly continuous as a function of x and u.*

Proof. To prove this result we will use the integral formulation (3.17). Consider a fixed $\epsilon > 0$ and $||(x,u)-(x_0,u_0)|| < \delta$, for some $\delta > 0$ that will be specified. Then:

$$\begin{aligned}&|\Phi(x,\rho,y,u)-\Phi(x_0,\rho,y,u_0)|\\&\leq |\Phi(x,\rho,y,u)-\Phi(x_0,\rho,y,u)|+|\Phi(x_0,\rho,y,u)-\Phi(x_0,\rho,y,u_0)|\end{aligned}$$

We will handle each of the terms above separately. We have:

x : Let $|x-x_0| < \delta_1$. Since the transition density of the non-jump OU process is uniformly continuous, we can select δ_1 such that $|p(x',x,s)-p(x',x_0,s)| < \epsilon/2$. Then:

$$\begin{aligned}&|\Phi(x,\rho,y,u)-\Phi(x_0,\rho,y,u)|\\&\leq \int_0^u \lambda e^{-\lambda s}\int_{\mathbb{R}}\int_{\mathbb{R}} \mathcal{T}_s\Phi(x'+z,\rho,y,u)|p(x',x,s)-p(x',x_0,s)|dF(z)dx'ds\\&\quad + e^{-\lambda u}|Q(x,u)-Q(x_0,u)|\\&\leq \int_0^u \lambda e^{-\lambda s}\int_{\mathbb{R}}\int_{\mathbb{R}} \mathcal{T}_s\Phi(x'+z,\rho,y,u)\frac{\epsilon}{2}dF(z)dx'ds + e^{-\lambda u}\frac{\epsilon}{2}\\&\quad \leq \int_0^u \lambda e^{-\lambda s}\frac{\epsilon}{2}ds + e^{-\lambda u}\frac{\epsilon}{2} = \frac{\epsilon}{2}.\end{aligned}$$

u : Similarly, let $|u-u_0| < \delta_2$. Then:

$$\begin{aligned}&|\Phi(x_0,\rho,y,u)-\Phi(x_0,\rho,y,u_0)| \leq\\&\leq \int_{u_0}^u \lambda e^{-\lambda s}\int_{\mathbb{R}}\int_{\mathbb{R}} \left|\mathcal{T}_s\Phi(x'+z,\rho,y,u)-\mathcal{T}_s\Phi(x'+z,\rho,y,u_0)\right| p(x',x,s)dF(z)dx'ds.\end{aligned}$$

By definition, we have $\|\mathcal{T}_s\| \leq 1$ and therefore: $|\Phi(x,\rho,y,u)-\Phi(x,\rho,y,u_0)| \leq \int_{u_0}^u \lambda e^{-\lambda s}ds \leq (u-u_0)\max_{s\in[0,u]}\lambda e^{-\lambda s} = \lambda(u-u_0) < \lambda\delta_2$.

By selecting $\delta_2 = \frac{\epsilon}{2\lambda}$ and $\delta = \delta_1 \wedge \delta_2$ we therefore obtain:

$$|\Phi(x,\rho,y,u)-\Phi(x_0,\rho,y,u_0)| < \epsilon,$$

as required.

□

We can now consider an appropriate fixed point result which will allow us to prove the existence of a solution to the IE for the survival probability. For this result we will refer to the Arzela-Ascoli and Schauder's fixed point theorems, as stated in B.3.1 and B.3.2 of the Appendix, respectively.

Proposition 3.4.9. *The integral equation (3.17) admits a continuous solution.*

Proof. Consider the metric space of all continuous functions $\Phi(\cdot,\rho,y,\cdot)$ on $\mathcal{D}\times[0,T]$, denoted by $X := C(\mathcal{D}\times[0,T])$. Furthermore, define the functional operator $\mathcal{A}$:

$X \to X$ by:

$$\mathcal{A}\Phi(x,\rho,y,u) = \int_{\mathbb{R}}\int_{\mathbb{R}} \mathcal{T}_s\Phi(x'+z,\rho,y,u)p(x',x,s)dF(z)dx'. \tag{3.30}$$

We begin by proving that the operator $\mathcal{A}$ is: (i) uniformly bounded, (ii) equi-continuous and (iii) compact. We separate these in the steps below:

(i) Uniform boundedness follows easily from the definition of Φ and the operator $\mathcal{T}_s$. Specifically:

$$|\mathcal{A}\Phi(x,\rho,y,u)| \leq 1.$$

(ii) For equi-continuity we must show that, given $\epsilon > 0$, there exists $\delta > 0$ such that if $||(x,u)-(x_0,u_0)|| < \delta$ then $|\mathcal{A}\Phi(x,\rho,y,u) - \mathcal{A}\Phi(x_0,\rho,y,u_0)| < \epsilon$, for all $\Phi \in X$. The proof follows very closely Lemma 3.4.8, but we include the steps for completeness.

To this end, we calculate:

$$\begin{aligned}&|\mathcal{A}\Phi(x,\rho,y,u) - \mathcal{A}\Phi(x_0,\rho,y,u_0)|\\ &\leq \int_{\mathbb{R}}\int_{\mathbb{R}} |\mathcal{T}_s\Phi(x'+z,\rho,y,u)p(x',x,s) - \mathcal{T}_s\Phi(x'+z,\rho,y,u_0)p(x',x_0,s)|dF(z)dx'.\end{aligned}$$

Consider the integrand. We can write:

$$\begin{aligned}&\mathcal{T}_s\Phi(x'+z,\rho,y,u)\Big(p(x',x,s) - p(x',x_0,s)\Big)\\ &+ p(x',x,s)\Big(\mathcal{T}_s\Phi(x'+z,\rho,y,u) - \mathcal{T}_s\Phi(x'+z,\rho,y,u_0)\Big),\end{aligned}$$

and therefore:

$$\begin{aligned}&|\mathcal{A}\Phi(x,\rho,y,u) - \mathcal{A}\Phi(x_0,\rho,y,u_0)| \leq \int_{\mathbb{R}}\int_{\mathbb{R}} \mathcal{T}_s\Phi(x'+z,\rho,y,u)|p(x',x,s) - p(x',x_0,s)|\\ &+ p(x',x_0,s)|\mathcal{T}_s\Phi(x'+z,\rho,y,u) - \mathcal{T}_s\Phi(x'+z,\rho,y,u_0)|F(z)dx'.\end{aligned}$$

We know that $p(x',x,s)$ is uniformly continuous in x and Φ is uniformly continuous in u. Therefore, given $\epsilon > 0$ we can select δ_1 such that

$$|p(x',x,s) - p(x',x_0,s)| < \epsilon/2,$$

for all x, x_0 such that $|x - x_0| < \delta_1$. Furthermore, from Lemma 3.4.8 recall that, for u, u_0 such that $|u - u_0| < \delta_2$ we have:

$$|\Phi(x'+z,\rho,y,u) - \Phi(x'+z,\rho,y,u_0)| < \lambda\delta_2,$$

for all $\Phi \in X$. Select $\delta_2 = \frac{\epsilon}{2\lambda}$ and let $\delta = \delta_1 \wedge \delta_2$. Hence:

$$\begin{aligned}&|\mathcal{A}\Phi(x,\rho,y,u) - \mathcal{A}\Phi(x_0,\rho,y,u_0)|\\ &\qquad\leq \int_{\mathbb{R}}\int_{\mathbb{R}} \mathcal{T}_s\Phi(x'+z,\rho,y,u)\frac{\epsilon}{2} + p(x',x_0,s)\lambda\delta dF(z)dx' \leq \epsilon.\end{aligned} \tag{3.31}$$

Notice that the choice of δ does not depend on Φ, only on the given ϵ. Hence, $|\mathcal{A}\Phi(x,\rho,y,u) - \mathcal{A}\Phi(x_0,\rho,y,u_0)| < \epsilon$, for all $\Phi \in X$ whenever $||(x,u) - (x_0,u_0)|| < \delta$, i.e., $\mathcal{A}$ is equi-continuous.

(*iii*) We have that $\mathcal{A}\Phi$ is uniformly bounded and equi-continuous and therefore compact by the Arzela-Ascoli theorem.

With the above, we now turn to the main result. Consider now the Banach space $\mathcal{C} = \{\phi \in X, ||\phi|| \leq 1\}$, where we have used the standard norm

$$||\phi|| := \sup_{x,u} |\phi|.$$

With the operator $\mathcal{A}$ defined as above, we have:

$$\Phi(x,\rho,y,u) = \int_0^u \lambda e^{-\lambda s}\mathcal{A}\Phi(x,\rho,y,u)ds + g(x,u),$$

with $g(x,u) := e^{-\lambda u}Q(x,\rho,y,u)$ and it is natural to define the operator $\mathcal{P}\Phi$, such that:

$$\mathcal{P}\Phi(x,\rho,y,u) = \int_0^u \lambda e^{-\lambda s}\mathcal{A}\Phi(x,\rho,y,u)ds + g(x,u). \tag{3.32}$$

Recall that $\mathcal{A}\Phi$ is bounded by 1 and, furthermore, by the definition of the operator, we also have that:

$$||\mathcal{A}\Phi|| \leq ||\Phi||.$$

Hence:

$$\begin{aligned}||\mathcal{P}\Phi|| &\leq ||\int_0^u \lambda e^{-\lambda s}\mathcal{A}\Phi(x,\rho,y,u)ds|| + ||g|| \leq \sup_{x,u}\int_0^u \lambda e^{-\lambda s}|\mathcal{A}\Phi(x,\rho,y,u)|ds + ||g|| \\ &\leq \sup_{x,u}(1-e^{-\lambda u})|\Phi| + \sup_{x,u} e^{-\lambda u}|Q(x,\rho,y,u)| \leq \max(||\Phi||, ||g||) \leq 1, \end{aligned} \tag{3.33}$$

concluding that $\mathcal{P}$ maps function $\Phi \in X$ to X. Notice that we can write $\mathcal{P} = \mathcal{J}\mathcal{A}$, with $\mathcal{J}\phi = \int_0^u \lambda e^{-\lambda s}\phi(s)ds$. It is straightforward to see that the linear operator $\mathcal{J}$ is compact, as is $\mathcal{A}$, as shown in Lemma 3.4.8. Therefore, we can apply Schauder's fixed point theorem to conclude that $\mathcal{P}$ has a fixed point in X, which solves the integral equation (3.17). □

3.5 Partial Integro-Differential Equations for the PD function

Generally, when considering various credit modelling tasks, such as forecasting probability of default and expected losses, it is common throughout the literature to use path simulation techniques, particularly for practical purposes. Specific examples and applications can be seen in e.g., Sak and Hörmann, 2012 and Virolainen, 2004. On the other hand, we will see in this section that the integral equation representations (3.24), (3.25) and (3.28) lead to PIDEs for the PD functions, which belong to families of well-studied equations. Hence, our approach relies solely on the equations derived in this and the previous section, which can be solved to retrieve the corresponding values, thereby eliminating the need for simulations and the larger errors which accompany such methods.

Natural questions arise related to the regularity conditions that the survival function (and hence the corresponding PD function) must satisfy; for example, in the one dimensional Lévy-driven OU case, classical solutions of PIDEs would require that $\Phi(x,u) \in \mathcal{C}^{2,1}\big(\mathcal{D}\times[0,T]\big)$, where $\mathcal{D} := [0,\infty)$, i.e., the survival probability function

would have to be twice and once continuously differentiable in x and u, respectively, on the corresponding domains. In many cases, the required differentiability conditions are often assumed. However, we can avoid making such assumptions by considering viscosity solutions of the PIDEs, a notion introduced in Crandall and Lions, 1983. Viscosity solutions and their applications in finance have been studied in e.g., Cont and Voltchkova, 2005b and Cont and Voltchkova, 2005a. We begin by showing that the generalized PD function is a viscosity solution of a PIDE that will be derived. Then, we continue by showing that this and the PIDEs that result from the regime switching and stochastic volatility models can be derived directly from the corresponding integral equations, if the required regularity conditions hold. In our setting, it is understood that the survival functions are in fact viscosity solutions to these PIDEs, yet these calculations are important for two reasons: firstly, they establish the connection between the integral equations and the corresponding PIDEs, and secondly they constitute an efficient method of obtaining the form of the PIDEs, for which we can then show that the survival functions are indeed viscosity solutions.

Finally, it is worth emphasizing the utility of the PIDEs obtained in this section. Specifically, the solutions to these equations are PD values across both initial positions, time and latent variables. Hence, we will see that, by considering numerical schemes for the PIDES, we obtain the complete evolution of the PD process that is required for applications in credit risk modelling. This is clearly preferable to common methods such as Monte Carlo estimations of the PD, where one must perform simulations of the underlying asset process for many different initial positions and time horizons, separately. This process is extremely computationally costly, especially when taking into account the order of convergence of many stochastic simulation schemes. For example, the Euler scheme for the simulation of the asset process we consider has a strong convergence of order 0.5, which is required since the PD value depends on the whole path of the asset process.

3.5.1 Viscosity solutions

Viscosity solutions for non-local PDEs have also been studied in e.g., Barles and Imbert, 2008, Cont and Voltchkova, 2005b and Hamadène and Morlais, 2016 and references therein. We reiterate the importance of this approach: requiring only continuity of the underlying function, which has been proven for the survival function under the proposed models, we can define solutions of the equations in a weak sense and subsequently approximate them using numerical schemes. We begin by showing that the generalized survival probability is a viscosity solution of an appropriate PIDE (provided in equation (3.34) below). For completeness, we include the definition of a viscosity solution below (altered to reflect the arguments of the survival function we are studying).

Definition 3.5.1. Consider an integro-differential operator for the function with arguments as above, $\mathcal{L}f(x,\rho,y,u)$ and a corresponding PIDE $\mathcal{L}f(x,\rho,y,u) = 0$. Then, a function $\phi(x,\rho,y,u)$ is called a viscosity supersolution (subsolution) of the PIDE if, for any $\rho \in \mathcal{R}$, for every $(x,y,u) \in \mathcal{D}\times\mathcal{V}\times[0,T]$, and every function $f(\cdot,\rho,\cdot,\cdot) \in \mathcal{C}^{2,1}\left(\tilde{\mathcal{D}}\times[0,T]\right)$, where $\tilde{D} := \mathcal{D}\times\mathcal{V}$, such that $\phi(x,\rho,y,u) = f(x,\rho,y,u)$ and $\phi \geq f$ ($\phi \leq f$), the inequality $\mathcal{L}f(x,\rho,y,u) \leq 0$ ($\mathcal{L}f(x,\rho,y,u) \geq 0$) holds. A function $\phi(x,\rho,y,u)$ is a viscosity solution of the PIDE if ϕ is simultaneously a viscosity supersolution and subsolution.

In what follows we will show that the survival function is a viscosity solution of an appropriate PIDE given below. We must first show that viscosity solutions do exist for the PIDEs in question. The proof that follows is based on the corresponding result in Belkina and Kabanov, 2016, extended to match the models studied in this work.

Proposition 3.5.2. *The survival probability function $\Phi(x,\rho,y,u)$ is a viscosity solution of the PIDE $\mathcal{L}f(x,\rho,y,u)=0$, where:*

$$\begin{aligned}\mathcal{L}f(x,\rho,y,u) := &-\frac{\partial f}{\partial u}(x,\rho,y,u) + k_\rho(\theta_\rho - x)\frac{\partial f}{\partial x}(x,\rho,y,u) + \kappa(\mu-y)\frac{\partial f}{\partial y}(x,\rho,y,u)\\ &+\frac{1}{2}\sigma_\rho^2 y\frac{\partial^2 f}{\partial x^2}(x,\rho,y,u) + \frac{1}{2}\xi^2 y\frac{\partial^2 f}{\partial y^2}(x,\rho,y,u) + \sum_{j\neq\rho} q_{\rho j}\Big(f(x,j,y,u) - f(x,\rho,y,u)\Big)\\ &+\int_{\mathbb{R}}\Big(f(x+z,\rho,y,u) - f(x,\rho,y,u)\Big)\nu(dz). \end{aligned} \tag{3.34}$$

Proof. We must show that $\Phi(x,\rho,y,u)$ is simultaneously a viscosity supersolution and subsolution of the PIDE.

We begin with the supersolution case. For any $\rho\in\mathcal{R}$, consider fixed $(x,y,u)\in\mathcal{D}\times\mathcal{V}\times[0,T]$, with $\Phi(x,\rho,y,u)=0$ when $x\leq 0$, by definition. Furthermore, consider a function $f(\cdot,\rho,\cdot,\cdot)\in\mathcal{C}^{2,1}\big(\tilde{D}\times[0,T]\big)$ such that $\Phi(x,\rho,y,u)=f(x,\rho,y,u)$ and $\Phi((\cdot,\rho,\cdot,\cdot))\leq f(\cdot,\rho,\cdot,\cdot)$ on $\mathcal{D}\times\mathcal{V}\times[0,T]$. Now, let $h>0$ and let ϵ_x,ϵ_y be small enough to ensure that $f\in\mathcal{C}^{2,1}\Big(\tilde{d}_\epsilon\times[0,T]\Big)$, where $\tilde{d}_\epsilon := B_{\epsilon_x}(x)\times B_{\epsilon_y}(y)$, i.e., we are considering a neighborhood of the fixed (x,u) where the functions will "touch". Finally, define the stopping time $\tau_h := \inf\{t\geq 0 : (G_t^x, Y_t^x)\notin(\overline{B_{\epsilon_x}(x)}\times\overline{B_{\epsilon_y}(y)})\}\wedge h$, noting that we choose $h<u$ to ensure that $\tau_h<u$. Then, by the Itô formula, and with the operator $\mathcal{L}$ is in (3.34), we have:

$$\begin{aligned} f(G^x_{\tau_h},R^\rho_{\tau_h},Y^y_{\tau_h},u-\tau_h) - f(G^x_0,R^\rho_0,Y^y_0,u) &= f(G^x_{\tau_h},R^\rho_{\tau_h},Y^y_{\tau_h},u-\tau_h) - f(x,\rho,y,u)\\ &= \int_0^{\tau_h}\mathcal{A}f(G^x_t,R^\rho_t,Y^y_t,u-t)dt + \int_0^{\tau_h}\sigma\frac{\partial f}{\partial x}(G^x_t,R^\rho_t,Y^y_t,u-t)dB_t\\ &+\int_0^{\tau_h}\int_{\mathbb{R}}\Big(f(G^x_t+z,R^\rho_t,Y^y_t,u-t) - f(G^x_t,R^\rho_t,Y^y_t,u-t)\Big)\tilde{N}(dt,dz)\\ &+\sum_{j\neq\rho}p_{\rho j}(t_h)\Big(f(G^x_{\tau_h},j,Y^y_{\tau_h},u-\tau_h) - f(G^x_{\tau_h},\rho,Y^y_{\tau_h},u-\tau_h)\Big)\\ &= \int_0^{\tau_h}\mathcal{A}f(G^x_t,R^\rho_t,Y^y_t,u-t)dt + m_t, \end{aligned} \tag{3.35}$$

where

$$\begin{aligned} m_t := &\int_0^{\tau_h}\sigma\frac{\partial f}{\partial x}(G^x_t,R^\rho_t,Y^y_t,u-t)dB_t\\ &+\int_0^{\tau_h}\int_{\mathbb{R}}\Big(f(G^x_t+z,R^\rho_t,Y^y_t,u-t) - f(G^x_t,R^\rho_t,Y^y_t,u-t)\Big)\tilde{N}(dt,dz) \end{aligned} \tag{3.36}$$

is a martingale and therefore so is the stopped process $m_{t\wedge\tau_h}$.

We have that $f(x,\rho,y,u) = \Phi(x,\rho,y,u)$ and recall that $\Phi(x,\rho,y,u) = \mathbb{E}[\Phi(G^x_{\tau_h}, R^\rho_{\tau_h}, Y^y_{\tau_h}, u-\tau_h)]$ almost surely, from Proposition 3.4.2. Therefore:

$$\Phi(G^x_{\tau_h}, R^\rho_{\tau_h}, Y^y_{\tau_h}, u-\tau_h) \geq$$
$$f(G^x_{\tau_h}, R^\rho_{\tau_h}, Y^y_{\tau_h}, u-\tau_h) = f(x,\rho,y,u) + \int_0^{\tau_h} \mathcal{A}f(G^x_t, R^\rho_t, Y^y_t, u-t)dt + m_t, \quad (3.37)$$

and so:

$$\mathbb{E}[\Phi(G^x_{\tau_h}, R^\rho_{\tau_h}, Y^y_{\tau_h}, u-\tau_h)] \geq \mathbb{E}[\Phi(x,\rho,y,u)] + \mathbb{E}\Big[\int_0^{\tau_h} \mathcal{A}f(G^x_t, R^\rho_t, Y^y_t, u-t)dt\Big] + \mathbb{E}[m_t]$$
$$\Rightarrow \Phi(x,\rho,y,,u) \geq \Phi(x,\rho,y,u) + \mathbb{E}\Big[\int_0^{\tau_h} \mathcal{A}f(G^x_t, R^\rho_t, Y^y_t, u-t)dt\Big], \quad (3.38)$$

and hence $\mathbb{E}\Big[\int_0^{\tau_h} \mathcal{A}f(G^x_t, R^\rho_t, Y^y_t, u-t)dt\Big] \leq 0$. The final step is as in Belkina and Kabanov, 2016; when h is sufficiently small we have $\tau_h = h$ and therefore, by the Lebesgue dominated convergence theorem we have:

$$\mathcal{A}f(x,\rho,y,u) = \lim_{h\downarrow 0} \frac{1}{h}\mathbb{E}\Big[\int_0^{\tau_h} \mathcal{A}f(G^x_t, R^\rho_t, Y^y_t, u-t)dt\Big] \leq 0, \quad (3.39)$$

showing that Φ is indeed a viscosity supersolution. It follows directly, by switching the inequalities in the steps above, that Φ is a viscosity subsolution and therefore the result is proven. □

Remark 3.5.3. Viscosity theory is not the only prism under which we can consider weak solutions. We can also study solutions in appropriate Sobolev spaces, for which we will need to define a notion of weak differentiability, and we can then use standard martingale approaches to obtain the corresponding PIDEs. We include details of this approach in Appendix B.2.

Remark 3.5.4. We furthermore note that if additional conditions hold, then the strong solutions that occur are equal to the viscosity solutions, as expected. The formulations via viscosity solutions are useful to obtain the form of the PIDEs the PD functions satisfy, and we will see that we can then consider additional conditions that lead to regular solutions through these equations.

3.5.2 The survival probability as a classical solution of PIDEs derived from the IE formulations

To build a consistent framework we must ensure that that the PIDEs obtained above (which are satisfied by the PD functions in the viscosity sense) are derivable from the integral equations formulation in section 3.4. In this section, we show that the integral equations indeed lead to the corresponding PIDEs. We will begin with the calculations under the regime switching and stochastic volatility models, upon which the corresponding result under the generalized model will be built. In all the results below, we consider a fixed time horizon $T > 0$.

Lemma 3.5.5.

(*i*) *Under the regime-switching model (3.3), the survival probability* $\Phi(x,\rho,u)$ *satisfies the PIDE:*

$$\frac{\partial\Phi}{\partial u}(x,\rho,u) = k_\rho(\theta_\rho - x)\frac{\partial\Phi}{\partial x}(x,\rho,u) + \frac{1}{2}\sigma_\rho^2\frac{\partial^2\Phi}{\partial x^2}(x,\rho,u) + \sum_{j\neq\rho} q_{\rho j}\Big(\Phi(x,j,u) - \Phi(x,\rho,u)\Big)$$
$$+ \int_{\mathbb{R}}\Big(\Phi(x+z,\rho,u) - \Phi(x,\rho,u)\Big)\nu(dz), \quad (x,\rho,u)\in\mathcal{D}\times\mathcal{R}\times[0,T], \tag{3.40}$$

with initial and boundary conditions:

$$\begin{aligned}\Phi(x,\rho,0) &= \mathbb{1}_{\{x>0\}}, \quad (x,\rho)\in\mathcal{D}\times\mathcal{R},\\ \Phi(0,\rho,u) &= 0, \quad (\rho,u)\in\mathcal{R}\times[0,T],\\ \Phi(x,\rho,u) &\to 1 \text{ as } x\to\infty, \quad (\rho,u)\in\mathcal{R}\times[0,T],\end{aligned}$$

where q_{ij} *are the elements of the generator* Q *of the switching pocess* R_t*, as defined in (1.2).*

(*ii*) *Under the stochastic volatility model (3.5), the survival probability* $\Phi(x,y,u)$ *satisfies the PIDE:*

$$\frac{\partial\Phi}{\partial u}(x,y,u)$$
$$= k(\theta - x)\frac{\partial\Phi}{\partial x}(x,y,u) + \kappa(\mu - y)\frac{\partial\Phi}{\partial y}(x,y,u) + \frac{1}{2}y\frac{\partial^2\Phi}{\partial x^2}(x,y,u) + \frac{1}{2}\xi^2 y\frac{\partial^2\Phi}{\partial y^2}(x,y,u)$$
$$+ \int_{\mathbb{R}}\Big(\Phi(x+z,y,u) - \Phi(x,y,u)\Big)\nu(dz), \quad (x,y,u)\in\mathcal{D}\times\mathcal{V}\times[0,T], \tag{3.41}$$

subject to the initial and boundary conditions:

$$\begin{aligned}\Phi(x,y,0) &= \mathbb{1}_{\{x>0\}}, \quad (x,y)\in\mathcal{D}\times\mathcal{V},\\ \Phi(0,y,u) &= 0, \quad (y,u)\in\mathcal{V}\times[0,T],\\ \Phi(x,y,u) &\to 1 \text{ as } x\to\infty, \quad (y,u)\in\mathcal{V}\times[0,T],\\ \frac{\partial\Phi}{\partial y}(x,y,u) &= 0 \text{ as } y\to\infty \quad (x,u)\in\mathcal{D}\times[0,T].\end{aligned} \tag{3.42}$$

Proof. (*i*) Our approach relies on taking advantage of the known fact that the transition densities satisfy the Kolmogorov equation. Specifically, we know that for $p(\cdot,x,u)$ we have:

$$\frac{\partial p}{\partial u}(\cdot,x,u) = k_\rho(\theta_\rho - x)\frac{\partial p}{\partial x}(\cdot,x,u) + \frac{1}{2}\sigma_\rho^2\frac{\partial^2 p}{\partial x^2}(\cdot,x,u). \tag{3.43}$$

Recall that the same holds for the survival distribution of the continuous OU, $Q(x,\rho,u)$:

$$\frac{\partial Q}{\partial u}(x,\rho,u) = \mathcal{L}_1 Q(x,\rho,u), \tag{3.44}$$

with the generator operator under the regime switching model $\mathcal{L}_1$ given by:

$$\mathcal{L}_1 Q(x,\rho,t) := \\ k_\rho(\theta_\rho - x)\frac{\partial f}{\partial x}(x,\rho,t) + \frac{1}{2}\sigma_\rho^2\frac{\partial^2 f}{\partial x^2}(x,\rho,t) + \sum_{j\neq\rho} q_{\rho j}\Big(Q(x,j,t) - f(x,\rho,t)\Big) \tag{3.45}$$

Furthermore, for the function $g(\rho,u) := \mathbb{E}[\Phi(\cdot,R_u,\cdot)|R_0=\rho]$, we have that:

$$\frac{\partial g}{\partial u}(\rho,u) = \sum_{j\neq\rho} q_{\rho j}\big(g(j,u) - g(\rho,u)\big). \tag{3.46}$$

We now begin the calculations for the PIDE by making the change of variables $t := u - s$:

$$\Phi(x,\rho,u) = \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_{\mathbb{R}} \mathcal{T}_{u-t}\Phi(x'+z,\rho,u)p(x',x,u-t)dF(z)dx'dt \\ + e^{-\lambda u}Q(x,\rho,u), \tag{3.47}$$

By the Leibniz rule, and substituting in the definition of the operator $\mathcal{T}_s$, we then have:

$$\frac{\partial\Phi}{\partial u}(x,\rho,u) = \lambda\int_0^\infty\int_{\mathbb{R}}\mathbb{E}[\Phi(x'+z,R_0^\rho,u)]p(x',x,0)dF(z)dx' \\ + \int_0^u -\lambda^2 e^{-\lambda(u-t)}\int_0^\infty\int_{\mathbb{R}}\mathbb{E}[\Phi(x'+z,R_{u-t}^\rho,t)]p(x',x,u-t)dF(z)dx'dt \\ + \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_{\mathbb{R}}\mathbb{E}[\Phi(x'+z,R_{u-t}^\rho,t)]\frac{\partial p}{\partial u}(x',x,u-t)dF(z)dx'dt \\ + \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_{\mathbb{R}}\frac{\partial}{\partial u}\mathbb{E}[\Phi(x'+z,R_{u-t}^\rho,t)]p(x',x,u-t)dF(z)dx'dt \\ -\lambda e^{-\lambda u}Q(x,\rho,u) + e^{-\lambda u}\frac{\partial Q}{\partial u}(x,\rho,u). \tag{3.48}$$

The first term in the expression above can be written as:

$$\lambda\int_0^\infty\int_{\mathbb{R}}\mathbb{E}[\Phi(x'+z,\rho,u)]\delta_x(x')dF(z)dx' = \lambda\int_{\mathbb{R}}\Phi(x+z,\rho,u)dF(z).$$

On the other hand, using the integral equation for $\Phi(x,\rho,u)$, the second term can be written as:

$$-\lambda\Phi(x,\rho,u) + \lambda e^{-\lambda u}Q(x,\rho,u).$$

Combining, we obtain:

$$\frac{\partial\Phi}{\partial u}(x,\rho,u) = \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_{\mathbb{R}}\mathbb{E}[\Phi(x'+z,R_{u-t}^\rho,t)]\frac{\partial p}{\partial u}(x',x,u-t)dF(z)dx'dt \\ + \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_{\mathbb{R}}\frac{\partial}{\partial u}\mathbb{E}[\Phi(x'+z,R_{u-t}^\rho,t)]p(x',x,u-t)dF(z)dx'dt \\ + \lambda\int_{\mathbb{R}}\Phi(x+z,\rho,u)dF(z) - \lambda\Phi(x,\rho,u) + e^{-\lambda u}\frac{\partial Q}{\partial u}(x,\rho,u). \tag{3.49}$$

We now consider the generator of $\Phi(x,\rho,u)$. It is straightforward to separate

the components of $\mathcal{L}_1\Phi(x,\rho,u)$ since only the transition density $p(x',x,u-t)$ depends on x and only $\mathbb{E}[\Phi(x'+z,R^{\rho}_{u-t},t)]$ depends on the regime. We then have:

$$\begin{aligned}
&\mathcal{L}_1\Phi(x,\rho,u) \equiv k_\rho(\theta_\rho - x)\frac{\partial\Phi}{\partial x}(x,\rho,u) + \frac{1}{2}\sigma_\rho^2\frac{\partial^2\Phi}{\partial x^2}(x,\rho,u) + \sum_{j\neq\rho} q_{\rho j}\Big(\Phi(x,j,u) - \Phi(x,\rho,u)\Big)\\
&= e^{-\lambda u}\mathcal{L}_1 Q(x,\rho,u) + \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z,R^{\rho}_{u-t},t)]\mathcal{L}p(x',x,u-t)dF(z)dx'dt\\
&+ \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_{\mathbb{R}} \sum_{j\neq\rho} q_{\rho j}\Big(\mathbb{E}[\Phi(x'+z,R^{j}_{u-t},t)]\\
&- \mathbb{E}[\Phi(x'+z,R^{\rho}_{u-t},t)]\Big)p(x',x,u-t)dF(z)dx'dt\\
&= e^{-\lambda u}\frac{\partial Q}{\partial u}(x,\rho,u) + \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z,R^{\rho}_{u-t},t)]\frac{\partial p}{\partial u}(x',x,u-t)dF(z)dx'dt\\
&+ \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_{\mathbb{R}} \frac{\partial}{\partial u}\mathbb{E}[\Phi(x'+z,R^{\rho}_{u-t},t)]p(x',x,u-t)dF(z)dx'dt,
\end{aligned} \tag{3.50}$$

where we have used the fact that $Q(x,\rho,u)$ and $p(x',x,u)$ satisfy (3.43) and (3.44), respectively. Using (3.49), we obtain:

$$\mathcal{L}_1\Phi(x,\rho,u) = \frac{\partial\Phi}{\partial u}(x,\rho,u) - \lambda\Big(\int_{\mathbb{R}}\Phi(x+z,\rho,u)dF(z) - \Phi(x,\rho,u)\Big),$$

which, upon rearranging, can be written as:

$$\begin{aligned}
\frac{\partial\Phi}{\partial u}(x,\rho,u) &= k_\rho(\theta_\rho - x)\frac{\partial\Phi}{\partial x}(x,\rho,u) + \frac{1}{2}\sigma_\rho^2\frac{\partial^2\Phi}{\partial x^2}(x,\rho,u)\\
&+ \sum_{j\neq\rho} q_{\rho j}\Big(\Phi(x,j,u) - \Phi(x,\rho,u)\Big) + \int_{\mathbb{R}}\Big(\Phi(x+z,\rho,u) - \Phi(x,\rho,u)\Big)\nu(dz).
\end{aligned} \tag{3.51}$$

(ii) The proof follows as above. Under model (3.5), for $p(\cdot,x,u)$ we now have:

$$\frac{\partial p}{\partial t}(\cdot,x,u) = k(\theta - x)\frac{\partial p}{\partial x}(\cdot,x,u) + \frac{1}{2}y\frac{\partial^2 p}{\partial x^2}(\cdot,x,u). \tag{3.52}$$

In this case, we also have to account for the transition density of the underlying volatility process, $q(\cdot,y,u)$, which satisfies:

$$\frac{\partial q}{\partial u}(\cdot,y,u) = \kappa(\mu - y)\frac{\partial q}{\partial y}(\cdot,y,u) + \frac{1}{2}\xi^2 y\frac{\partial^2 q}{\partial y^2}(\cdot,y,u), \tag{3.53}$$

and recall that for $Q(x,y,u)$ we have:

$$\frac{\partial Q}{\partial u}(x,y,u) = \mathcal{L}_2 Q(x,y,u), \tag{3.54}$$

with the generator under the stochastic volatility model $\mathcal{L}_2$ given by:

$$\mathcal{L}_2 f(x,y,t) := k(\theta - x)\frac{\partial f}{\partial x}(x,y,t) + \kappa(\mu - y)\frac{\partial f}{\partial y}(x,y,t) + \frac{1}{2}y\frac{\partial^2 f}{\partial x^2}(x,y,t) + \frac{1}{2}\xi^2 y\frac{\partial^2 Q}{\partial y^2}(x,y,t) \tag{3.55}$$

Differentiating (3.27) with respect to x and u, and comparing $\mathcal{L}_2\Phi(x,y,u)$ with $\frac{\partial\Phi}{\partial u}(x,y,u)$ we obtain the required PIDE, using the same steps as in (i).

□

We can now present the main result for the generalized model. Even though the steps are similar as the cases above, the dependence on both the regime and volatility processes creates additional terms and it is therefore worth outlining the proof in detail.

Theorem 3.5.6. *Under the generalized asset process model (3.7) the survival probability* $\Phi(x,\rho,y,u)$ *satisfies the PIDE:*

$$\begin{aligned}
&\frac{\partial\Phi}{\partial u}(x,\rho,y,u)\\
&= k_\rho(\theta_\rho - x)\frac{\partial\Phi}{\partial x}(x,\rho,y,u) + \kappa(\mu - y)\frac{\partial\Phi}{\partial y}(x,\rho,y,u) + \frac{1}{2}\sigma_\rho^2 y\frac{\partial^2\Phi}{\partial x^2}(x,\rho,y,u) + \frac{1}{2}\xi^2 y\frac{\partial^2\Phi}{\partial y^2}(x,\rho,y,u)\\
&+ \sum_{j\neq\rho} q_{\rho j}\Big(\Phi(x,j,y,u) - \Phi(x,\rho,y,u)\Big) + \int_{\mathbb{R}}\Big(\Phi(x+z,\rho,y,u) - \Phi(x,\rho,y,u)\Big)\nu(dz),
\end{aligned} \tag{3.56}$$

for $(x,\rho,y,u) \in \mathcal{D}\times\mathcal{R}\times\mathcal{V}\times[0,T]$, *with initial and boundary conditions:*

$$\begin{aligned}
\Phi(x,\rho,y,0) &= \mathbb{1}_{\{x>0\}}, \quad (x,\rho,y) \in \mathcal{D}\times\mathcal{R}\times\mathcal{V},\\
\Phi(0,\rho,y,u) &= 0, \quad (\rho,y,u) \in \mathcal{R}\times\mathcal{V}\times[0,T],\\
\Phi(x,\rho,y,u) &\to 1, \text{ as } x\to\infty, \quad (\rho,y,u) \in \mathcal{R}\times\mathcal{V}\times[0,T],\\
\frac{\partial\Phi}{\partial y}(x,y,u) &= 0, \text{ as } y\to\infty \quad (x,\rho,u) \in \mathcal{D}\times\mathcal{R}\times[0,T].
\end{aligned} \tag{3.57}$$

Proof. For the transition density $p(\cdot,x,u)$ we have:

$$\frac{\partial p}{\partial t}(\cdot,x,u) = k_\rho(\theta - x)\frac{\partial p}{\partial x}(\cdot,x,u) + \frac{1}{2}\sigma_\rho^2 y\frac{\partial^2 p}{\partial x^2}(\cdot,x,u), \tag{3.58}$$

and for $q(\cdot,y,u)$ and $g(\rho,u) := \mathbb{E}[\Phi(\cdot,R_u^\rho,\cdot,\cdot)]$ we know that (3.53) and (3.46) hold, respectively. In this case, for $Q(x,\rho,y,u)$ we have:

$$\frac{\partial Q}{\partial u}(x,\rho,y,u) = \mathcal{L}_3 Q(x,\rho,y,u), \tag{3.59}$$

with the generator under the generalized model, $\mathcal{L}_3$, given by:

$$\mathcal{L}_3 f(x,\rho,y,u) := k_\rho(\theta_\rho - x)\frac{\partial f}{\partial x}(x,\rho,y,t) + \kappa(\mu - y)\frac{\partial f}{\partial y}(x,\rho,y,t) + \frac{1}{2}\sigma_\rho^2 y\frac{\partial^2 f}{\partial x^2}(x,\rho,y,t) + \frac{1}{2}\xi^2 y\frac{\partial^2 f}{\partial y^2}(x,\rho,y,t) + \sum_{j\neq\rho} q_{\rho j}\Big(f(x,j,y,t) - f(x,\rho,y,t)\Big), \tag{3.60}$$

As in the results above it will be useful to work with the definition of $\mathcal{T}_s$, i.e., version (3.25) of the integral equation. With the change of variables $t = u - s$ and applying the Leibniz rule we now get:

$$\begin{aligned}
\frac{\partial \Phi}{\partial u}(x,\rho,u) &= \lambda\int_0^\infty\int_0^\infty\int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z,R_0^\rho,\nu,u)]q(\nu,y,0)p(x',x,0)dF(z)dx'd\nu \\
&+ \int_0^u -\lambda^2 e^{-\lambda(u-t)}\int_0^\infty\int_0^\infty\int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z,R_{u-t}^\rho,\nu,t)]q(\nu,y,u-t)p(x',x,u-t)dF(z)dx'd\nu dt \\
&+ \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_0^\infty\int_{\mathbb{R}} \frac{\partial}{\partial u}\mathbb{E}[\Phi(x'+z,R_{u-t}^\rho,\nu,t)]q(\nu,y,u-t)p(x',x,u-t)dF(z)dx'd\nu dt \\
&+ \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_0^\infty\int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z,R_{u-t}^\rho,\nu,t)]\frac{\partial q}{\partial u}(\nu,y,u-t)p(x',x,u-t)dF(z)dx'd\nu dt \\
&+ \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_0^\infty\int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z,R_{u-t}^\rho,\nu,t)]q(\nu,y,u-t)\frac{\partial p}{\partial u}(x',x,u-t)dF(z)dx'd\nu dt \\
&- \lambda e^{-\lambda u}Q(x,\rho,y,u) + e^{-\lambda u}\frac{\partial Q}{\partial u}(x,\rho,y,u).
\end{aligned} \tag{3.61}$$

From the first term we have:

$$\lambda\int_0^\infty\int_0^\infty\int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z,\rho,\nu,u)]\delta_y(\nu)\delta_x(x')dF(z)dx'd\nu = \lambda\int_{\mathbb{R}}\Phi(x+z,\rho,y,u)dF(z),$$

whereas, from (3.28), the second term can be written as $-\lambda\Phi(x,\rho,y,u) + \lambda e^{-\lambda u}Q(x,\rho,y,u)$, and hence, we have:

$$\begin{aligned}
\frac{\partial \Phi}{\partial u}(x,\rho,y,u) &= \lambda\int_{\mathbb{R}}\Phi(x+z,\rho,y,u)dF(z) - \lambda\Phi(x,\rho,y,u) + e^{-\lambda u}\frac{\partial Q}{\partial u}(x,\rho,y,u) \\
&+ \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_0^\infty\int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z,R_{u-t},\nu,t)]\frac{\partial q}{\partial u}(\nu,y,u-t)p(x',x,u-t)dF(z)dx'd\nu dt \\
&+ \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_0^\infty\int_{\mathbb{R}} \mathbb{E}[\Phi(x'+z,R_{u-t},\nu,t)]q(\nu,y,u-t)\frac{\partial p}{\partial u}(x',x,u-t)dF(z)dx'd\nu dt \\
&+ \int_0^u \lambda e^{-\lambda(u-t)}\int_0^\infty\int_0^\infty\int_{\mathbb{R}} \frac{\partial}{\partial u}\mathbb{E}[\Phi(x'+z,R_{u-t},\nu,t)]q(\nu,y,u-t)p(x',x,u-t)dF(z)dx'd\nu dt.
\end{aligned} \tag{3.62}$$

Taking the derivatives of $\Phi(x,\rho,y,u)$ with respect to x and y is straightforward. Therefore, using (3.62) with (3.58), (3.53) and (3.46), we obtain:

$$\mathcal{L}_3\Phi(x,\rho,y,u) = \frac{\partial\Phi}{\partial u}(x,\rho,y,u) - \lambda\Big(\int_{\mathbb{R}}\Phi(x+z,\rho,y,u)dF(z) - \Phi(x,\rho,y,u)\Big),$$

and the expected PIDE follows. □

It is easy to see that the same steps can be used to obtain an equivalent PIDE for the simple PD function (corresponding to a regime switching model with one

regime), whose integral equation representation is given by (3.26). The PIDE is shown below; we omit the proof for brevity, as it follows directly as a special case of the results above.

Corollary 3.5.7. *Under model (1.14) the survival probability $\Phi(x,u)$ satisfies the PIDE:*

$$\frac{\partial \Phi}{\partial u} = k(\theta - x)\frac{\partial \Phi}{\partial x} + \frac{1}{2}\sigma^2\frac{\partial^2 \Phi}{\partial x^2} + \int_{\mathbb{R}} \Big(\Phi(x+z,u) - \Phi(x,u)\Big)\nu(dz) = 0, \quad (x,u) \in \mathcal{D} \times [0,T], \tag{3.63}$$

with initial and boundary conditions:

$$\begin{aligned} \Phi(x,0) &= \mathbb{1}_{\{x>0\}}, \quad x \in \mathcal{D}, \\ \Phi(0,u) &= 0, \quad u \in [0,T], \\ \Phi(x,u) &\to 1 \text{ as } x \to \infty, \quad u \in [0,T]. \end{aligned} \tag{3.64}$$

Remark 3.5.8. It is worth noting that the approach we have developed results in PIDEs consistent with the ruin probability examined in Mishura and Ragulina, 2016, where the asset process is given by:

$$X_t(x) = x + \int_0^t r(X_s(x) + c)ds + \sum_{i=1}^{N_t}(-Y_i),$$

with jump distribution $f(y)$ on $\mathbb{R}_+$ and jump intensity λ. It is shown that the survival probability $\phi(x,t) = \mathbb{P}\Big(\inf_{r\le t} X_r(x) > 0\Big)$ satisfies the PIDE:

$$\frac{\partial \phi}{\partial t} - (rx+c)\frac{\partial \phi}{\partial x} + \lambda\Big(\phi(x,t) - \int_0^x \phi(x-y,t)dF(y)\Big) = 0.$$

Noting that $\phi(x,t) = 0$ for $x < 0$, this can be rewritten as:

$$\frac{\partial \phi}{\partial t} - (rx+c)\frac{\partial \phi}{\partial x} - \lambda\Big(\int_{\mathbb{R}} \big(\phi\big(x+(-y),t\big) - \phi(x,t)\big)dF(y)\Big) = 0,$$

and therefore, from the definition of the Lévy measure $\nu(\cdot)$, we obtain:

$$\frac{\partial \phi}{\partial t} = (rx+c)\frac{\partial \phi}{\partial x} + \int_{\mathbb{R}} \Big(\phi\big(x+(-y),t\big) - \phi(x,t)\Big)\nu(dy) = 0,$$

We can see that this PIDE is equivalent to that in (3.63), given that the diffusion term is zero and therefore the second derivative to x is zero. This confirms that our results are consistent with existing definitions and models that have been studied in the literature, with the important difference that we take advantage of the integral equations and corresponding continuity results, which allow us to consider more complex models and to obtain appropriate solutions in all cases.

3.5.3 Regularity of the one-dimensional PD function

Finally, we study the case of the one dimensional model to show that the resulting survival and PD function enjoys the properties required to consider equation (3.63) in the strict sense. Specifically, this requires that $\Phi(x,u)$ is (at least) twice and once differentiable with respect to the spatial and temporal variables, respectively. These results are based on the corresponding calculations for the general family of second

order parabolic PIDEs, as studied in Garroni, Menaldi, et al., 1992. A reminder of the main results that we will use are given in Appendix B.4.

We first rewrite (3.63) in a concise form for the calculations that follow. Note that we adopt the notation of the appropriate spaces used in Garroni, Menaldi, et al., 1992, which is also used in Appendix B.4. It will be useful to set $\mathcal{D} := (0, \infty)$. This will allow us to consider a smooth initial condition, rather than the Heaviside function as in formulation of the PIDE in Corollary 3.5.7, since $\Phi(x, u) = 0$, for $x \leq 0$, by definition. Then, for a fixed time until maturity $T > 0$, we write:

$$\begin{cases} L\Phi(x,u) = I\Phi(x,u) & \text{for } (x,u) \in Q_T := \mathcal{D} \times [0,T] \\ \Phi(x,0) = 1 & \text{for } x \in \mathcal{D} \\ \Phi(x,t) = \mathbb{1}_{x>0} & \text{for } x \in \Sigma_T := \partial\mathcal{D} \times [0,T], \end{cases} \tag{3.65}$$

where we define the operator L by $L\Phi(x,u) := \frac{\partial \Phi}{\partial u} - \mathcal{L}\Phi(x,u)$ and the integral operator I by $I\Phi(x,u) := \int_{\mathbb{R}} \Big(\Phi(x+z,u) - \Phi(x,u)\Big)\nu(dz)$, and $\partial\mathcal{D}$ is the standard notation for the boundary of domain $\mathcal{D}$. Our aim is to show that the above PIDE has a solution satisfying appropriate regularity conditions. To this end, we first define some relevant function spaces that will be required for the subsequent regularity results.

Definition 3.5.9. Consider $\Omega \subset \mathbb{R}^n$ an open set, with closure $\bar{\Omega}$. Furthermore, consider a fixed time horizon $T > 0$ and define $Q_T = \Omega \times [0,T]$, with closure $\bar{Q}_T$. We then define the following spaces, for $0 < \alpha < 1$:

- $C^0(\bar{\Omega})$ is the Banach space of bounded continuous functions in $\bar{\Omega}$, with the natural supremum norm:

$$\|\cdot\|_{C^0(\bar{\Omega})} \equiv \|\cdot\|_{0,\bar{\Omega}} = \sup_{\Omega} |\cdot|$$

- $C^{2,1}(\bar{Q}_T)$ is the Banach space of functions $\varphi(x,t)$ belonging to $C^0(\bar{Q}_T)$ together their derivatives $\frac{\partial f}{\partial x}, \frac{\partial^2 f}{\partial x^2}, \frac{\partial f}{\partial t}$ in $\bar{Q}_T$ with natural norm.

- $C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)$ is the Banach space of function φ in $C^0(\bar{Q}_T)$ which are Hölder continuous in $\bar{Q}_T$ with exponent α in x and $\frac{\alpha}{2}$ in t i.e. having a finite value for the seminorm

$$\langle f \rangle^{(\alpha)}_{\bar{Q}_T} \equiv \langle f \rangle^{(\alpha)}_{x,\bar{Q}_T} + \langle f \rangle^{\left(\frac{\alpha}{2}\right)}_{t,\bar{Q}_T}$$

where

$$\langle f \rangle^{(\alpha)}_{x,\bar{Q}_T} = \inf\left\{C \geq 0 : \left|f(x,t) - f\left(x',t\right)\right| \leq C\left|x - x'\right|^{\alpha}, \forall x, x', t\right\}$$
$$\langle f \rangle^{\left(\frac{\alpha}{2}\right)}_{t,\bar{Q}_T} = \inf\left\{C \geq 0 : \left|f(x,t) - f\left(x,t'\right)\right| \leq C\left|t - t'\right|^{\frac{\alpha}{2}}, \forall x, t, t'\right\}$$

The quantity

$$\|f\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)} \equiv \|f\|_{\alpha,\bar{Q}_T} = \|f\|_{0,\bar{Q}_T} + \langle f \rangle^{(\alpha)}_{\bar{Q}_T}$$

defines a norm.

- $C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$ is the Banach space of functions $f(x,t)$ in $C^{2,1}(\bar{Q}_T)$ having a finite value for the seminorm:

$$\langle f \rangle_{\bar{Q}_T}^{(2+\alpha)} = \langle \partial_t f \rangle_{\bar{Q}_T}^{(\alpha)} + \sum_{i,j=1}^{d} \langle \partial_{ij} f \rangle_{\bar{Q}_T}^{(\alpha)} + \sum_{i=1}^{d} \langle \partial_i f \rangle_{t,\bar{Q}_T}^{\frac{1+\alpha}{2}}.$$

  Then, the quantity

$$\|f\|_{C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)} \equiv \|f\|_{2+\alpha,\bar{Q}_T} = \sum_{2r+s\leq 2} \|\partial_t^r \partial_x^s f\|_{0,\bar{Q}_T} + \langle f \rangle_{\bar{Q}_T}^{(2+\alpha)}$$

  defines a norm.

Proposition 3.5.10. *Consider a fixed time horizon $T > 0$ and the space $Q_T := \mathcal{D} \times [0,T]$, with closure $\bar{Q}_T$. Then, PDE (3.65) has a solution $\Phi(x,u)$, such that $\Phi(x,u) \in C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$.*

Proof. The proof relies on an appropriate fixed point argument. To this end, we first define the mapping $\mathcal{T}v = \Phi$, such that v is a solution of $L\Phi(x,u) = Iv$. Note that from Theorem B.9, there exists a unique $\Phi \in C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$ solving the local counterpart of (3.65), where the right hand side of the PDE is zero.

Consider now a function $v \in C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$. It follows that $Iv \in C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)$ and we also have that:

$$\|Iv\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)} \leq \varepsilon \|\nabla v\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)} + C(\varepsilon)\|v\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)},$$

from Theorem B.4.2. Furthermore, by definition of the mapping $\mathcal{T}$ we have that if $v \in C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$ then $\Phi = \mathcal{T}v \in C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$. Hence, $\mathcal{T}$ is a map from $C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$ onto itself and is also single-valued, by the uniqueness of the solution of the PDE.

We will now show that $\mathcal{T}$ is also a contraction in order to then apply Banach's fixed point argument. To this end, consider $v, v' \in C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$, with the corresponding mappings $Tv, Tv' \in C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$. By the definition of the mapping T have that:

$$\begin{cases} L\Phi(x,u) = Iv & \text{for } (x,u) \in Q_T := \mathcal{D} \times [0,T] \\ L\Phi'(x,u) = Iv' & \text{for } (x,u) \in Q_T := \mathcal{D} \times [0,T], \end{cases} \tag{3.66}$$

and therefore $L\hat{\Phi}(x,u) = I\hat{v}$, with $\hat{\Phi} := \Phi - \Phi'$ and $\hat{v}$ is defined analogously. Hence:

$$\begin{aligned} \|\hat{\Phi}\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)} =& \|\mathcal{T}v - \mathcal{T}v'\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)} \leq C\|I\hat{v}\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)} \\ \leq& \varepsilon \|\nabla \hat{v}\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)} + C(\varepsilon)\|\hat{v}\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)}, \end{aligned} \tag{3.67}$$

where the last inequality follows from Theorem B.4.2. To show that $\mathcal{T}$ is indeed a contraction, we first note that all terms in the final expression above are bounded by $\|\hat{v}\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)}$. We need $\|\hat{\Phi}\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)} \leq k\|\hat{v}\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)}$, with $k < 1$. For this, notice that the first term in the final expression above can be made arbitrarily small, however the second depends on the C value, which in turn depends on the time horizon T. We can therefore make $C(\epsilon) < 1$ if we consider a small enough horizon, i.e., $T = \delta$, creating a solution $\Phi(x,u) \in C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_\delta)$. We can apply this approach to find

an appropriate solution in $\Omega \times [\delta, 2\delta]$, continuing until the entire interval $[0, T]$ is covered. □

Remark 3.5.11. Based on the results given by Garroni, Menaldi, et al., 1992, the result above can be extended for higher dimensions with $x \in \mathbb{R}^d$ and an analogous parabolic operator L. Therefore, Proposition 3.5.10 holds for the stochastic volatility model, as well. Furthermore, in the case of the regime switching model we obtain a simple coupling of parabolic PIDEs (identical to that for the one dimensional model), and it is therefore expected that the regularity result holds for the PD function $\Phi(x, \cdot, u)$.

With this result we have shown that we can go beyond the notion of viscosity solutions in the case of the PD functions and obtain solutions that satisfy all required regularity properties. In the next Chapter we will consider numerical solutions to the PIDEs, some of which we can now interpret as strong solutions, under the conditions mentioned above.

3.6 Conclusion

In this Chapter we have focused on a generalized approach of estimating PD values, considering both the cases of variable starting times and maturities. We show that under certain conditions imposed on the models representing the asset processes, these two cases can be dealt with equivalently and lead to important novel representations of the PD function. Specifically, with the integral equation approach, we can construct a robust mathematical framework that allows us to develop both theoretical and numerical tools for the calculation of the PD values. This methodology has important advantages over standard Monte Carlo methods, as well as over existing approaches using PIDEs, as we are able to consider sophisticated models that incorporate multiple latent variables, without sacrificing mathematical rigor for required regularity assumptions.

In terms of practicality and applications, the proposed framework covers many of the difficulties financial institutions face due to the new regulatory requirements for provision calculations, as well as continuous credit risk monitoring for SICR events. We hypothesize that such this approach could be useful for practitioners, given that it constitutes a complete and efficient modelling framework, with which one can calculate Point-in-Time and Lifetime PD values, each of which are used extensively in credit risk management. Specifically, this framework is motivated by the needs created by the IFRS 9 regulations, under which forecasting credit losses accurately and efficiently is of paramount importance. We show how the PD estimations can be used to calculate Stage 2 provisions, as well as more advanced, scenario-based provisions, and of extensive further applications in credit risk modelling.

Finally, we note that this approach most likely is best fit for corporate and small business loans, where the estimation of asset processes has been documented in well-established work. Of course, it is possible that with new developments in payment services and Open-Banking solutions (in accordance to the Payment Services Directive 2), such methods could be applied to individual consumers, given sufficient historical data. An example of recent work done in this direction is Tobback and Martens, 2019. To conclude, it is important to mention that the LGD parameter is also of great importance for provision calculations; in this work we considered a constant LGD, however, in practice, LGD values require separate model development, often related to current macroeconomic variables, as shown in Bellotti and

Crook, 2012. Future research could focus on considering appropriate models for the evolution of the LGD, in combination with the PD function.

Chapter 4

Numerical Schemes for the PD functions and credit risk modelling

4.1 Background

In this Chapter, we will develop numerical schemes to solve the PIDEs obtained in Chapter 3 and use the resulting PD values in specific examples of the aforementioned IFRS 9 modelling tasks. We choose to focus on the numerical solutions of the PIDEs, rather than the corresponding integral equations, as we will be able to employ standard finite difference schemes to estimate the solutions, as detailed below. For clarity and illustrative purposes, we will first consider the one dimensional OU model given by (1.14) (recall that we have shown that the PD function $\Phi(x,u) \in C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$ is a strong solution to the corresponding PIDE). From the resulting finite difference scheme we will then be able to build the solutions for the regime switching and stochastic volatility models. We will focus on these models for our numerical solutions, as they cover the applications in credit risk that we consider in this Chapter, noting the case of the generalized model (3.7) can be developed by combining the methods that follow. However, due to the additional terms, the corresponding numerical scheme suffers from the well-known "curse of dimensionality" problem. We further discuss this issue and how we can employ modern numerical techniques to overcome it in Chapter 5.

Our approach in this section follows the methodology developed in Cont and Voltchkova, 2005a and d'Halluin, Forsyth, and Vetzal, 2005. We extend the numerical schemes by considering variable coefficients and further develop the corresponding methods for the regime switching and stochastic volatility models. Due to the additional variables, we will see that these require careful handling of the derivative discretizations to ensure the required stability and monotonicity properties hold. As mentioned, we will start with the one dimensional model, which produces the PIDE given in (3.63), the finite difference scheme for which is similar to that developed in Cont and Voltchkova, 2005a. However, this first step will allow us to explicitly account for the variable drift term and detail its effect on the finite difference scheme and is therefore worth presenting the analytical calculations.

4.2 One dimensional model

Before implementing the numerical methods, it is important to discuss the spatial and temporal domains over which the schemes will be solved. We consider a spatial domain $x \in \mathcal{D} \subset \mathbb{R}$. Therefore, for the construction of the numerical scheme one can consider the interval $x \in [0, S]$ with non-trivial solutions $\Phi(x,u) \in (0,1)$ (in practice, the value of S depends on the parameters of the underlying processes and

its approximation may require Monte Carlo simulations). However, given that the PIDEs contain the non-local integral terms, and the OU process is defined on $\mathbb{R}$, we will define $\mathcal{D} = [-B, B]$, for some constant $B > S$ and extend the boundary conditions $\Phi(x, \cdot) = 0$ for $x \in [-B, 0)$ and $\Phi(x, \cdot) = 1$ for $x \in (S, B]$. This way, we will be able to calculate the integral term, as detailed below. For the temporal domain, we simply consider $t \in [0, 1]$ (we rewrite u as t as there is not risk of confusion in what follows). Given the added complexity from the non-local term, we give a detailed explanations of each of the three aforementioned schemes in this section, along with examples of specific asset value processes and, subsequently, examples of the modelling tasks pertaining to credit risk under the IFRS 9 framework we previously discussed.

We write (3.63) as follows:

$$\frac{\partial \Phi}{\partial t} = k(\theta - x)\frac{\partial \Phi}{\partial x} + \frac{1}{2}\sigma^2 \frac{\partial^2 \Phi}{\partial x^2} + \int_{\mathbb{R}} \Phi(x + z, t)\nu(dz) - \Phi(x, t)\int_{\mathbb{R}} \nu(dz). \tag{4.1}$$

Note that in the discretized version of this PIDE we will also have to approximate the integral with respect to the Lévy measure. We employ an implicit scheme leading to a backward time centered space (BTCS) method, and handle the non-local term explicitly, as in Cont and Voltchkova, 2005a. Consider space and time grids, with step sizes Δx and Δt, and with N and T total points, respectively. Therefore, we have that Φ_p^q represents the survival probability at the grid point $t = t_0 + q\Delta t$, $x = x_0 + p\Delta x$, i.e. $\Phi_p^q = \Phi(x_0 + p\Delta x, t_0 + q\Delta t)$. Furthermore, let L, D and U be number of grid points in the intervals $[-B, 0)$, $[0, S]$ and $(S, B]$, respectively, so that $N = L + D + U$.

For the integral terms, we first must approximate the jump density by considering a ball around the x-value of the grid:

$$\bar{f}_i = \frac{1}{\Delta x}\int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} f(x)dx. \tag{4.2}$$

Then, noting that $\nu(dz) = \lambda F(dz)$, we can approximate the first and second integral terms in (4.1) using:

$$\mathcal{I}\Phi_p^q := \sum_{i=-J/2}^{J/2} \Phi_{p+i}^q \bar{f}_i \Delta z, \tag{4.3}$$

$$\hat{I} = \sum_{i=-J/2}^{J/2} \bar{f}_i \Delta z, \tag{4.4}$$

for some $J \in \mathbb{Z}_+$ large enough to ensure that $\hat{I}$ is sufficiently close to 1. In the above, we have defined the operator $\mathcal{I} : \mathcal{C} \to \mathcal{C}$, where $\mathcal{C}$ is the Banach space as defined in Proposition 3.4.9. We will refer to this as the integral operator. For simplicity, we will be taking $\Delta x = \Delta z$ in the calculations and numerical results below.

The resulting implicit scheme for PIDE (4.1) is given by:

$$\frac{\Phi_p^{q+1} - \Phi_p^q}{\Delta t} = k(\theta - x_p)\frac{\Phi_{p+1}^{q+1} - \Phi_{p-1}^{q+1}}{2\Delta x} + \frac{1}{2}\sigma^2 \frac{\Phi_{p+1}^{q+1} - 2\Phi_p^{q+1} + \Phi_{p-1}^{q+1}}{\Delta x^2} + \lambda \mathcal{I}\Phi_p^q - \lambda \hat{I}\Phi_p^q, \tag{4.5}$$

which, upon rearranging, can be written as:

$$-\Phi_{p-1}^{q+1} c_p \Delta t + \Phi_p^{q+1}(1 + a_p \Delta t) - \Phi_{p-1}^{q+1} b_p \Delta t = (1 - \lambda \Delta t \hat{I})\Phi_p^q + \lambda \Delta t \mathcal{I}\Phi_p^q, \tag{4.6}$$

for $q = 1, 2, \dots T-1$ and with coefficients a_p, b_p and c_p, for $p = 0, 1, \dots, N-1$, given by:

$$c_p = \frac{\sigma^2}{2\Delta x^2} - \frac{k(\theta - x_p)}{2\Delta x}$$
$$a_p = \frac{\sigma^2}{\Delta x^2}$$
$$b_p = \frac{\sigma^2}{2\Delta x^2} + \frac{k(\theta - x_p)}{2\Delta x} \tag{4.7}$$

Hence, system (4.5) can be written in the matrix form below:

$$M\Phi^{q+1} = \Lambda\Phi^q + b, \text{ for } q = 0, 1, \dots, T-1,$$

where $\Phi^q, b \in \mathbb{R}^N$ and $M \in \mathbb{R}^{N\times N}$ are given by:

$$\Phi^q = \begin{pmatrix} \Phi_0^q \\ \Phi_1^q \\ \vdots \\ \Phi_{N-1}^q \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ b_U \end{pmatrix}, \quad M = \begin{bmatrix} I_L & 0_D & 0_U \\ 0_L & \mathcal{M} & 0_U \\ 0_L & 0_D & I_U \end{bmatrix}, \tag{4.8}$$

with $b_U = (1, \cdots, 1)^T \in \mathbb{R}^U$, $I_n, 0_n$ being the $n \times n$-dimensional identity and zero matrices, respectively, $\mathcal{M} \in \mathbb{R}^{D\times D}$ given by:

$$\mathcal{M} = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -c_1\Delta t & 1 + a_1\Delta t & -b_1\Delta t & 0 & \cdots & 0 & 0 & \\ 0 & -c_2\Delta t & 1 + a_2\Delta t & -b_2\Delta t & \cdots & 0 & 0 & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & 0 & \cdots & -c_{N-1}\Delta t & 1 + a_{N-1}\Delta t & -b_{N-1}\Delta t \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 \end{pmatrix}, \tag{4.9}$$

and $\Lambda \in \mathbb{R}^{N\times N}$:

$$\Lambda = \begin{pmatrix} \lambda\Delta t\bar{f}_{-1} & \lambda\Delta t\bar{f}_0 + \hat{F} & \lambda\Delta t\bar{f}_1 & \cdots & \lambda\Delta t\bar{f}_{J/2} & 0 & \cdots & 0 \\ \lambda\Delta t\bar{f}_{-2} & \lambda\Delta t\bar{f}_{-1} & \lambda\Delta t\bar{f}_0 + \hat{F} & \cdots & \lambda\Delta t\bar{f}_{J/2-1} & \lambda\Delta t\bar{f}_{J/2} & \cdots & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & \cdots & \lambda\Delta t\bar{f}_{-J/2+1} & \lambda\Delta t\bar{f}_{-J/2} & \cdots & \lambda\Delta t\bar{f}_{J/2} \end{pmatrix}, \tag{4.10}$$

where $\hat{F} := 1 - \lambda\Delta t\hat{I}$. At each time step we can then calculate $\Phi^{q+1} = M^{-1}(\Lambda\Phi^q + b)$, to obtain the solution at time $t = q + 1$.

For the implementation of the numerical scheme we must analyze the necessary properties pertaining to its stability and monotonicity, for which we use the same definitions and approach as in Cont and Voltchkova, 2005a. Specifically, we define these conditions as follows.

Definition 4.2.1.

(*i*) Scheme (4.5) is stable if and only if, for a bounded initial condition, there exist Δx and Δt such that the solution exists and is bounded, i.e. $|\Phi_p^q| \leq C$, for all p, q and some $C > 0$.

(*ii*) Scheme (4.5) is monotone, i.e. for two initial conditions Φ^0 and $\tilde{\Phi}^0$:

$$\Phi^0 > \tilde{\Phi}^0 \Rightarrow \Phi^q > \tilde{\Phi}^q,$$

for all q. Note that the comparison of the two vectors is to be understood in the element-by-element sense. This condition is often referred to as the discrete comparison principle.

These conditions must hold in order to avoid spurious oscillations in the numerical solutions and nonsensical values. We will show that the numerical scheme for the PIDE we have obtained is conditionally stable and monotone. As we will see from the result below, the condition is not restrictive and can easily be satisfied when selecting the parameters of the scheme, without significant computational cost.

Proposition 4.2.2. *Scheme (4.5) is stable and monotone if*

$$\Delta x \leq \frac{\sigma^2}{k\theta} \text{ and } \Delta t \leq \frac{1}{\lambda \hat{I}}. \tag{4.11}$$

Proof. We prove the two results separately, starting with stability. Let Φ^0 be a bounded initial condition, i.e. $||\Phi^0||_\infty < \infty$. Following Cont and Voltchkova, 2005a, we will proceed by induction and contradiction. Let $||\Phi^q||_\infty \leq ||\Phi^0||_\infty$ and suppose $||\Phi^{q+1}||_\infty > ||\Phi^0||_\infty$. Therefore, there exists $p_0 \in \{1, \dots, N-1\}$ such that $|\Phi^{q+1}_{p_0}| = ||\Phi^{q+1}||_\infty$ and $|\Phi^{q+1}_{p_0}| \geq |\Phi^{q+1}_p|$, for all $p \in \{1, \dots, N-1\}$. We will prove that this leads to a contradiction. Observing that $a_p = b_p + c_p$, we can write:

$$||\Phi^{q+1}||_\infty = |\Phi^{q+1}_{p_0}| = \big[-c_{p_0}\Delta t + (1 + a_{p_0}\Delta t) - b_{p_0}\Delta t \big] |\Phi^{q+1}_{p_0}|. \tag{4.12}$$

To proceed, we will need coefficients a_p, b_p, c_p be non-negative. This is true for a_p, for all p. From the remaining coefficients, we obtain the condition:

$$\frac{\sigma^2}{2\Delta x^2} \geq \frac{|k(\theta - x_p)|}{2\Delta x},$$

and hence by requiring: $\sigma^2 \geq k||\theta - x||_\infty \Delta x$, we can ensure that the condition is satisfied at all points on the x-grid. Given that Φ is identically zero for $x < 0$, this can be succinctly written as:

$$\Delta x \leq \frac{\sigma^2}{k\theta}. \tag{4.13}$$

Continuing from (4.12), and noting that all coefficients are non-negative, we now have:

$$\begin{aligned} ||\Phi^{q+1}||_\infty &\leq -c_{p_0}\Delta t |\Phi^{q+1}_{p_0-1}| + (1 + a_{p_0}\Delta t)|\Phi^{q+1}_{p_0}| - b_{p_0}\Delta t |\Phi^{q+1}_{p_0+1}| \\ &\leq | -c_{p_0}\Delta t \Phi^{q+1}_{p_0-1} + (1 + a_{p_0}\Delta t)\Phi^{q+1}_{p_0} - b_{p_0}\Delta t \Phi^{q+1}_{p_0+1}|, \end{aligned} \tag{4.14}$$

and from (4.6), in combination with the induction hypothesis, it follows that:

$$||\Phi^{q+1}||_\infty \leq |(1 - \lambda\Delta t\hat{I})\Phi^q_{p_0} + \lambda\Delta t \mathcal{I}\Phi^q_{p_0}| \leq ||\Phi^0||_\infty, \tag{4.15}$$

where the last steps hold if $1 - \lambda\Delta t\hat{I} \geq 0$, leading to the second condition. Then, since $\Phi^q_{p_0} \leq ||\Phi^q||_\infty$, and $\mathcal{I}\Phi^q_{p_0} \leq \hat{I}||\Phi^q||_\infty$, the above contradicts the assumption that $||\Phi^{q+1}||_\infty > ||\Phi^0||_\infty$. Hence, the scheme is stable, provided (4.13) is satisfied.

Monotonicity is proven in a similar way. Specifically, let $\Phi^q, \tilde{\Phi}^q$ be two solutions corresponding to initial conditions $\Phi^0, \tilde{\Phi}^0$, respectively, with $\Phi^0 \geq \tilde{\Phi}^0$ and define $d^q := \Phi^q - \tilde{\Phi}^q$. We will proceed by induction and contradiction, as above. We have that $d^0 \geq 0$ and also assume $d^q \geq 0$. We suppose that $d^{p+1} < 0$, i.e., there exists p_0 such that $\inf_p d_p^{q+1} = d_{p_0}^{q+1} < 0$. Then:

$$\inf_p d_p^{q+1} = d_{p_0}^{q+1} = -c_{p_0}\Delta t d_{p_0}^{q+1} + (1 + a_{p_0}\Delta t) d_{p_0}^{q+1} - b_{p_0}\Delta t d_{p_0}^{q+1}$$
$$\geq -c_{p_0}\Delta t d_{p_0-1}^{q+1} + (1 + a_{p_0}\Delta t) d_{p_0}^{q+1} - b_{p_0}\Delta t d_{p_0+1}^{q+1} = (1 - \lambda\Delta t\hat{I}) d_{p_0}^q + \lambda\Delta t\mathcal{I} d_{p_0}^q \geq 0, \tag{4.16}$$

where the last step follows from the induction hypothesis and we have supposed that condition (4.13) is satisfied. By contradiction, we therefore conclude that $d^{p+1} > 0$, as required. □

4.3 Regime switching

We now turn to the regime switching model, with regimes $r \in \mathcal{R}$ and a total of R regimes. In the BTCS discretized version of (3.40) we let $\Phi_{p,r}^q$ represent the survival probability at the grid point $t_q = t_0 + q\Delta t$, $x_p = x_0 + p\Delta x$, when the underlying Markov process is originally in state $r \in \mathcal{R}$, i.e., $\Phi_{p,r}^q = \Phi(x_p, r, t_q)$, with $p = 0, 1, \ldots N-1, q = 0, 1, \ldots T, r \in \mathcal{R}$. The discretized PIDE can be written as:

$$-\Phi_{p-1,r}^{q+1} c_{p,r}\Delta t + \Phi_{p,r}^{q+1}\left(1 + a_{p,r}\Delta t\right) - \Phi_{p-1,r}^{q+1} b_{p,r}\Delta t - \sum_{j\neq r} q_{rj}\Phi_{p,j}^{q+1}\Delta t$$
$$= (1 - \lambda\Delta t\hat{I})\Phi_{p,r}^q + \lambda\Delta t\mathcal{I}\Phi_{p,r}^q \tag{4.17}$$

where $\mathcal{I}\Phi_{p,r}^q$ and $\hat{I}$ are as in (4.3) and (4.4), respectively. The coefficients of this scheme are then given by:

$$c_{p,r} = \frac{\sigma_r^2}{2\Delta x^2} - \frac{k_r(\theta_r - x_p)}{2\Delta x}$$
$$a_{p,r} = \frac{\sigma_r^2}{\Delta x^2} + \sum_{j\neq r} q_{\rho j}$$
$$b_{p,r} = \frac{\sigma_r^2}{2\Delta x^2} + \frac{k_r(\theta_r - x_p)}{2\Delta x}. \tag{4.18}$$

In matrix notation, the regime switching PIDE can be written as:

$$M^{RS}\Phi^{q+1} = \Lambda^{RS}\Phi^q + b^{RS}, \tag{4.19}$$

where the block-form matrices $\Phi, M^{RS}, \Lambda^{RS} \in \mathbb{R}^{NR\times NR}$ and $b^{RS} \in \mathbb{R}^{NR}$ are given by:

$$\Phi^q = \begin{bmatrix} \Phi_{\cdot,1}^q \\ \Phi_{\cdot,2}^q \\ \vdots \\ \Phi_{\cdot,R}^q \end{bmatrix}, \quad b^{RS} = \begin{bmatrix} b \\ b \\ \vdots \\ b \end{bmatrix}, \quad \Lambda^{RS} = \begin{bmatrix} \Lambda & 0_N & \cdots & 0_N \\ 0_N & \Lambda & \cdots & 0_N \\ \vdots & \vdots & \vdots & \vdots \\ 0_N & 0_N & \cdots & \Lambda \end{bmatrix},$$

$$M^{RS} = \begin{bmatrix} M_{r_1} & -\Delta t q_{r_1 r_2} I_N & \cdots & -\Delta t q_{r_1 r_R} I_N \\ -\Delta t q_{r_2 2 r_1} I_N & M_{r_2} & \cdots & -\Delta t q_{r_2 r_R} I_N \\ \vdots & \vdots & \vdots & \vdots \\ -\Delta t q_{r_R r_1} I_N & -\Delta t q_{r_R r_2} I_N & \cdots & M_{r_R} \end{bmatrix}, \tag{4.20}$$

with $\Phi^q_{\cdot,r} = (\Phi^q_{0,r}, \dots \Phi^q_{N-1,r})^T$, b, Λ, as in (4.8) and (4.10) and the regime-specific matrices $M_{r_i} \in \mathbb{R}^{N\times N}$ for $r_i \in \mathcal{R}, i = 1, \dots, R$, are as in (4.8) by replacing a_p, b_p, c_p with $a_{p,r}, b_{p,r}, c_{p,r}$. As in the discretization of the PIDE for the one dimensional OU model, we will have to prove the appropriate stability and monotonicity results for the regime switching model.

Lemma 4.3.1. *Scheme (4.17) is stable and monotone if*

$$\Delta x \leq \frac{\sigma_r^2}{k_r \theta_r} \text{ and } \Delta t \leq \frac{1}{\lambda \hat{I}} \tag{4.21}$$

for all $r \in \mathcal{R}$.

Proof. Let Φ^0 be a bounded initial condition for the survival probability. Note that this initial condition accounts for all regimes. As above, we will proceed by induction and contradiction. Let $||\Phi^q||_\infty \leq ||\Phi^0||_\infty$ and suppose $||\Phi^{q+1}||_\infty > ||\Phi^0||_\infty$. In this case, this means that there exists $(p_0, r_0) \in \{0, 1, \dots, N-1\} \times \mathcal{R}$ such that $|\Phi^{q+1}_{p_0,r_0}| = ||\Phi^{q+1}||_\infty$, with $|\Phi^{q+1}_{p_0,r_0}| \geq |\Phi^{q+1}_{p,r}|$, for all $(p, r) \in \{0, 1, \dots, N-1\} \times \mathcal{R}$. Hence:

$$\begin{aligned}
||\Phi^{q+1}||_\infty = |\Phi^{q+1}_{p_0,r_0}| &= \Big[- c_{p_0,r_0}\Delta t + (1 + a_{p_0,r_0}\Delta t) - b_{p_0,r_0}\Delta t - \sum_{j\neq r_0} q_{r_0 j}\Delta t \Big] |\Phi^{q+1}_{p_0,r_0}| \\
&\leq -c_{p_0,r_0}|\Phi^{q+1}_{p_0-1,r_0}|\Delta t + (1 + a_{p_0,r_0}\Delta t)|\Phi^{q+1}_{p_0,r_0}| - b_{p_0,r_0}|\Phi^{q+1}_{p_0+1,r_0}|\Delta t - \sum_{j\neq r_0} q_{r_0 j}|\Phi^{q+1}_{p_0,j}|\Delta t \\
&\leq | - c_{p_0,r_0}\Phi^{q+1}_{p_0-1,r_0}\Delta t + (1 + a_{p_0,r_0}\Delta t)\Phi^{q+1}_{p_0,r_0} - b_{p_0,r_0}\Phi^{q+1}_{p_0+1,r_0}\Delta t - \sum_{j\neq r} q_{r_0 j}\Phi^{q+1}_{p_0,j}\Delta t| \\
&\leq |(1 - \lambda\Delta t\hat{I})\Phi^q_{p_0,r_0} + \lambda\Delta t\mathcal{I}\Phi^q_{p_0,r_0}| \leq ||\Phi^0||_\infty,
\end{aligned}$$

where the last inequality follows from the same calculations as in Proposition 4.2.2. In the above, we must have $a_{p,r}, b_{p,r}, c_{p,r} > 0$ for each regime $r \in \mathcal{R}$, leading to the first condition in (4.21).

For monotonicity, again let $\Phi^q, \tilde{\Phi}^q$ be two solutions corresponding to $\Phi^0, \tilde{\Phi}^0$, respectively, with $\Phi^0 \geq \tilde{\Phi}^0$. Assume $d^q := \Phi^q - \tilde{\Phi}^q > 0$ and suppose $d^{q+1} \leq 0$. Hence, there exists p_0, r_0 such that $\inf_{p,r} d^{q+1}_{p,r} = d^{q+1}_{p_0,r_0} < 0$. Proceeding as in Proposition 4.2.2:

$$\begin{aligned}
\inf_{p,r} d^{q+1}_{p,r} = d^{q+1}_{p_0,r_0} &= \Big[- c_{p_0,r_0}\Delta t + (1 + a_{p_0,r_0}\Delta t) - \sum_{j\neq r_0} q_{r_0 j}\Delta t - b_{p_0,r_0}\Delta t \Big] d^{q+1}_{p_0,r_0} \\
&\geq -c_{p_0,r_0} d^{q+1}_{p_0-1,r_0}\Delta t + (1 + a_{p_0,r_0}) d^{q+1}_{p_0,r_0}\Delta t - \sum_{j\neq r_0} q_{r_0 j} d^{q+1}_{p_0,j}\Delta t - b_{p_0,r_0} d^{q+1}_{p_0+1,r_0}\Delta t \\
&= (1 - \lambda\Delta t\hat{I}) d^q_{p_0,r_0} + \lambda\Delta t\mathcal{I} d^q_{p_0,r_0} \geq 0.
\end{aligned}$$

Stability and monotonicity for scheme (4.17) thus follow by contradiction. □

4.4 Stochastic volatility model

Finally, we present the numerical scheme for model (3.41). For this case, we must consider a discretization of the volatility process $y \in \mathcal{V}$ of size V. For the numerical implementation we use $y \in [0, Y_{\max}]$ for some appropriate value $Y_{\max}$. As above, we adopt the notation $\Phi^q_{p,j}$ for the survival probability at the grid point $t_q = t_0 + q\Delta t$, $x_p = x_0 + p\Delta x$, and $y_j = y_0 + j\Delta y$, i.e. $\Phi^q_{p,j} = \Phi(x_p, y_j, t_q)$, with $p = 0, 1, \dots N - 1, q = 0, 1, \dots T$ and $j = 0, 1, \dots, V - 1$.

In this case the discretization scheme requires an alternative approach. Specifically, when the coefficient of the diffusion term becomes 0 or $y \to 0$, the analogous to the previous stability and monotonicity conditions fails. The solution to this is to consider an Alternating Direction Implicit (ADI) approximation to the first derivative terms corresponding to both the asset and CIR volatility processes, as shown below:

$$\frac{\partial \Phi}{\partial x} \approx \begin{cases} \frac{\Phi^{q+1}_{p+1,j} - \Phi^{q+1}_{p,j}}{\Delta x}, & \text{if } k(\theta - x_p) \geq 0 \\ \frac{\Phi^{q+1}_{p,j} - \Phi^{q+1}_{p-1,j}}{\Delta x}, & \text{if } k(\theta - x_p) < 0 \end{cases} \tag{4.22}$$

$$\frac{\partial \Phi}{\partial y} \approx \begin{cases} \frac{\Phi^{q+1}_{p,j+1} - \Phi^{q+1}_{p,j}}{\Delta y}, & \text{if } \kappa(\mu - y_j) \geq 0 \\ \frac{\Phi^{q+1}_{p,j} - \Phi^{q+1}_{p,j-1}}{\Delta y}, & \text{if } \kappa(\mu - y_j) < 0 \end{cases} \tag{4.23}$$

Furthermore recall that at the boundary $y = Y_{\max}$ we have:

$$\frac{\Phi^{q+1}_{p,V} - \Phi^{q+1}_{p,V-2}}{2\Delta y} \approx \frac{\partial \Phi}{\partial y}(x, Y_{\max}, u) = 0, \tag{4.24}$$

and therefore $\Phi_{p,V} \approx \Phi_{p,V-2}$. This allows us to approximate the second derivative at the boundary by:

$$\frac{\Phi^{q+1}_{p,V} - 2\Phi^{q+1}_{p,V-1} + \Phi^{q+1}_{p,V-1}}{\Delta y^2} = \frac{2\left(\Phi^{q+1}_{p,V-2} - 2\Phi^{q+1}_{p,V-1}\right)}{\Delta y^2} \tag{4.25}$$

We can now write the implicit scheme for the PIDE corresponding to the stochastic volatility model:

$$\begin{aligned} -\Phi^{q+1}_{p-1,j} c_{p,j}\Delta t + \Phi^{q+1}_{p,j}\left(1 + a_{p,j}\Delta t\right) - \Phi^{q+1}_{p-1,j} b_{p,j}\Delta t - \Phi^{q+1}_{p,j-1} e_{p,j}\Delta t - \Phi^{q+1}_{p,j+1} f_{p,j}\Delta t \\ = (1 - \lambda\Delta t\hat{I})\Phi^q_{p,j} + \lambda\Delta t\mathcal{I}\Phi^q_{p,j}, \end{aligned} \tag{4.26}$$

where the coefficients are given by:

$$
\begin{aligned}
c_{p,j} &= \frac{y_j}{2\Delta x^2} - \frac{k(\theta - x_p)}{\Delta x}\mathbb{1}_{\{k(\theta-x_p)<0\}},\\
a_{p,j} &= \frac{y_j}{\Delta x^2} + \frac{\xi^2 y_j}{\Delta y^2} + \left|\frac{k(\theta - x_p)}{\Delta x}\right| + \left|\frac{\kappa(\mu - x_p)}{\Delta y}\right|,\\
b_{p,j} &= \frac{y_j}{2\Delta x^2} + \frac{k(\theta - x_p)}{\Delta x}\mathbb{1}_{\{k(\theta-x_p)>0\}},\\
e_{p,j} &= \frac{\xi^2 y_j}{2\Delta y^2}\mathbb{1}_{\{y\neq Y_{\max}\}} + \frac{\xi^2 y_j}{\Delta y^2}\mathbb{1}_{\{y=Y_{\max}\}} - \frac{\kappa(\mu - x_p)}{\Delta y}\mathbb{1}_{\{\kappa(\mu-y_j)<0 \,\cap\, y\neq 0 \,\cap\, y\neq Y_{\max}\}},\\
f_{p,j} &= \frac{\xi^2 y_j}{2\Delta y^2}\mathbb{1}_{\{y\neq Y_{\max}\}} + \frac{\kappa(\mu - x_p)}{\Delta y}\mathbb{1}_{\{\kappa(\mu-y_j)>0 \,\cap\, y\neq Y_{\max}\}}.
\end{aligned}
\tag{4.27}
$$

The solution of the resulting scheme:

$$M^{SV}\Phi^{q+1} = \Lambda^{SV}\Phi^q + b^{SV}$$

will result in estimations of the survival probability at each state of the underlying stochastic volatility process. Therefore, as in the regime switching model, we obtain the vectors $\Phi^q = [\Phi^q_{\cdot,0}\ \Phi^q_{\cdot,1}\cdots\Phi^q_{\cdot,V-1}]^T \in \mathbb{R}^{NV\times NV}$, $b^{SV} = [b\ b\cdots b]^T \in \mathbb{R}^{NV}$ and matrix $M^{SV} \in \mathbb{R}^{NV\times NV}$ as given below, in block form:

$$
M^{SV} = \begin{bmatrix}
M_0 & -\Delta t f_{p,0} I_N & 0_N & \cdots & 0_N\\
-\Delta t e_{p,1} I_N & M_1 & -\Delta t f_{p,1} I_N & \cdots & 0_N\\
\vdots & \vdots & \vdots & \vdots & \\
0_N & \cdots & -\Delta t e_{p,V-2} I_N & M_{V-2} & -\Delta t f_{p,V-2} I_N\\
0_N & \cdots & 0_N & -\Delta t e_{p,V-1} I_N & M_{V-1}
\end{bmatrix},
$$

where $M_j \in \mathbb{R}^{N\times N}$ for $j = 0, 1, \ldots, V-1$ is given by (4.8) by replacing a_p, b_p, c_p with $a_{p,j}, b_{p,j}, c_{p,j}$ and $\Lambda^{SV} \in \mathbb{R}^{NV\times NV}$ is in the same form as Λ^{RS} in (4.20). We now prove the required stability and monotonicity results for the stochastic volatility case.

Lemma 4.4.1. *Scheme (4.26) is unconditionally stable and monotone.*

Proof. The proof follows almost identically to the regime switching case. Again, let Φ^0 be a bounded initial condition for the survival probability, $||\Phi^q||_\infty \leq ||\Phi^0||_\infty$ and suppose $||\Phi^{q+1}||_\infty > ||\Phi^0||_\infty$. Hence, there exists $(p_0, j_0) \in \{0, 1, \ldots, N-1\} \times \{0, 1, \ldots, V-1\}$ such that $|\Phi^{q+1}_{p_0,j_0}| = ||\Phi^{q+1}||_\infty$, with $|\Phi^{q+1}_{p_0,j_0}| \geq |\Phi^{q+1}_{p,j}|$, for all $p \in \{0, 1, \ldots, N-1\} \times \{0, 1, \cdots, V-1\}$. Hence, noting that no conditions on $\Delta x, \Delta t$ need be imposed since all the coefficients are positive by construction, we have:

$$
\begin{aligned}
&||\Phi^{q+1}||_\infty = |\Phi^{q+1}_{p_0,j_0}| = \left[-c_{p_0,j_0}\Delta t + (1 + a_{p_0,j_0}\Delta t) - b_{p_0,j_0}\Delta t - e_{p_0,j_0}\Delta t - f_{p_0,j_0}\Delta t\right]|\Phi^{q+1}_{p_0,j_0}|\\
&\leq -c_{p_0,j_0}|\Phi^{q+1}_{p_0-1,j_0}|\Delta t + \left(1 + a_{p_0,j_0}\Delta t\right)|\Phi^{q+1}_{p_0,j_0}| - b_{p_0,j_0}|\Phi^{q+1}_{p_0+1,j_0}|\Delta t\\
&- e_{p_0,j_0}|\Phi^{q+1}_{p_0,j_0-1}|\Delta t - f_{p_0,j_0}|\Phi^{q+1}_{p_0,j_0+1}|\Delta t \leq |(1 - \lambda\Delta t\hat{I})\Phi^q_{p_0,j_0} + \lambda\Delta t\mathcal{I}\Phi^q_{p_0,j_0}| \leq ||\Phi^0||_\infty,
\end{aligned}
$$

To conclude, we prove the scheme is monotone. Consider initial conditions $\Phi^0, \tilde{\Phi}^0$, respectively, with $\Phi^0 \geq \tilde{\Phi}^0$. With $d^q := \Phi^q - \tilde{\Phi}^q > 0$, we also suppose

$d^{q+1} < 0$, i.e., there exists a pair (p_0, j_0) such that $\inf_{p,j} d^{q+1}_{p,j} = d^{q+1}_{p_0,j_0} < 0$, and calculate:

$$\begin{aligned}\inf_{p,j} d^{q+1}_{p,j} = d^{q+1}_{p_0,j_0} &= \big[- c_{p_0,j_0}\Delta t + (1 + a_{p_0,j_0}\Delta t) - b_{p_0,j_0}\Delta t - e_{p_0,j_0}\Delta t - f_{p_0,j_0}\Delta t\big] d^{q+1}_{p_0,j_0} \\ &\geq -c_{p_0,j_0} d^{q+1}_{p_0-1,r_0}\Delta t + (1 + a_{p_0,j_0}\Delta t) d^{q+1}_{p_0,j_0} - b_{p_0,j_0} d^{q+1}_{p_0+1,j_0}\Delta t \\ &- e_{p_0,j_0} d^{q+1}_{p_0,j_0-1}\Delta t - f_{p_0,j_0} d^{q+1}_{p_0,j_0+1}\Delta t = (1 - \lambda\Delta t\hat{I}) d^{q}_{p_0,j_0} + \lambda\Delta t\mathcal{I} d^{q}_{p_0,j_0} \geq 0.\end{aligned}$$

□

Remark 4.4.2. It is worth noting that the resulting system for the PD function is dense due to the jump integral term, adding to the computational complexity of the scheme. Hence, additional methods such as implicit handling of the jump term and/or Crank-Nicolson schemes can be useful. We omit these methods from the present work, as they are not the main focus, however we refer the interested reader to relevant research, such as d'Halluin, Forsyth, and Vetzal, 2005, Jwo, 2020 and Carr and Mayo, 2007.

Remark 4.4.3. As previously mentioned, for the credit risk modelling tasks we will consider, using either the regime switching or the stochastic volatility model suffices. We will see multiple such examples in the following sections 4.5 and 4.5.2. Similar calculations as those considered above can be applied to PIDE (3.56), for the estimation of the survival probability under the generalized model. Stability and monotonicity follow from combining Lemmata 4.3.1 and 4.4.1. However, a mentioned, the combination of the regime switching and stochastic volatility variables lead to an intractable numerical scheme, plagued with the "curse of dimensionality".

4.5 Applications in credit risk

4.5.1 IFRS 9 provision calculations

As discussed, the IFRS 9 framework requires practitioners to take into consideration multiple risk factors and their evolution for provision calculations and other modelling tasks. Naturally, the evolution of the PD is of paramount importance in these credit risk problems. Specifically for provisioning, using the PD function we can now estimate provisions for Stage 1 and Stage 2 exposures. Recall that financial institutions must account for additional provisions for exposures which display a significant increase in credit risk. These forward-looking lifetime provisions must be calculated per exposure, with some minor differences depending on the type of portfolio (e.g., for corporate loan portfolios many consider contamination effects). In this section, we display how the framework outlined above can be used to calculate the provisions under IFRS 9. The main contribution is the calculation of Expected Lifetime provisions for Stage 2 exposures which, as previously stated, is a novel requirement introduced by the these regulatory standards. We provide specific examples of provision calculations for each case below.

These calculations depend on multiple risk parameters corresponding to the credit exposure; the PD and Loss Given Default (LGD), as well as the amortization schedule, which affects the Exposure at Default (EAD), i.e., the remaining value of the loan which is not repaid in the case of default. Naturally, these risk parameters may vary according to each application and case. For example, many consider the LGD to evolve according to some stochastic process with a correlation to the

PD (see e.g., Miu and Ozdemir, 2006, Witzany, 2011). As our main focus is the PD function, we will consider a constant LGD and typical amortization schedule under the assumption of a zero interest rate in the examples that follow. The methodology, however, can be generalized to also consider a an appropriate LGD function (or stochastic process) and any type of amortization.

4.5.1.1 Stage 1 provisions

For Stage 1 loans standard regulations apply and we need only consider provisions as the Expected Losses (EL) that can be incurred on the current exposure. This calculation is given by the simple formula:

$$EL := \mathbb{E}[L_t] = EAD_t LGD_t PD_t. \tag{4.28}$$

Using the implicit numerical schemes we can calculate the PD value representing the probability the loan defaults within some fixed time t, represented straightforwardly by

$$PD_t = \Psi(x, t).$$

For example, the probability of a default event occurring within the current unit of time (typically year) is $PD_1 = \Psi(x, 1)$. An example of the provision calculation for varying initial positions is given below.

Example 4.5.1. Consider a Stage 1 loan, with asset process given by:

$$dG_t = k(\theta - G_t)dt + \sigma dB_t + \int_{\mathbb{R}} zN(dt, dz), \;\; G_0 = x, \tag{4.29}$$

where $(k, \theta, \sigma) = (0.5, 3.5, 2.0)$, the Compound Poisson Process has normally distributed jumps, with size $Z \sim N(0.0, 0.2)$ and rate $\lambda = 1.0$. We select $\mathcal{D} = [-10, 10]$, with $S = 8.0$, as estimated by Monte Carlo experiments, $N = 1001$ and $T = 101$. The resulting survival probability is graphed in Fig. 4.1. Then, depending on the initial position of the asset at the time of loan origination, the provisions are calculated as $EL = 100 \cdot PD \cdot 75\%$. In Table 4.1 we present the results for some initial positions $x \in [0, 1]$.

Initial Position	$PD_1(\%)$	$LGD(\%)$	Provisions (%)
0.0	100.00	75	75.00
0.1	83.24	75	62.43
0.2	68.22	75	51.17
0.3	55.01	75	41.26
0.4	43.64	75	32.73
0.5	34.06	75	25.55
0.6	26.16	75	19.62
0.7	19.78	75	14.84
0.8	14.73	75	11.05
0.9	10.82	75	8.11
1.0	7.83	75	5.87

TABLE 4.1: Provision calculations for Stage 1 loan, given initial position of the asset process.

Hence, if, at the time of calculation, the asset process is estimated to start at $x = 0.6$, the provisions are 19.62% of the current exposure. ◁

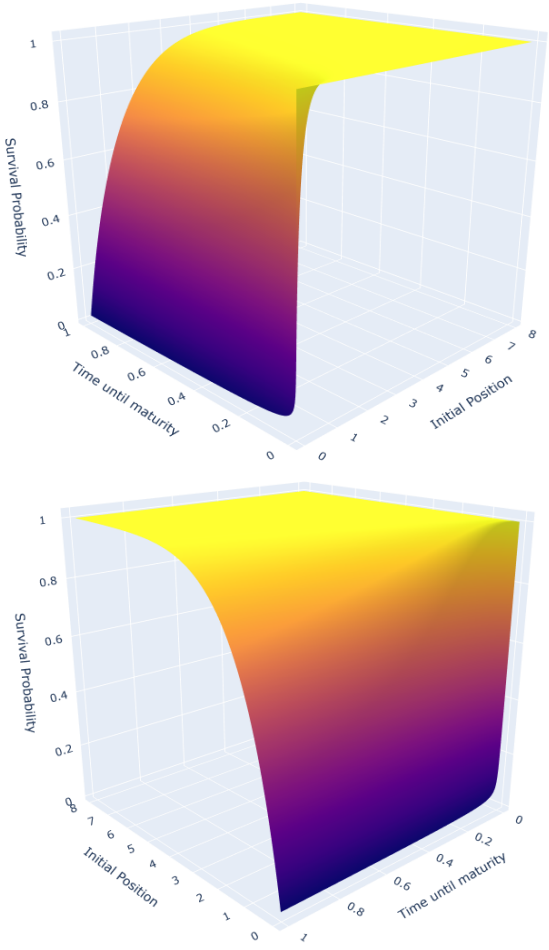

FIGURE 4.1: Survival probability for asset process with $(k, \theta, \sigma) = (0.5, 3.5, 2.0)$, using the BTCS scheme with $J = 150$.

As expected, the provisions are a decreasing function of the initial position. A similar table can be produced at any point during the lifetime of the loan, by estimating the corresponding PD values.

4.5.1.2 Stage 2 Provisions

We now turn to loan provision calculations for Stage 2 loans. Under IFRS 9, if and when loans transition to Stage 2, the lender is obligated to consider all future losses for provisioning purposes. Hence, at any time t, and assuming discrete amortization payments, the following formula for the expected losses occuring at some time $i > t$ applies:

$$\mathbb{E}[L_i] = \mathbb{E}\Big[\frac{1}{(1+r_i)^{i-t}} LGD_i PD_i^{PiT} EAD_i | \mathcal{F}_t\Big]. \tag{4.30}$$

The corresponding formula for the Lifetime Expected Credit Losses (ECL - also referred to as Expected Lifetime Provisions) at time t is given by:

$$ECL_t = \mathbb{E}\Big[\sum_{i=t+1}^{T} \frac{1}{(1+r_i)^{i-t}} LGD_i PD_i^{PiT} EAD_i | \mathcal{F}_t\Big], \tag{4.31}$$

where r_i is the interest rate at time i and T the maturity. In the above, PD_i^{PiT} represents the conditional Point-in-Time PD, which is the probability of default occurring at a given future time period. Specifically, we define:

$$PD_u^{PiT} = \mathbb{P}\Big(\inf_{r\leq u} G_r^x \leq 0, \inf_{r\leq u-1} G_r^x > 0\Big). \tag{4.32}$$

In order to calculate the PD^{PiT} in terms of the PD function resulting from the solution of the PIDE, we note that:

$$\mathbb{P}\Big(\inf_{r\leq u} G_r^x \leq 0\Big) = \mathbb{P}\Big(\inf_{r\leq u-1} G_r^x \leq 0\Big) + \mathbb{P}\Big(\inf_{r\leq u} G_r^x \leq 0, \inf_{r\leq u-1} G_r^x > 0\Big),$$

and hence:

$$PD_u^{PiT} = \Psi(x,u) - \Psi(x,u-1) = \Phi(x,u-1) - \Phi(x,u).$$

Therefore, (4.31) can now be written as:

$$ECL_t = \sum_{i=1}^{T} \big(\Phi(x,i-1) - \Phi(x,i)\big)\mathbb{E}\Big[\frac{LGD_i EAD_i}{(1+r_i)^{i-t}}|\mathcal{F}_t\Big]. \tag{4.33}$$

Example 4.5.2. Consider a credit exposure with asset process as in Example 4.5.1. However, we now suppose the exposure has been transferred to Stage 2, with remaining maturity $T = 10$ years. We also consider that the asset process of the borrower is currently estimated at $x = 1.80$. To estimate the Stage 2 provisions we require the PD function and use (4.33) (as mentioned, we consider $r = 0$ for simplicity):

Time until maturity (u)	EAD_u	$PD_u(\%)$	$PD_u^{PiT}(\%)$	$LGD(\%)$	EL_t
10.0	100	21.69	1.59	75	1.19
9.0	90	20.10	1.76	75	1.19
8.0	80	18.34	1.97	75	1.18
7.0	70	16.37	2.20	75	1.16
6.0	60	14.17	2.49	75	1.12
5.0	50	11.68	2.81	75	1.05
4.0	40	8.87	3.10	75	0.93
3.0	30	5.77	3.14	75	0.71
2.0	20	2.63	2.24	75	0.34
1.0	10	0.39	0.39	75	0.03
ECL					8.89

TABLE 4.2: Provision calculations for Stage 2 loan, given an asset process with initial position $x = 1.80$.

The exposure and expected losses are in percentages of the remaining exposure. As shown, the current Lifetime provisions are given by the sum of the final column: $ECL = 8.89\%$. ◁

In practice, we expect that when an exposure is classified as Stage 2, the parameters of the underlying asset process may differ, compared to the Stage 1 counterpart. In the example above, we purposely consider the same asset process so as to highlight the differences in the final provision estimations.

4.5.1.3 Provisions under the regime switching model

As discussed, the new regulatory framework aims to ensure that all financial institutions have accounted for future losses and abrupt changes in credit risk parameters, which can create severe losses and subsequent liquidity and solvency issues, both for institutions and their customers. As risk classification is widely considered a Markov process both in theory and by practitioners, considering transiton probabilities for loans can allow us to forececast PD values and estimate worst-case scenario provisions for loan exposures. We note that, in practice, estimating the parameters of the asset prices under each regime may be difficult. However, many financial institutions consider such models and its mathematical framework is well established, see e.g., Chatterjee, 2015 and Bruche, 2005. Another approach is to use historical parameters from Stage 1 and Stage 2 loans to estimate the changes that occur when a loan transitions between Stages. For this example, we will be estimating the provisions under the regime switching model developed above. To this end, we consider an IFRS 9 compliant transition matrix:

$$P = \left(\begin{array}{c|ccc} \text{IFRS 9 Rating} & \text{Stage 1} & \text{Stage 2} & \text{Stage 3} \\ \hline \text{Stage 1} & p_{11} & p_{12} & p_{12} \\ \text{Stage 2} & p_{12} & p_{22} & p_{13} \\ \text{Stage 3} & 0 & 0 & 1 \end{array} \right).$$

For a loan originating in Stage 1, we can now forecast credit losses by taking into account the probability of a SICR (significant increase in credit risk) event. Under the regime switching model, in the case of a transition to another Stage, we will need to estimate the PD values for the asset process governed by the new parameters. Using the straightforward notation Φ^i or PD^i to emphasize the Stage (regime) under which the specific PD value is estimated, we can then define the "*Stage-weighted provisions*", given by:

$$WP_t := p_{11} EAD_t LGD_t PD^1 + p_{12} \sum_{i=t}^{T} \left(\Phi^2(x, i-1) - \Phi^2(x, i) \right) \mathbb{E}\left[\frac{LGD_i EAD_i}{(1+r_i)^{i-t}} | \mathcal{F}_t \right] + p_{13} EAD_t LGD_t, \tag{4.34}$$

where the third term occurs in the case of default (i.e. transition to Stege 3), we have that $PD = 100\%$. This calculation holds for the case where we consider that the transition to Stage 2 occurs one period (e.g., year) after. However, we can also consider the cases where the deterioration occurs at any point $k > t$. For this calculation we require the k-step transition matrix of the underlying rating process, which is known to be P^k, whose elements will be symbolized as below:

$$P^k = \left(\begin{array}{c|ccc} \text{IFRS 9 Rating} & \text{Stage 1} & \text{Stage 2} & \text{Stage 3} \\ \hline \text{Stage 1} & p_{11}^k & p_{12}^k & p_{12}^k \\ \text{Stage 2} & p_{12}^k & p_{22}^k & p_{13}^k \\ \text{Stage 3} & 0 & 0 & 1 \end{array} \right).$$

with the understanding that p_{ij}^k represents the k-th step transition probability. We then have:

$$\mathbb{E}[WP_k|\mathcal{F}_t] = p_{11}^k EAD_k LGD_k PD_k^1 + p_{12}^k \sum_{i=k}^{T} \big(\Phi^2(x,i-1) - \Phi^2(x,i)\big)\mathbb{E}\Big[\frac{LGD_i EAD_i}{(1+r_i)^{i-t}}|\mathcal{F}_t\Big] + p_{13}^k EAD_t LGD_t. \tag{4.35}$$

At any point, with the dynamics of the underlying Markov process, we can obtain the corresponding WP_t values and calculate the above *Expected Stage-weighted provisions*. This calculation takes the future evolution of the loan, as well as the regime into consideration to provide an estimation that incorporates all scenarios. For illustrative purposes, we consider the example below.

Example 4.5.3. Consider an asset process governed by the regime-switching model below:

$$dG_t = \begin{cases} k_1(\theta_1 - G_t)dt + \sigma_1 dB_t + \int_{\mathbb{R}} zN(dt,dz), \ G_0 = x, \text{ if } R_t = \text{Stage 1}, \\ k_2(\theta_2 - G_t)dt + \sigma_2 dB_t + \int_{\mathbb{R}} zN(dt,dz), \ G_0 = x, \text{ if } R_t = \text{Stage 2}, \\ k_3(\theta_3 - G_t)dt + \sigma_3 dB_t + \int_{\mathbb{R}} zN(dt,dz), \ G_0 = x, \text{ if } R_t = \text{Stage 3}, \end{cases} \tag{4.36}$$

with regime-specific parameters given in Fig. 4.2 and $\mathcal{D} = [-6.0, 6.0]$, with $S = 4.0$ (we can also consider a different limit value S for each regime. However, in this example the Monte Carlo estimates indicate that the same value suffices). We consider normally distributed jumps with $Z \sim N(0.0, 0.5)$, rate $\lambda = 1.0$. We have set $N = 1001, T = 1001$ for the space and time grids, respectively. Furthermore, the generator matrix of the underlying Markov process given by:

$$Q = \begin{pmatrix} -0.5 & 0.3 & 0.2 \\ 0.3 & -0.6 & 0.3 \\ 0.0 & 0.0 & 0.0 \end{pmatrix}.$$

The graphs in Fig. 4.2 display the estimated survival probability in each regime (Stage), resulting from solving scheme (4.19).

We consider an initial position of $x = 0.3$ and maturity $T = 10$. The transition matrix of the underlying Markov process is obtained by calculating $P = \exp(Q)$:

$$P = \left(\begin{array}{c|ccc} \text{IFRS 9 Rating} & \text{Stage 1} & \text{Stage 2} & \text{Stage 3} \\ \hline \text{Stage 1} & 0.63 & 0.18 & 0.19 \\ \text{Stage 2} & 0.18 & 0.57 & 0.25 \\ \text{Stage 3} & 0.00 & 0.00 & 1.00 \end{array}\right). \tag{4.37}$$

We will perform the provisioning scenario analysis by forecasting the Stage-weighted provisions, given by (4.35) for the next four years. We first calculate the $k-$step transition matrices:

$$P^2 = \begin{pmatrix} 0.43 & 0.21 & 0.36 \\ 0.21 & 0.36 & 0.43 \\ 0.00 & 0.00 & 1.00 \end{pmatrix}, \quad P^3 = \begin{pmatrix} 0.31 & 0.20 & 0.50 \\ 0.20 & 0.24 & 0.56 \\ 0.00 & 0.00 & 1.00 \end{pmatrix}, \quad P^4 = \begin{pmatrix} 0.23 & 0.17 & 0.60 \\ 0.17 & 0.18 & 0.66 \\ 0.00 & 0.00 & 1.00 \end{pmatrix}.$$

At time $t = 0$ we consider the forward looking scenarios and can calculate the Stage 1 and Stage 2 provisions. Recall that Stage 1 provisions are given by (4.28). Stage 2 (expected lifetime) provisions are given in column ECL_t in Table 4.3 below, which

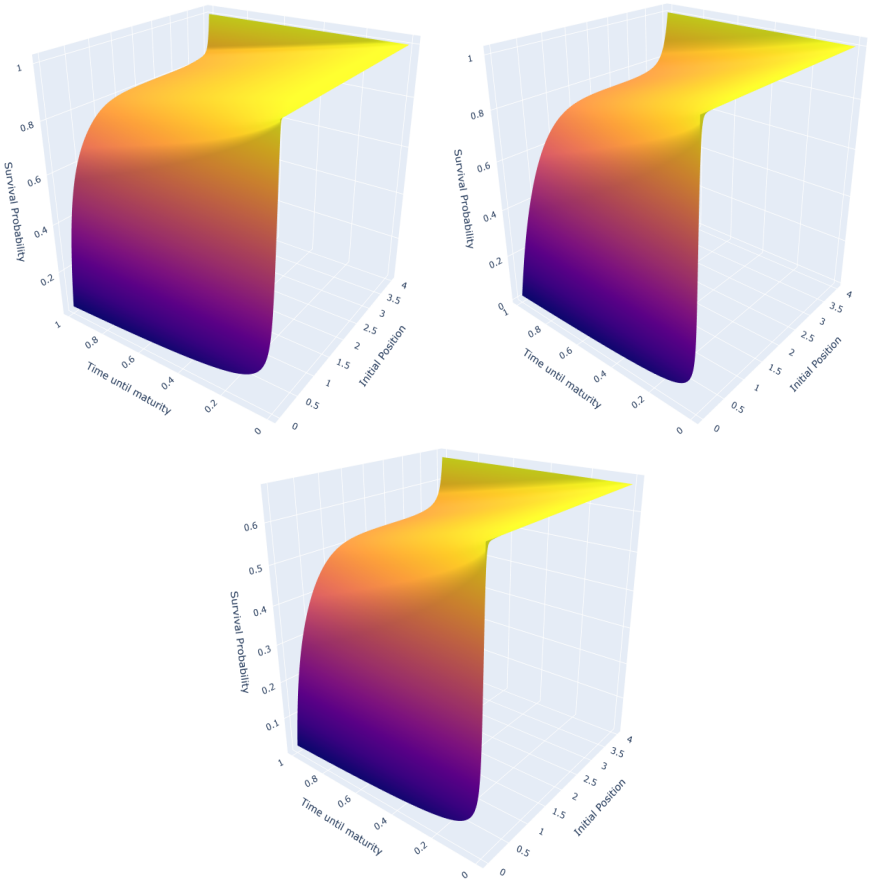

FIGURE 4.2: Survival probability for (4.36) with $(k_1, k_2, k_3) = (0.3, 0.2, 0.0), (\theta_1, \theta_2, \theta_3) = (0.8, 0.5, 0.0), (\sigma_1, \sigma_2, \sigma_3) = (0.3, 0.5, 0.0)$, with $R_0 =$ Stage 1 (top left), $R_0 =$ Stage 2 (top right). The average survival probability across all regimes is also shown (bottom), which also account for the survival probability when $R_0 =$ Stage 3, for which we have $\Phi(x, u) \equiv 0$.

also contains the Point-in-Time Stage 1 and Stage 2 PD required to calculate the provisions.

Time until maturity (u)	EAD_u	$\Psi^1(x,u)(\%)$	$\Psi^2(x,u)(\%)$	Stage 1 $PD_u^{PiT}(\%)$	Stage 2 $PD_u^{PiT}(\%)$	$LGD(\%)$	EL_u
10.0	100	4.76	10.55	0.52	1.53	75%	1.147
9.0	90	4.24	9.02	0.50	1.50	75%	1.012
8.0	80	3.74	7.52	0.48	1.44	75%	0.864
7.0	70	3.26	6.08	0.48	1.33	75%	0.698
6.0	60	2.78	4.75	0.47	1.17	75%	0.527
5.0	50	2.31	3.58	0.47	0.99	75%	0.371
4.0	40	1.84	2.59	0.46	0.80	75%	0.240
3.0	30	1.38	1.79	0.46	0.65	75%	0.146
2.0	20	0.92	1.14	0.47	0.58	75%	0.087
1.0	10	0.45	0.56	0.45	0.56	75%	0.042

TABLE 4.3: Stage 1 and 2 PDs and expected losses.

As shown in Example 4.5.2, the Lifetime Provisions can be calculated by the sum of the expected losses column, EL_u,. We can now calculate the Stage 1 and Stage 2 provisions at each subsequent time period, which we will use to calculate the Stage-weighted provisions for the next four years, calculated by (4.35). The results are shown in Table 4.4, where the final column below contains the Stage-weighted provisions.

Time until maturity (u)	EAD_u	Stage 1 Provisions	ECL	Stage 3 Provisions	WP_u
10.0	100	0.390	5.13	75.00	15.42
9.0	90	0.338	3.99	67.50	25.28
8.0	80	0.288	2.98	60.00	30.69
7.0	70	0.247	2.11	52.50	31.92

TABLE 4.4: Stage-weighted provision calculations for the next 4 years.

The large difference observed even between the Lifetime and Stage-weighted provisions is evidence of the importance of such scenario analysis in provision calculations. Particularly in cases similar to that examined in this example, where the probability of transitioning to a default state is quite high the results can have an extremely large effect, which risk managers must account for in risk and provisioning policies. ◁

4.5.2 Further Applications in credit risk modelling

4.5.2.1 Pricing of Credit Default Swaps

Another financial field in which the PD function plays a paramount role is credit derivatives pricing. In particular, we consider the fair price of Credit Default Swap (CDS). A default swap is a contract that protects the holder of an underlying swap from the losses caused by the default to the obligation's issuer. Therefore, the evolution of the PD values can be used for the pricing, hedging and managing of such options. Extensive work has been done on modeling and pricing CDSs, such as in Cariboni and Schoutens, 2007 and Houweling and Vorst, 2005. Specifically, it can be shown that the price of the CDS is given by:

$$CDS = (1-R)\left(-\int_0^T e^{-rs} d\Phi(x,s)\right) - c\int_0^T e^{-rs}\Phi(x,s)ds,$$

and the corresponding par spread:

$$c^* = \frac{(1-R)\left(-\int_0^T e^{-rs} d\Phi(x,s)\right)}{\int_0^T e^{-rs}\Phi(x,s)ds},$$

where R is the specific recovery rate and r is the risk-free rate. The above expression can be discetized as follows:

$$c^* = \frac{(1-R)\sum_{i=1}^n e^{-rt_i}(\Phi(x,t_{i-1}) - \Phi(x,t_i))}{\frac{1}{2}\sum_{i=1}^n e^{-rt_i}(\Phi(x,t_{i-1}) + \Phi(x,t_i))\Delta t_i}, \tag{4.38}$$

where the Trapezoidal rule has been used for the discretization of the denominator. Estimating the price and par rate of CDS therefore requires the term structure of the underlying risk-free and survival probability processes. We present a simplified example, whereby the interest rate is again considered to be zero.

Example 4.5.4. Consider a CDS with maturity $T = 10$ years and recovery rate $R = 0.5$, where the asset process evolves according to the following stochastic volatility model:

$$\begin{cases} dG_t = k(\theta - G_t)dt + \sigma(Y_t)dB_t + \int_{\mathbb{R}} zN(dt,dz), & G_0 = x \\ dY_t = \kappa(\mu - Y_t)dt + \xi\sqrt{Y_t}dW_t, & Y_0 = y. \end{cases}$$

The parameters of the stochastic model are given in Fig. 4.3, we let $\mathcal{D} = [-5.0, 75.0]$ and consider a spatial and temporal discretization with 200 and 1000 steps respectively. Jumps are again normally distributed, with size $Z \sim N(0.3, 0.5)$, rate $\lambda = 1.0$ and $J = 90$. Furthermore, we set a grid with 200 steps for the volatility $\mathcal{V} = [0.0, 200.0]$. The parameters and resulting graph of the PD function can be seen in Figure 4.3.

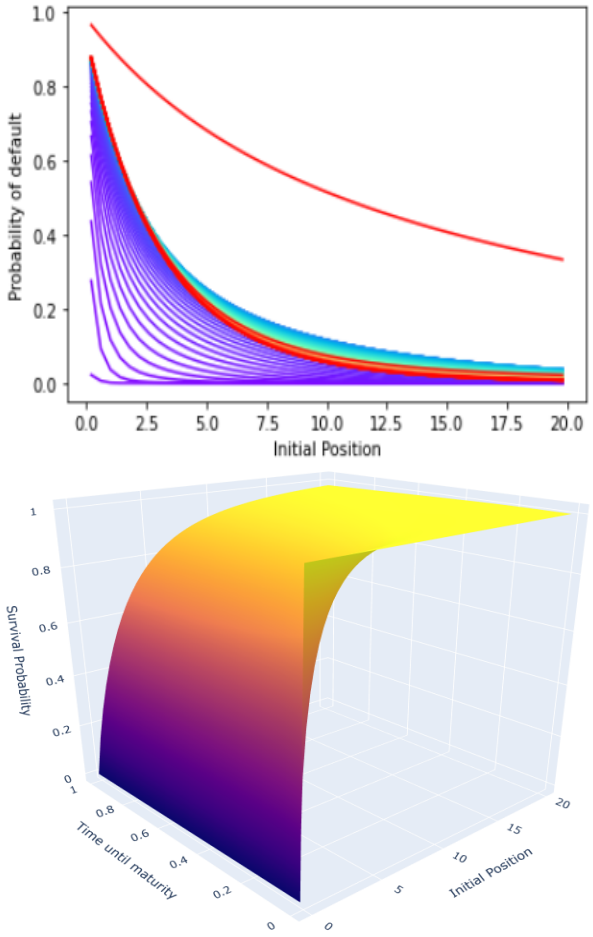

FIGURE 4.3: (Left) Evolution of the PD under the stochastic volatility model with $(k, \theta, \kappa, \mu, \xi) = (2.0, 2.0, 0.05, 0.1, 0.07)$, for various values of the starting volatility Y_0. (Right) The average survival probability across all volatility values.

We assume an initial position of $x = 3.0$ and plot the evolution of the average survival probability in Fig. 4.4. The resulting par spread is calculated using (4.38) to obtain $c^* = 0.33$.

◁

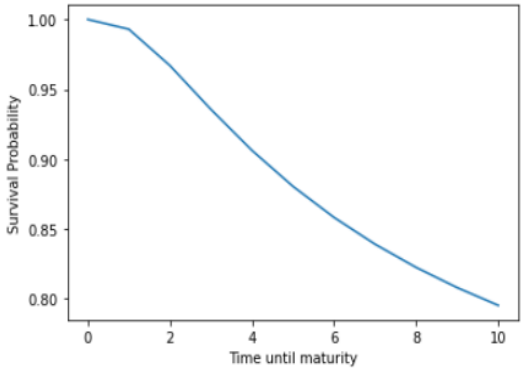

FIGURE 4.4: Evolution of the average survival probability under the stochastic volatility model described in Example 4.5.4, with starting point $x = 0.3$.

4.5.2.2 Credit Portfolio Optimization

For many financial institutions, one of the most important tasks is securitization of credit exposures. Ultimately, this can be formulated as an optimization problem. The PD function affects the risk of each exposure and, by extension, the corresponding return as well. To this end, we present a simple example to show how such an optimization problem can be solved under the stochastic volatility PD model.

Example 4.5.5. Suppose a securitization agency creates a portfolio consisting of loans (or credit derivatives), each with different underlying asset process. The resulting PD functions will differ depending on the loan's (or derivative's) characteristics and asset value processes. Slightly abusing notation, we suppose that, for a portfolio of three loans, the corresponding PD functions are given by $PD_i, i = 1, 2, 3$, estimated using the methodology developed above.

The agency aims to select the investment allocated to each of the credit exposures. Specifically, it poses the following portfolio optimization problem: suppose $w_i, i = 1, 2, 3$ and $r_i, i = 1, 2, 3$ represent the weight of total investment allocated to each institution's set of loans and their average return, respectively. Consider, furthermore, that the required portfolio rate of return is set to be R^*. For the credit exposure i, at time t, the expected loss is given by $EL_t^i = EAD_t^i LGD_t^i PD_t^i$, and we can then define the total loss function for the agency by $L(t) = \sum_1^3 w_i EL_t^i$. In order to rebalance the portfolio at each period, the securitization agency is then interested in solving a portfolio optimization problem (we present a very simple such problem, which can be solved analytically to illustrate the use of the method). The optimization we consider is the following:

$$\begin{aligned} \min_{\mathbf{w}} \mathbb{E}[U(L(t))], \text{ subject to} \\ w_1 r_1 + w_2 r_2 + w_3 r_3 = R, \\ w_1 + w_2 + w_3 = 1, \end{aligned}$$

for an appropriate loss function U. While the following analysis can be extended to any convex loss function, for the sake of simplicity, we illustrate the calculation selecting the quadratic loss function $U(L) = bL^2 - L$ (in the sense of a negative utility function). To standardize the optimization problem, we consider that EAD_i

is given as a percentage of the original loan value and for simplicity we consider a constant $LGD_t^i = 1$ for $i = 1, 2, 3$ and all t. At any point in time t, the expected loss utility is then:

$$\sum_{i=1}^{3} U(w_i EAD_i) PD_i^{PiT},$$

where PD_i^{PiT} is the current Point-in-Time default probability corresponding to exposure i. The agency must optimize the portfolio by solving:

$$\begin{aligned} \text{minimize } f(w_1, w_2, w_3) := \sum_{i=1}^{3} b(w_i EAD_i)^2 - w_i EAD_i, \text{ subject to} \\ w_1 r_1 + w_2 r_2 + w_3 r_3 = R, \\ w_1 + w_2 + w_3 = 1, \end{aligned}$$

This simple quadratic optimization problem can now be solved either analytically or numerically. As in Example 2.5.6, we can calculate:

$$w_3^* = \frac{PD_1^{PiT}(2bEAD_1^2\delta\epsilon - \epsilon) + PD_2^{PiT}(\gamma - 2baEAD_2^2\beta\gamma) - PD_3^{PiT}}{2(bPD_1^{PiT}EAD_1^2\epsilon^2 + bPD_2^{PiT}EAD_2^2\gamma^2 + bPD_3^{PiT}EAD_3^2)},$$

where $\beta = \frac{R-r_1}{r_2-r_1}, \gamma = \frac{r_3-r_1}{r_2-r_1}, \delta = \frac{r_2-R}{r_2-r_1}$ and $\epsilon = \frac{r_3-r_2}{r_2-r_1}$. A straightforward substitution using the two conditions will result in the corresponding values w_1^* and w_2^*. We consider the above setting with average returns from each institution's instruments $r = (r_1, r_2, r_3)^T$ and current exposures $EAD = (EAD_1, EAD_2, EAD_3)^T$ given by:

$$r = (0.1 \ \ 0.3 \ \ 0.1)^T, \quad EAD = (0.9 \ \ 0.8 \ \ 0.7)^T.$$

In order to obtain the vector containing the PD values, we consider the three asset classes described by the processes below:

$$\begin{aligned} dG_t^1 &= k_1(\theta_1 - G_t^1)dt + \sigma_1 dB_t + \int_{\mathbb{R}} zN(dt, dz), \quad x_0 = 1.00 \\ dG_t^2 &= k_2(\theta_2 - G_t^2)dt + \sigma_2 dB_t + \int_{\mathbb{R}} zN(dt, dz), \quad x_0 = 0.20 \\ dG_t^3 &= k_3(\theta_3 - G_t^3)dt + \sigma_3 dB_t + \int_{\mathbb{R}} zN(dt, dz), \quad x_0 = 0.50, \end{aligned}$$

with $(k_1, k_2, k_3) = (0.5, 0.8, 0.5), (\theta_1, \theta_2, \theta_3) = (3.5, 3.0, 2.5), (\sigma_1, \sigma_2, \sigma_3) = (2.0, 1.5, 2.5)$, $J = 150, \lambda = 1.0$ and jump distributions $Z \sim N(0.0, 0.2)$ for all three. Solving PIDE (3.63), we obtain the PD^{PiT} values:

$$PD^{PiT} = (0.0783 \ \ 0.1447 \ \ 0.0447)^T,$$

and fixing the expected total return to be $R = 25.00\%$, the resulting optimal weights $w^* = (w_1^* \ \ w_2^* \ \ w_3^*)^T$ are:

$$w^* = (0.163 \ \ 0.750 \ \ 0.087)^T.$$

◁

For extensive work on portfolio optimization problems with defaultable assets, we refer the interested reader to e.g., Asanga et al., 2014. Furthermore, empirical studies of the applicability of standard ruin probabilities in practice can be found in

Braun, Schmeiser, and Schreiber, 2015. In the example above, we focus on a case where an agency must assess and optimize a portfolio of loan exposures with varying characteristics. Such cases could be loans originating in different sectors; e.g., in Pasricha et al., 2020a, the authors consider a portfolio of risky bonds originating from the Industry and Service sector.

4.6 Conclusion

In this Chapter we developed and studied Finite Difference techniques for solving the PIDEs and obtain the corresponding PD functions. As shown, the FD approach can be used efficiently to estimate the PD values, but careful handling is needed to ensure that the required properties (stability and monotonicity) are satisfied. Furthermore, we have seen that by using the ADI scheme more complex PIDEs can be solved (e.g., (3.41), under the stochastic volatility model), which opens the door to a large family of asset value models that can be used in various applications. As mentioned, the FD schemes also have important advantages over Monte Carlo methods, as we obtain the PD as a function of both the initial position and time until maturity, as well as of any latent (regime or volatility) variables. On the other hand, the corresponding Monte Carlo estimate requires simulating sufficient paths of the asset value process, for a given combination of the variables. This is obviously a computationally expensive process, and is one of the reasons why FD methods are in preferred in recent literature.

We posit that the schemes developed in this Chapter can be useful to practitioners, since, as we have shown, the provisioning problems that arise due to IFRS 9 can be efficiently tackled using the resulting PD evolution. Additionally, taking advantage of the efficiency of these numerical methods, we have seen that additional calculations can be performed related to IFRS 9, such as Stage-weighted provisioning, as well as other problems in credit risk modelling. Overall, the FD schemes presented in this Chapter are tools that can provide a robust framework for PD estimation and be implemented in provisioning and credit risk policies.

Finally, despite the aforementioned advantages, we have also seen that the FD schemes have important limitations. Specifically, the approach suffers from the "curse of dimensionality", making it intractable in cases of higher dimensional asset value models. This may pose a significant issue, since the handling of models of increased complexity has become necessary in the context of modern credit risk modelling under IFRS 9. In addition, the goal of the numerical methods is to, ideally, estimate the PD for any value of the input variables. However, even though the FD scheme provides PD values as a function of the inputs, as mentioned, this holds only at the exact points on the grid, and hence assumptions must be taken in order to estimate the PD at other points. Both these limitations will be addressed using the field of Machine Learning models, which have gained significant popularity in recent years. In the next Chapter, we will develop Neural Network models to estimate the PD functions, and see how these compare to the FD schemes, both from a theoretical and practical point of view.

Chapter 5

A practical Deep learning framework for Probability of Default modelling

5.1 Background

In recent years many important advancements have been made in the application of Machine Learning (ML) techniques to Ordinary and Partial Differential Equations (ODE & PDE). In particular, seminal work has considered Artificial Neural Networks (ANN) with multiple hidden layers to solve such equations with impressive success and accuracy. This has given rise to a new field of Deep Learning for ODEs and PDEs. Various approaches have been considered but the most widely adopted currently are the Physics Informed Neural Networks (PINN), developed in Raissi, Perdikaris, and Karniadakis, 2019 and further studied and applied in Cai et al., 2021; Pang, Lu, and Karniadakis, 2019 and Cuomo et al., 2022. Furthermore, in recent work (see Frey and Köck, 2022), new methods have been proposed regarding the architecture and training process, relying on the celebrated Feynman-Kac formula to construct the loss function.

Given the growing interest in such Deep Learning techniques, we will be considering these approaches in the context of the default probability estimations for asset value processes driven by appropriate stochastic processes as defined in Chapter 3. In particular, we continue to consider the assets of a debtor to follow a jump diffusion process, and define the Probability of Default (PD), for an appropriate time until maturity and threshold. As we have seen, this problem is of great importance in modern credit risk modelling under IFRS 9. As the framework set by IFRS 9 requires financial institutions to forecast future credit losses that could be incurred, and consider scenario analyses based on mathematically-sound models, it is of paramount importance that practitioners consider various modelling techniques. To this end, we will use recent developments in the application of Deep Neural Networks to PDEs and apply such methods to the PIDEs obtained for the PD processes in Chapter 3. This can be done for all the models previously considered, as we will detail in this Chapter.

In addition to providing an alternative to the FD schemes constructed in Chapter 4, Deep Neural Networks (DNNs) have the important advantage that they do not suffer from the "curse of dimensionality". This allows us to consider asset models of higher dimensions, such as the generalized model (recall that the generalized model combines the regime switching and stochastic volatility models). Therefore, DNNs and similar models can become useful tools for credit risk modelling, especially in cases with larger complexity. To showcase the importance of this capability, we will

use the DNN framework to tackle two modelling problems which are intractable using the previous FD schemes:

- PD calculation using the generalized asset value process given by (3.7).
- Developing an approach to estimate PD values given a family of stochastic processes the asset value may follow. This is particularly applicable under IFRS 9, since it is reasonable to consider that the underlying parameters of the asset value process change themselves as time evolves. Such changes might be caused by macroeconomic or industry factors, for example. We will see how a DNN model can be trained to estimate the evolution of the PD under this setting and use the model for provisioning scenario analyses.

We rely on the aforementioned recent work for the architecture of the Neural Network models and consider extensions to the regime switching, stochastic volatility and generalized models, and their applications to credit risk modelling under IFRS 9. Furthermore, we discuss model architecture, training parameters/algorithms and computational requirements for the practical implementation of Deep learning models versus the Finite Difference counterparts, as well as interesting findings from the DNN models that warrant further research. These are important considerations since practitioners must consider a range of factors when deciding which approaches to implement, from latent variables affecting the portfolio to computational power and recalibration periods.

5.1.1 Neural Network architecture and training

Neural Networks models, commonly referred to as simply Neural Networks, have risen to prominence in the Machine Learning and Artificial Intelligence fields in recent years, having been used in a vast amount of fields, spanning both theoretical and practical problems. Accounts of their vast applications can be found in e.g., Fadlalla and Lin, 2001; Papik et al., 1998 and Kumar and Thakur, 2012. In turn, the study of such models has lead to many different architectures such as Convolutional (CNN), which have been widely used in image recognition and classification (see e.g., Guo et al., 2017 and Hijazi, Kumar, Rowen, et al., 2015), or Recurrent (RNN) (Medsker and Jain, 2001), which can handle multiple other forms of data, such as text (Sutskever, Martens, and Hinton, 2011). In general, the structure of any NN model relies on the Multi-Layer Perceptron (MLP) architecture, described in the following definition.

Definition 5.1.1. Consider a multi-dimensional input vector $\mathrm{x} \in \mathbb{R}^n$, with dependent variable $y \in \mathbb{R}$. Then, a MLP feed-forward Neural Network, commonly referred to as a Deep Neural Network (DNN), with L hidden layers and activation function $g : \mathbb{R}^n \to \mathbb{R}^n$ consists of iterations of the following equations:

$$\begin{aligned} \mathrm{x}^{(0)} &= \mathrm{x} \\ \mathrm{x}^{(i)} &= g\left(W^{(i-1)}\mathrm{x}^{(i-1)} + b^{(i-1)}\right), \text{for all } 1 \leq i \leq L \\ y &= W^{(L)}\mathbf{x}^{(L)} + b^{(L)}, \end{aligned} \tag{5.1}$$

where the i-th hidden layer $\mathrm{x}^{(i)}$ is a vector of length n_i for $1 \leq i \leq L$, n_i is the size of the i-th hidden layer and the activation function is applied to each coordinate of its input, i.e.,

$$g(\mathbf{x}) = \left(g\left(x_1\right), g\left(x_2\right), \ldots, g\left(x_n\right)\right).$$

An illustration of the DNN architecture is given in Fig. 5.1. Of great importance is the selection of appropriate activation functions. It can be shown that non-linear activation functions are required in order to train DNNs to solve non-trivial problems, and, depending on the setting and framework, there exist many such functions that can be used. Some of the most commonly used sigmoid, hyperbolic tangent, Rectified Linear Unit (ReLU) and Gaussian. In depth studies on the selection of activation function can be found in e.g., Pratiwi et al., 2020 and Hayou, Doucet, and Rousseau, 2018. Finally, we remark that the other hyperparameters, such as the number of hidden layers and the nodes per layer have also been shown to affect model performance. Empirical and statistical studies can be found in e.g., Liao et al., 2022 and Eggensperger et al., 2013. Despite their obvious importance, selecting activation functions and model hyperparameters still largely remains an "art", relying heavily on the experience of the modeler, rather than a science in itself, given that robust techniques have not yet been established. We will discuss this further in subsequent sections in the context of credit risk modeling.

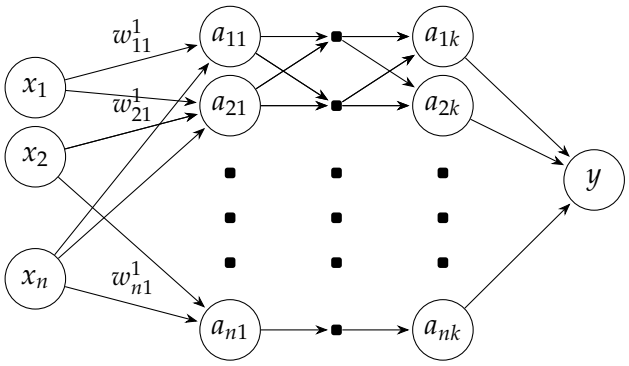

FIGURE 5.1: Fully connected Neural Network with an n-dimensional input layer, k dense layers.

The architecture and mathematical foundation of Neural Networks as machine learning models has been established since the first half of the 20th century. However, the lack of an efficient training algorithm, i.e., mathematical framework for the estimation of the trainable parameters $W^{(i)}$ and $b^{(i)}$, for these complex models meant that they could not be used for the wide range of problems for which they are theoretically applicable. The backpropagation algorithm (Hecht-Nielsen, 1992) was developed and able to efficiently solve the problem of training such a model. Simply put, backpropagation is used to train the model by calculating the errors in its predictions and updating the parameters in order to minimize the error. This result, coupled with the exponential increase in computational power in recent years have lead to significant breakthroughs in DNNs, their applications and research.

Backpropagation itself utilizes a gradient descent algorithm, which depends on the learning rate parameter, which affects the amount by which the parameters are altered during the training process. For most applications, decaying, step-wise learning rates are used, which is the approach we adopt in the following sections.

5.2 DNN models for solving PDE

In this section we consider two methods that can be used to train DNNs to calculate solutions of PDEs. In Blechschmidt and Ernst, 2021 a seminal overview of these

approaches is given and applied to various PDEs. Following the notation in this paper, we briefly introduce these methods below.

5.2.1 Physics Informed Neural Networks (PINNs)

Physics Informed Neural Networks, as its name suggests, takes into consideration the "physics" of the problem, meaning the fundamental properties of the process, as dictated by the analytic PDE (or PIDE) the process satisfies. This approach was introduced in Raissi, Perdikaris, and Karniadakis, 2019, and has since been used in a wide variety of applications. For example, in Mao, Jagtap, and Karniadakis, 2020 the authors consider using PINNs to model aerodynamic flows, and in Sahli Costabal et al., 2020 the authors employ PINN to assist in the diagnosis of atrial fibrillation.

To briefly describe the mathematical framework, consider a PDE of the form:

$$\begin{aligned} \partial_t u(x,t) + \mathcal{L}u(x,t) = 0, \qquad & (x,t) \in \mathcal{D} \times (0,T], \\ u(x,0) = u_0(x), \quad & x \in \mathcal{D}, \end{aligned} \tag{5.2}$$

where $\mathcal{L}$ is a (possibly nonlinear) differential operator. We consider bounded $\mathcal{D} \subset \mathbb{R}^d$ and initial values given by $u_0 : \mathcal{D} \to \mathbb{R}$. Boundary conditions can take on various forms with a standard choice being the Dirichlet case:

$$u(x,t) = u_b(x,t), \quad (x,t) \in \mathcal{D} \times (0,T]. \tag{5.3}$$

The goal is to train a DNN with a two-dimensional input (x,t) to "learn" the solution $u(t,x)$. To do this, we consider a loss function that relies on the fact that the dynamics of the function are governed by (5.2), thereby considering the "physics" of $u(x,t)$. To this end, we define the residual function corresponding to the PDE (or PIDE):

$$r_\theta(x,t) = \partial_t u(x,t) + \mathcal{L}u(x,t), \tag{5.4}$$

where θ represents the set of trainable parameters in the Neural Network model. Then, using the notation from Blechschmidt and Ernst, 2021, we will consider the collocation points in the interior of the domain $X^r := \{(x_i^r, t_i^r)\}_{i=1}^{N_r} \in \mathcal{D} \times (0,T]$, on the boundary $X^b := \{(x_i^b, t_i^b)\}_{i=1}^{N_b}$ and for the initial condition $X^0 := \{(x_i^0, t_i^0)\}_{i=1}^{N_0}$, which will be used for training, and the resulting Neural Network approximation of the solution is represented by $u_\theta(x,t)$. The total loss is then given by:

$$L_\theta(X) := L_\theta^r(X^r) + L_\theta^0(X^0) + L_\theta^b(X^b), \tag{5.5}$$

with the components encompassing the three important terms dictating the dynamics of the solution, i.e., the interior of the domain, along with the boundary and initial conditions:

- $L_\theta(X^r) = \frac{1}{N_r} \sum_{i=1}^{N_r} |r_\theta(x_i^r, t_i)|^2$,
- $L_\theta(X^b) = \frac{1}{N_b} \sum_{i=1}^{N_b} |u_\theta(x_i^b, t_b) - u(x_i^b, t_b)|^2$,
- $L_\theta(X^0) = \frac{1}{N_0} \sum_{i=1}^{N_0} |u_\theta(t_0, x_i^0) - u(0, x_i^0)|^2$.

Hence, minimizing the loss function (5.5) is equivalent to calculating $u_\theta(x,t)$ that best approximates $u(x,t)$ satisfying (5.2) with the corresponding initial and boundary conditions.

It is obvious that this approach is extremely useful, as well as convenient; one only needs to know the form of the PDE in order to train the DNN. It is worth noting, however, that difficulties may arise due to the need to explicitly estimate each derivative term in the PDE, in order to construct the loss function. Even though modern Machine Learning packages in Python such as *Tensorflow* (Abadi, 2016) and *PyTorch* (Paszke et al., 2019) provide methods for calculating derivatives, the approximation of integral terms in the case of PIDEs is more difficult. Hence, for the PIDEs in the case of the Lévy-driven processes as obtained and studied in Chapter 3, we will consider a method that relies on simulations of the underlying asset value process, detailed in what follows.

5.2.2 DNNs for PDEs using stochastic simulation

A different approach to training Neural Networks to solve PDEs has been studied in Beck et al., 2021 and further in the case of PIDEs in Frey and Köck, 2022. As mentioned in the overview provided by Blechschmidt and Ernst, 2021, this method is applicable to the well-known class of PDEs that arise as solutions to the Kolmogorov backward or forward equations of Itô processes satisfying the SDE:

$$dG_t = \mu(G_t, t)dt + \sigma(G_t, t)dW_t, \quad G_0 = x. \tag{5.6}$$

The corresponding backward equation is then given by:

$$\frac{\partial u}{\partial t} + \mu(x,t)\frac{\partial u}{\partial x} + \frac{1}{2}\sigma^2(x,t)\frac{\partial^2 u}{\partial x^2} = 0, \quad t \in [0,T], x \in \mathcal{D}, \tag{5.7}$$

$$u(x,T) = g(x,T). \tag{5.8}$$

Recall that to obtain the forward problem we simply use the change of variables $u := T - t$ to define the time until maturity, as seen in Chapter 3. This approach requires applying the Feynman-Kac formula in order to establish the relationship between the solution of the PDE and the underlying SDE, using the terminal condition. The Feynman-Kac representation can then be used to construct an appropriate loss function. To this end, consider the random variable $Y = g(G_T^x, T)$. Following Frey and Köck, 2022, the solution of (5.7), $u(x,t)$ can be written as:

$$u(x,t) = \mathbb{E}\big[g(G_T, T)|\mathcal{F}_t\big] = \mathbb{E}\big[g(G_T, T)|X_t = x\big], \tag{5.9}$$

where the equality is a result of the Markov property, and therefore we can write:

$$u(x,t) = \mathbb{E}[Y|\mathcal{F}_t]. \tag{5.10}$$

It then follows that, for a fixed time t, $u(x,t)$ is the solution of the minimization problem:

$$\min_u \mathbb{E}[|Y - u(x)|]. \tag{5.11}$$

Hence, to train a DNN model we can use the estimator of the expectation above as the loss function:

$$L_\theta(x) = \frac{1}{M}\sum_{i=1}^{M}\big(Y_i - u_\theta(x,t)\big)^2, \tag{5.12}$$

where $u_\theta(x,t)$ is the Neural Network approximation of the solution and Y_i is the random variable representing the payoff function, as described above, corresponding to the $i-th$ simulated asset process path. Training therefore consists of simulating M paths of the asset value process and calculating the payoff in order to construct the loss function and estimate the trainable model parameters. Notice how this construction only takes into account different values of the initial position x we therefore estimate the solution to the PDE only as a function of x, i.e., $u(x;t)$. We will extend this in order to produce the PDs as functions of time, as well, in the following sections.

One can easily observe important differences between the two approaches presented for training DNN models for the solutions of PDEs. Notably, in this approach we rely on stochastic simulation of the underlying process to generate training data. This means that explicit estimations of the derivatives and integral terms that arise in (5.7) are not required. This approach is also quite practical, as it only requires simulating paths of the underlying stochastic processes. Therefore, the simulation-based DNN can be easily developed for all the asset value models we have examined (with appropriate changes in the neural network architecture).

5.3 Simulation-based DNN models for PD processes

In this section we describe the development of the simulation-based Neural Network models for the estimation of the PD values. First, we apply the method to the one dimensional, regime switching and stochastic volatility models, highlighting again that this approach requires the imposition of an appropriate terminal condition at the time of maturity $t=T$, via the payoff function h. Hence, we will consider the backward formulation of the PIDEs obtained in Chapter 3, obtaining PDs as a function of the starting time t (considering fixed maturity T) rather than the remaining time until maturity u, was previously considered. For clarity, we recall this version of the definition of the PD process below.

Definition 5.3.1. Consider the Lévy-driven OU stochastic asset process $(G_t)_{t\geq 0}$, depending on variables $X^1, X^2, \ldots, X^N$ (which could be discrete or continuous). The corresponding survival probability function is then:

$$\Phi(x, x_0^1, \ldots, x_0^N, t) = \mathbb{P}\Big(\inf_{t\leq r\leq T} G_r > 0 | G_t = x, X_t^1 = x^1, \ldots, X_t^N = x^N \Big), \tag{5.13}$$

with corresponding Probability Default process:

$$\Psi(x, x_0^1, \ldots, x_0^N, t) = 1 - \Phi(x, x_0^1, \ldots, x_0^N, t). \tag{5.14}$$

Training the Neural Network now consists of simulating paths of the asset process $G_t, t\in[0,T]$, for various values of the initial position $x\in\mathcal{D}$ in order to calculate the payoff function, which is given by $Y := g(T, G_T^x) = \mathbb{1}_{\{\inf_{t\leq r\leq T} G_r^x \leq 0\}}$. Hence, we obtain the set of random variables Y, from many initial positions $x\in\mathcal{D}$, which will be used to train the Neural Network using the loss function given in (5.12).

Notice that, in the training process described above, fixed starting and maturity times t and T are considered and, hence, the resulting model will be able to estimate the PD only as a function of the initial position; this is similar to the process that one must apply when performing a Monte Carlo estimation, and is therefore sub-optimal, as we want to estimate PD values with variable time until maturity, creating a framework comparable to that of the Finite Difference schemes detailed

in Chapter 4. We can therefore improve this method by simulating paths for various values of the time until maturity $u = T - t$ (by changing either the starting or maturity time) when creating the training dataset and considering the DNN with a two-dimensional input layer (initial position and maturity time). In what follows, we start by considering a fixed maturity time to train a model with only one input, which we will use to gauge the best choice of model hyperparameters, and will subsequently extend the Neural Networks to account for the additional variables.

5.3.1 One dimensional input layer (initial position)

We begin by considering a fixed $t = 0$ and $T = 1.0$, and training a DNN model as a function of only the initial position. This requires a Neural Network model with a one-dimensional input and output layer (the initial position and probability of default, respectively), as shown in Fig. 5.2.

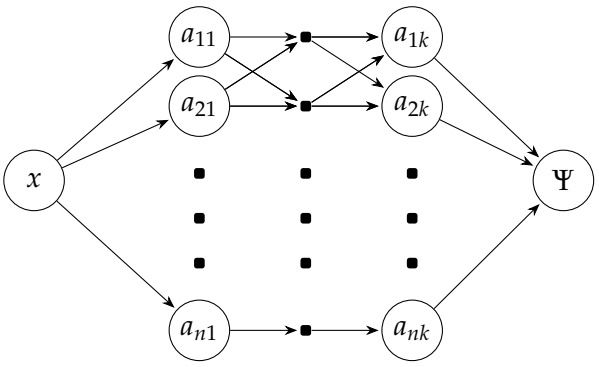

FIGURE 5.2: Fully connected Neural Network with a one-dimensional input layer and a one-dimensional output layer.

We do this for the one dimensional, as well as for the regime switching and stochastic volatility models (considering a fixed value of the initial regime and volatility value, respectively). Finding the best choice of hidden layers for this model, will allow us to make an educated estimate for the optimal selection of hidden layers when training the extended DNN that estimates the PD as a function of both initial position and maturity. Recall that in its simplest, one dimensional, form we consider a jump-diffusion asset value process:

$$dG_t = k(\theta - x)dG_t + \sigma dW_t + \int_{\mathbb{R}} zN(dt, dz), G_0 = x,$$

with the corresponding regime switching and stochastic volatility models given by (3.3) and (3.5), respectively. We consider these models with parameters as given in Examples 4.5.1, 4.5.3 and 4.5.4, respectively. We train the DNN models with 20,000 initial position values and corresponding payoff values. Fig. 5.3 shows the PD functions resulting from DNN models trained with hidden layers ranging from 1 to 5. For all three models, we can easily see that a single hidden layer is insufficient, yielding solutions that do not adequately follow the desired properties that characterize the PD process, such as monotonicity and boundedness. However, adding only one more hidden layer vastly improves the results from all models, and when using $L \geq 3$ hidden layers we obtain nearly indistinguishable solutions.

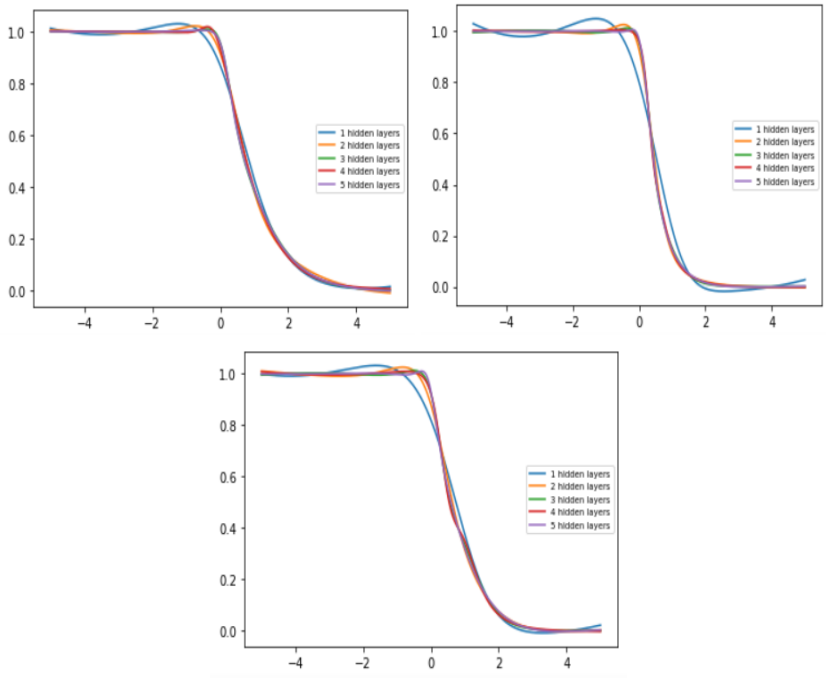

FIGURE 5.3: Graphs of the PD as a function of the initial position for (top left) the DNN model with $\Psi(x,0)$, (top right) the DNN model with $\Psi(x,\rho,0)$ and $\rho = 2$, and (bottom) the DNN model with $\Psi(x,y,0)$ and $y = 0.2$.

5.3.2 Two dimensional input layer (initial position and maturity)

We now extend the DNN models by creating models with both spatial and temporal inputs, as shown in Fig. 5.4. Using the estimations above, we will consider 5 hidden layers and 30,000 training points. Note that each training point now consists of an initial position, maturity time (equivalently, time until maturity u, since we keep $t = 0$; we proceed with this terminology hereinafter) and payoff value, allowing the networks to "learn" how the maturity time affects the PD values. Fig. 5.5 displays the resulting survival, as functions of the initial position and time until maturity, estimated by the DNN models (we display the survival function to ease comparisons with the solution obtained using the Finite Difference scheme). Despite the additional computational power required for the training of DNN model with 2 inputs (due to the increased number training data and model parameters), their advantages over the corresponding DNNs with a single input are obvious; with a single model we obtain the full term structure of the PD function that can be used for the many applications described in the previous Chapters. The extra computational cost is inconsequential relative what would be required for training multiple models for different maturity times.

5.3.3 Extending the input dimension (latent variables)

We have seen that the model can be trained on paths with variable initial position as well as maturity time, thereby resulting in a Neural Network with two inputs. However, in the case of the regime switching and stochastic volatility models it is important to also account for the the latent variables. We can do this by considering

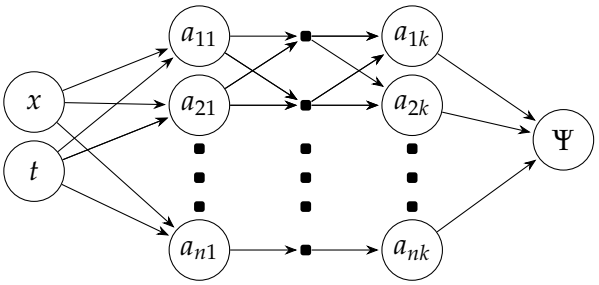

FIGURE 5.4: Fully connected Neural Network with a two dimensional input layer, k hidden layers and a one dimensional output layer.

another model input representing the initial regime or volatility, respectively. Indeed, a Neural Network with such an input is more appropriate for comparisons with the Finite Difference approach, where we obtain solutions across the regimes and volatility grid.

The approach is similar to that described above. We generate paths of the (regime switching or stochastic volatility) model by drawing random initial values of the latent variables, in combination with the initial position and maturity time. Training is then done in the same fashion, using the appropriate payoff function. This extension requires only a simple addition to the input layer of the Neural Network, which now becomes three-dimensional.

It is worth noting that various architectures for the Neural Network may need to be considered before arriving at an appropriate and satisfactory model. In particular, when considering 5 hidden layers, each with 10 neurons, as above, the resulting PD functions do not satisfy standard properties such as monotonicity or stability. It is possible that such occurrences are a case of overfitting, which Neural Networks are often prone to, due to the large number of trainable parameters. For the estimation of the PD functions, we concluded on 3 hidden layers each with 10 neurons for the regime switching model and 3 hidden layers with 7 neurons for the stochastic volatility model. For these higher dimensional models, 50,000 path samples were used for training. Fig. 5.6 displays the resulting PD functions for the given choices of latent variables (which are now an input for the Neural Network model, rather that fixed parameters as in the models previously developed). Hence, the resulting PD functions are now directly comparable to those obtained from the FD scheme, which we will discuss in the sequel.

5.3.4 DNN model for the Generalized Probability of Default

We are now in the position to utilize the DNN framework to address one of the modelling problems presented in the beginning of the Chapter: calculating the PD function under the generalized asset value model. As seen in Chapter 4, one of the most prolific issues when considering more complex models for PD value estimations is the "curse of dimensionality", as faced when considering the Finite Difference method for the PIDE arising from the generalized asset process, given by (3.56). Using the DNN framework we can overcome this issue for such complex models. Indeed, Neural Networks are considered superior in precisely such cases, i.e., in the existence of large, high dimensional data. and will take advantage of this to estimate the PD function under the generalized model $\Psi(x, \rho, y, u)$.

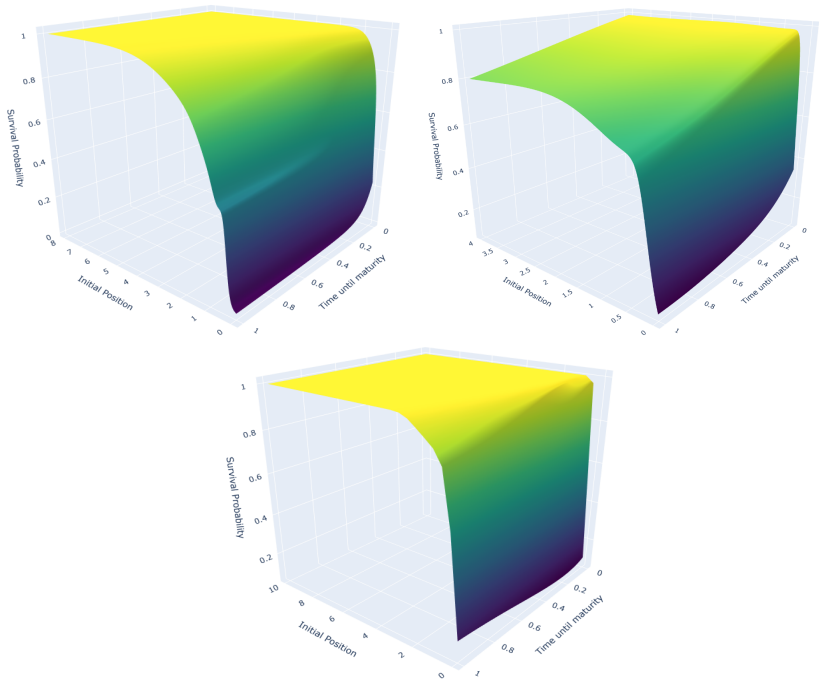

FIGURE 5.5: (Top left) DNN model for $\Phi(x,u)$ (Top right) DNN model for $\Psi(x,\rho,u)$ with $\rho = 2$ (Bottom) DNN model for $\Psi(x,y,u)$ with $y = 2.0$.

The extension is relatively straightforward, as the only structural change that is required is adding another dimension to the input layer, with the resulting architecture shown in Fig. 5.7. Naturally, additional complexity arises due to the simulation process, as both latent variables must now be simulated in order to generate the required training data. A detailed example with the corresponding parameters is given below, as this model has not been explicitly dealt with in the previous numerical methods.

Example 5.3.2. Consider the generalized model (3.7), with parameters of the asset value process (now depending on the underlying regime) given by:

$$(k_\rho, \theta_\rho, \sigma_\rho) = \begin{cases} (0.3, 0.8, 0.3) & \text{if } \rho = 0, \\ (0.2, 0.5, 0.5) & \text{if } \rho = 1, \\ (0.1, 0.6, 0.4) & \text{if } \rho = 2, \end{cases}$$

parameters of the volatility process $(\kappa, \mu, \xi) = (0.05, 0.1, 0.07)$, and jumps are again normally distributed, with size $Z \sim N(0.3, 0.5)$ with rate $\lambda = 1.0$. We train a Neural Network with 3 hidden layers and 7 neurons per layer (further discussion on hyperparameter selection is given in Section 5.5.1). For the training, 60,000 paths of the generalized asset process are simulated. The resulting PD function for a selected pair of initial regime and volatility values is given in Fig. 5.8.

This application displays the utility of using DNNs to estimate the PD functions. The additional variables do not create undue effort when constructing the model, as the only additional effort comes from simulating the corresponding asset value process. On the other hand, a Finite Difference scheme to solve PIDE (3.56) and obtain PD function under the generalized model would be extremely complicated due

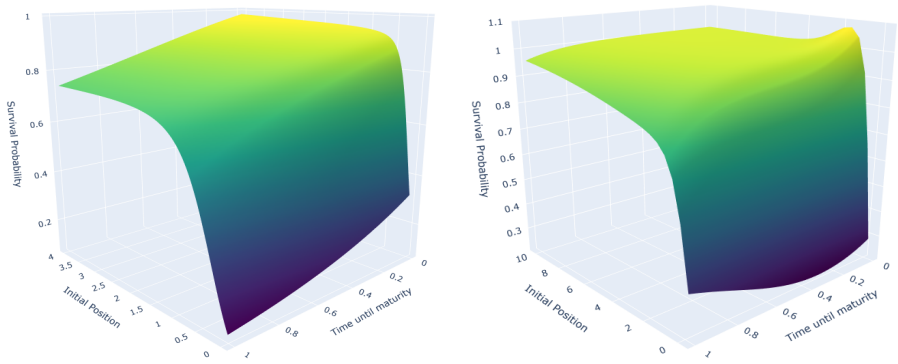

FIGURE 5.6: (Left) DNN model for $\Phi(x, \rho, u)$ with $\rho = 2$. (Right) DNN model for $\Psi(x, y, u)$ with $y = 2.0$.

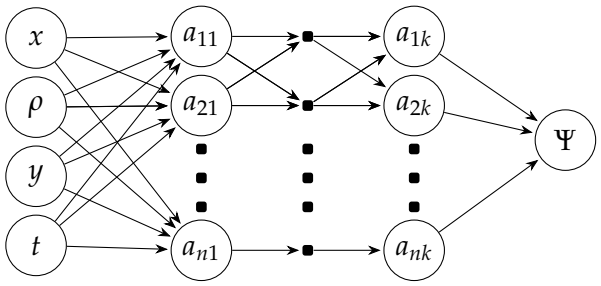

FIGURE 5.7: Fully connected Neural Network with a four-dimensional input layer, k hidden layers and a one-dimensional output layer.

to the number of terms, the structure of the scheme required to ensure monotonicity and stability properties hold (i.e., utilizing the ADI scheme as detailed for the stochastic volatility model), the combination of terminal conditions and, finally, the dimension of the resulting matrices constructed during the iterations of the numerical scheme. In practice, such a scheme would require significantly more computational power than the training of the corresponding DNN.

5.4 DNN models for scenario analysis in credit risk

As shown above, DNNs can be largely beneficial when considering problems in high dimensional spaces. This stems from the remarkable property that such models enjoy regarding their ability to "learn" complicated functions via the architecture and activation functions used. From a theoretical standpoint the this ability is described by the Universal Approximation Theorem (see e.g., Chen and Chen, 1995; Lu et al., 2021 for details). DNN architectures rely on this mathematical framework, and fascinating results surrounding this field have been studied in depth in the context of Hilbert's famous 13th Problem and celebrated results by Kolmogorov and Arnold (Brattka, 2007). In this section we will consider how DNNs can be used for more general credit risk modeling tasks, namely scenario analysis. We will see that this problem can be tackled by considering a DNN model with a high dimensional input corresponding to the asset process parameters. We begin by discussing the use and applicability of scenario analysis in credit risk.

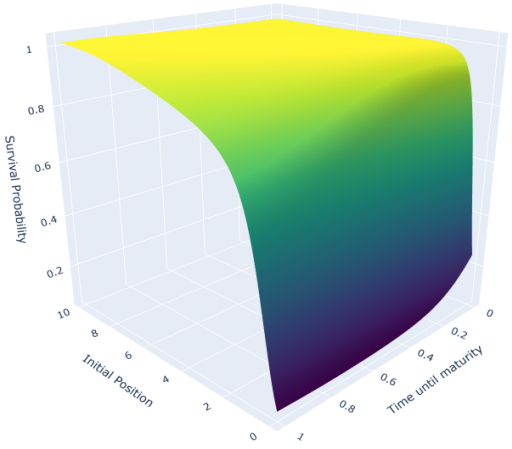

FIGURE 5.8: DNN model for $\Psi(x, \rho, y, u)$ with $\rho = 1$ and $y = 2.0$.

5.4.1 Scenario analysis in credit risk modeling

Understanding the construction of the DNN allows us to also understand its benefits and how these can be maximized by practitioners. Specifically in the field of credit risk, we are often interested in various (optimistic, pessimistic) scenarios which can largely affect individual and portfolio provisions. One way of capturing such cases is to incorporate a regime switching model, as we have already seen. However, it is also important to consider the cases in which credit managers want to account for the effect of non-observable factors that cause changes in the underlying model. In such cases, it is more appropriate to consider a family of stochastic process to which the asset value process belongs. This can be done by considering a model for the asset value process with parameter vector $\mathbf{\Theta} = (\theta_1, \dots, \theta_n)$, which follows an appropriate multidimensional distribution function, $\mathbf{\Theta} \sim F$, i.e., $\theta_i \sim F_i$ for all i and F_i is defined on an appropriate support $\mathcal{S}_i \subset \mathbb{R}$. Naturally, significant differences can then occur in credit risk modeling tasks, such as provisioning. To incorporate such dynamics into a single model we can use a DNN where the input now consists of the parameter vector $\mathbf{\Theta}$.

Considering such a model in the framework of credit risk has important implications. Given that the parameters of the model themselves now vary, in order to reach deterministic results for provisions one would have to be able to accurately estimate the point-in-time value of the parameters $\theta_1, \dots, \theta_n$ or, alternatively, average across the support of their distributions in order to obtain the expected value of the required results. By introducing some simple notation, these cases can easily be represented mathematically. To this end, consider, as above, the asset value process G_t, with $G_0 = x$ and a *risk function* $R : \mathcal{D} \times \mathcal{S} \to \mathbb{R}$, such that $R(x, \mathbf{\Theta})$ represents the result of the credit risk modelling task. Hence, if $\mathbf{\Theta}$ is observable then it is obvious that the risk function is also calculable. On the other hand, if the parameters of the asset value process can not be directly observed (as is the case in most real-life applications) we can use the distribution of the parameter vector to calculate the expected value of the risk function. This gives:

$$R(x) := \mathbb{E}[R(x, \mathbf{\Theta})] = \int_{\mathbb{R}^n} R(x, \mathbf{\Theta}(\mathbf{z}))d\mathbf{F}(\mathbf{z}). \tag{5.15}$$

For example, the above estimation can be used for all risk modelling problems

considered in the previous Chapter under the model uncertainty framework. In particular, provisioning tasks under IFRS 9 now depend directly on the models used for forecasting future losses and therefore, incorporating the more general family of stochastic processes is a way to account for infinitely many scenarios which affect the evolution of individual exposures and, by extension, the credit portfolio. The Deep Learning framework used to estimate PD processes can be used to address these problems; as mentioned, this will require appropriate changes in the input layer of the network (see Fig. 5.9) as well as the subsequent training process.

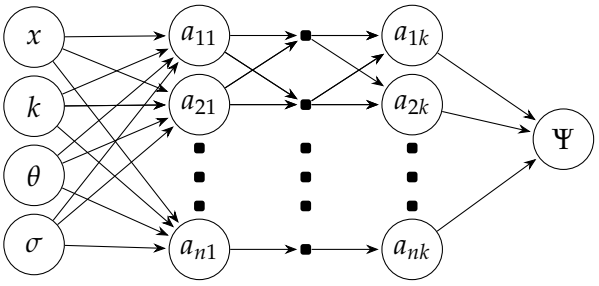

FIGURE 5.9: Fully connected Neural Network with a 4-dimensional input layer and k hidden layers.

5.4.2 DNN model for PDs under a family of stochastic asset processes

In the setting described above, we consider a DNN with 4 inputs: the initial position and the elements of the parameter vector of the asset process (for illustrative purposes we do not consider the temporal input, but this can be easily added, as seen above). Hence, in order to create the training data we simulate from the family of OU processes by randomly generating values of the parameter vector and then simulating the evolution of the process with the given combination of parameters. Naturally, adequate training of the DNN may require a significant increase in the volume of training data, as well as in the time required to train the model; this however is counteracted by the advantage of obtaining a model capable of producing the PD process for an entire family of asset value processes, without the need to re-train the DNN when considering different asset value models. Even though the application of such a model is straightforward given the general DNN architecture, for completeness, we give an example below.

Example 5.4.1. Consider the one dimensional asset value model (5.1) where the stochastic coefficients $\boldsymbol{\Theta} = (k, \theta, \sigma)$ are each randomly sampled from the same distribution $Unif(0.0, 5.0)$, the Lévy jumps are normally distributed with $N(0.0, 0.5)$ and the jump rate is $\lambda = 1.0$. For simplicity, we also consider a fixed maturity, $T = 1.0$. To illustrate the usefulness of this approach, we tested various number of simulations and model architectures, from which we found that satisfactory models can be developed with as low as 30,000 simulations of the the asset value process and a Neural Network with 3 hidden layers and 7 neurons per layer. The resulting model has a four-dimensional input layer (x, k, θ, σ), and, by fixing the parameter set we can obtain the PD as a function of the initial position, as required for the various IFRS 9 modelling tasks previously addressed. Fig. 5.10 displays the resulting PD

functions predicted by the DNN for four parameter sets:

$$(k,\theta,\sigma) = \begin{cases} (0.5,0.5,0.5), \\ (2.0,2.0,2.0), \\ (3.0,3.0,3.0), \\ (4.0,4.0,4.0). \end{cases}$$

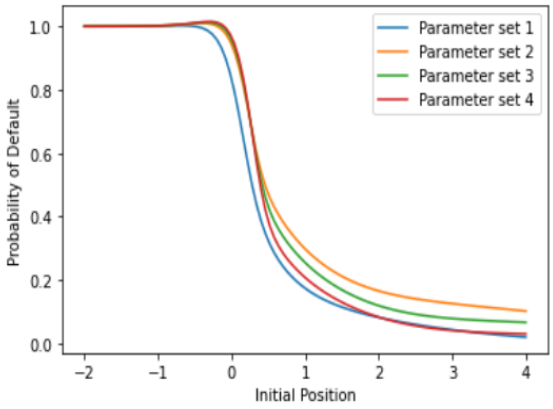

FIGURE 5.10: PD functions for various choices of asset value process parameters.

Naturally, if any of the parameters are observable these can be considered constant and the remaining can be used for the risk function calculation. We believe that this approach can be useful to practitioners as it also provides the necessary tools to calculate the probabilities of various scenarios occurring, by simply using the distribution of the parameter set.

Remark 5.4.2. It is important to remark on the selection of hyperparameters for training the Neural Networks. For the models developed above, many different combinations of layers and neurons per layer were tested, with many choices leading to nonsensical PD functions. Multiple tests showed that, in such cases, choosing different hyperparameters in combination with increasing the number of sample paths for training proved to be most effective. For all experiments a softplus activation function and Glorot uniform kernel initializer are used (changing these did not improve the resulting models). Furthermore, following Frey and Köck, 2022, the models were trained over 10,000 epochs with an Adam optimizer and a piecewise decaying learning rate of 0.1, 0.01, 0.001 and 0.0001 up until 2000, 4000, 6000 and 10,000 epochs, respectively. Finally, Table 5.1 shows the choices of training path simulations, layers and neurons per layer for these DNN models.

Asset value model	Simulated paths	Layers	Neurons per layer
One dimensional	30,000	5	10
Regime switching	50,000	3	10
Stochastic volatility	50,000	3	7
Generalized	60,000	3	7

TABLE 5.1: Neural Network architectures for the three asset value models.

An interesting observation is that, for the more complex asset value processes, the best models were those with less trainable parameters than the model for the PD in the one dimensional case, i.e., less layers and/or neurons per layer. Simultaneously, more asset value paths simulations were used for training as the complexity of the asset value process increases. Therefore, one possible explanation is that the combination of these two changes assists in avoiding overfitting, which Neural Networks are known to be prone to.

5.5 Comparisons with Finite Difference methods

In this section we will quantify and further discuss the main differences between the PD functions obtained from the Neural Network models and the Finite Difference methods examined in Chapter 4. We will first focus on comparisons of the PD values obtained under the three models for which we are able to examine the solutions using both methods. Furthermore, we give details pertaining to the computational time required for both approaches, which is also directly related to the Neural Network's ability to overcome the issues pertaining to the computational cost of increased dimensionality, thereby shedding light on the considerations that practitioners must take into account when deciding on a methodology for their credit risk modelling framework.

5.5.1 Errors in the PD functions

To compare the solutions we will follow Frey and Köck, 2022 and consider the relative error, using the Finite Difference survival probability $\Phi(x,t)$ value as a benchmark (we will work with the survival probability as in Chapter 4, for consistency). Hence, for every (x,t) we measure the quantity:

$$\epsilon(x,t) = \frac{|\Phi^{NN}(x,t) - \Phi^{FD}(x,t)|}{\Phi^{FD}(x,t)}.$$

For the vast majority of credit modelling tasks have discussed we are mainly interested in the error within a domain sufficiently far away from the boundary. It is within this space that possible errors would have the largest effect when using the PD function in practice and it is therefore of great importance to ensure that the two methods deliver PD values as close as possible. Fig. 5.11 displays the relative difference function for the three main models we have considered, from which we obtain maximum errors of approximately 1.6%, 6.0% and 15.8% for the one dimensional, regime switching and stochastic volatility models respectively, when examined within selected domains illustrated in the corresponding graphs. The increase in relative error as we consider more complicated models is expected, given the complexity of the underlying PIDE, which creates the need for significantly more asset value path simulations upon which the Neural Network will be trained.

However, to fully compare the DNN model to the FD scheme, we must also consider points near and on the boundaries. To the best of our knowledge the effects of the initial and boundary conditions have not been carefully examined. In Frey and Köck, 2022, the DNN model is compared to Monte Carlo methods only on the interior of the domain, as previously described. Fig. 5.12 displays the absolute error functions, $|\Phi^{NN}(x,t) - \Phi^{FD}(x,t)|$, in the whole domain, from which we can see that significant differences arise. Specifically, near the boundary and initial positions we now see that very large errors arise, which dissipate the further we move from $x = 0$

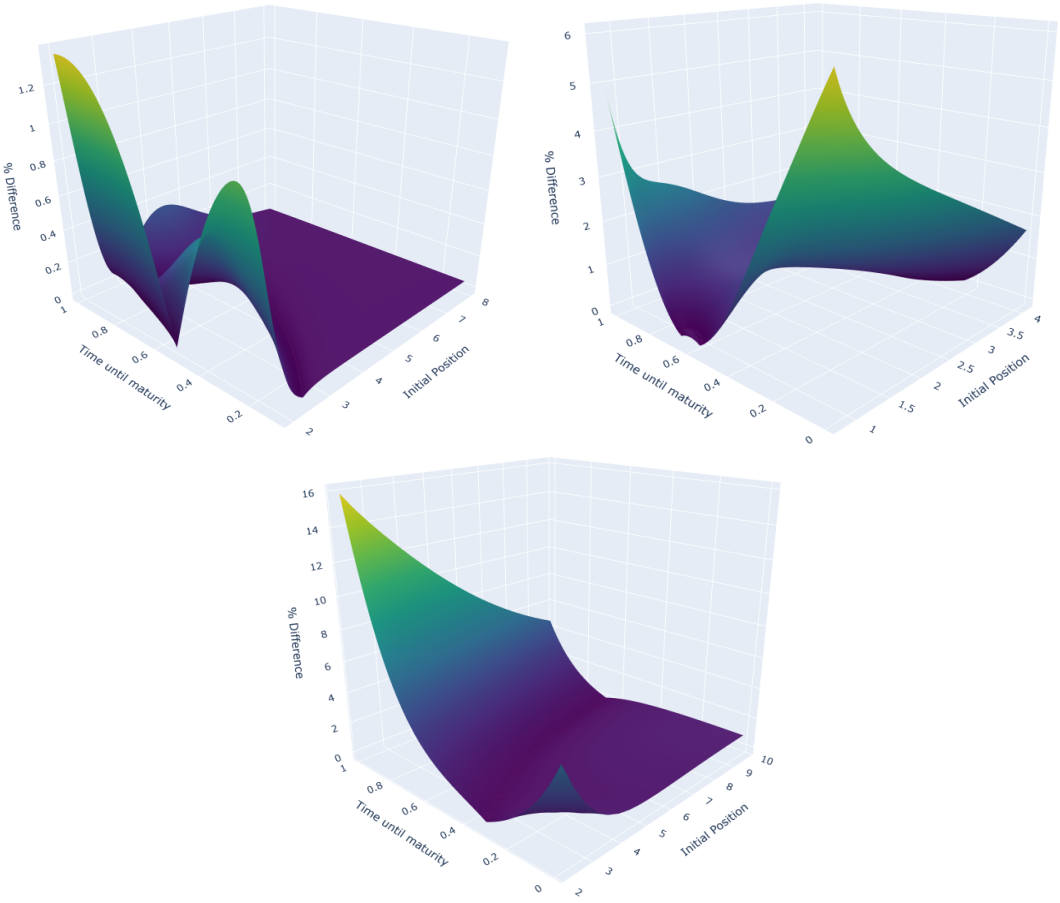

FIGURE 5.11: Relative error of the survival probability from the NN models under the (top left) one-dimensional, (top right) regime switching and (bottom) stochastic volatility asset value model.

or $t = 0$. We posit that this could be explained by the fundamental difference in handling the boundary and initial conditions in the two methods; in the FD schemes recall that we manually impose the boundary and initial conditions by appropriately constructing the coefficient matrices, but on the other hand, no such conditions are directly imposed when training the DNN model. This means that the model must "learn" these solutions using the paths and corresponding payoff values. Hence, it is expected that the differences are largest near this set of points, and the effect of these errors decreases further within the spatial and temporal domain, leading to the insignificant differences observed in Fig. 5.11. Additional tests were conducted in attempt to improve these errors. Specifically, we a) increased the number of total simulated asset value paths used for training, and b) increased the points generated from the initial and boundary conditions, however no significant improvements in the errors were observed.

It is worth emphasizing the importance of these large errors for practitioners. Even though they might not greatly affect the results of many modelling problems we have considered, their existence is a disadvantage of the Neural Network framework for PD estimation, particularly given the fact that these models are still "black-boxes", making it difficult to understand the reason the large errors arise. Furthermore, estimating the magnitude of the errors may also be difficult as it requires having a benchmark, such as the FD solution, which will often suffer from the "curse of

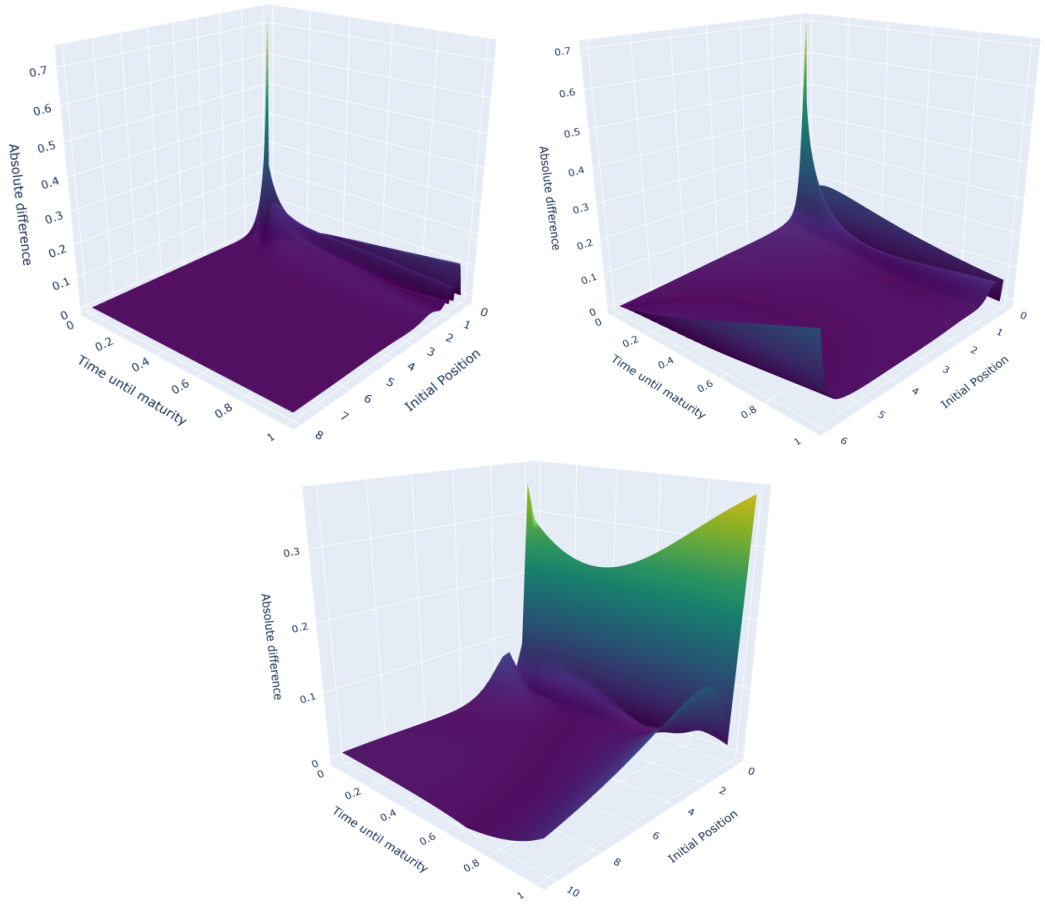

FIGURE 5.12: Absolute error of the survival probability from the NN models under the one-dimensional (top left), regime switching (top right) and stochastic volatility (bottom) asset value model.

dimensionality", thereby inducing a large risk by the DNN for more complex asset value models, since the errors cannot be estimated. Despite this risk, as we have now validated that the solutions in the interior the domains (i.e., sufficiently far from the boundaries) are nearly identical for a large family of stochastic models, the Neural Networks models can be powerful tools to efficiently deal with increasingly complex asset value models and predict the corresponding PD functions.

5.5.2 Applicability of the PD functions and computational requirements

In addition to the numerical, it is important for practitioners to discuss the structural differences between the two approaches examined. In both methods we have been able to obtain the PD estimates under the regime switching and stochastic volatility models as a functions of the spatial, temporal and latent variables (regime or volatility, respectively), i.e., $\Psi(x, \rho, t)$ and $\Psi(x, y, t)$. However, there exists a significant difference in the resulting PD functions: the FD method provides PD estimates covering values of the variables only at the specific grid points created by the discretization (x_i, t_j) for $(i, j) \in \{0, 1, \ldots, N\} \times \{0, 1, \ldots, M\}$, whereas the Neural Network leads to a solution for every $(x, t) \in \mathcal{D} \times [0, T]$. This is an important advantage of the DNN models since it means that we are able to obtain the entire term structure of the PD function without requiring interpolation or similar approximation

techniques, which would be necessary when using the FD scheme. Naturally, this advantage is accompanied with an increase in computational cost, as we will see below. Depending on the application, business needs and setting, practitioners may prefer one of these methods based on this trade-off. For example, if the application requires estimating the PD at any possible values of the initial position and maturity it would be preferable to use the DNN model and take advantage of the ability to predict the PD value for any given (x, t). On the other hand, if we are interested in estimations with a predetermined initial value and maturity input, such as in e.g., derivatives pricing, the FD is useful due to its computational accessibility.

To further illustrate these points, we will compare the time required to estimate the PD function with the FD and DNN methods. Table 5.2 displays these results for each of the three asset value models. For the Finite Difference methods, we use the grids as described in Examples 4.5.1, 4.5.3 and 4.5.4 in Chapter 4. For completeness, we recall that for the one dimensional model a grid of total size $N \times T = 1001 \times 101$ is used, whereas for the regime switching and stochastic volatility models we have grid sizes of $N \times T \times R = 1001 \times 1001 \times 3$ and $N \times T \times V = 201 \times 1001 \times 201$, respectively. On the other hand, as outlined in Table 5.1, 30,000 paths were used to train the DNN under the one-dimensional model and 50,000 under the regime switching and stochastic volatility models. All experiments were conducted in *Python* 3.8.8, run on a Windows 10 Pro with Intel(R) Core(TM) i5-1035G1 CPU Processor; the Finite Difference methods were implemented using *Pandas*, *NumPy*, *SciPy* and the Neural Network were trained using *TensorFlow*.

Model	FD approximation time (seconds)	DNN training time (seconds)
One dimensional	209	66,243
Regime switching	1,141	96,699
Stochastic volatility	154,542	190,734
Generalized	Intractable	194,208

TABLE 5.2: Comparison of computational time required for FD and NN methods.

Note that the DNN training times given in Table 5.2 also account for the time required to simulate the asset value paths. These results provide a quantification of various points previously addressed; firstly, we can see that indeed the FD schemes are easier to implement in terms of computational cost. Particularly, the PD estimations under the one dimensional and regime switching can be obtained very easily. However, the effects of the "curse of dimensionality" are evident, as the time required to obtain the PD estimations under the stochastic volatility model drastically increases. It is easy to see the increase in computational power required to develop the DNN models, compared to the FD solution, for the one dimensional and regime switching asset processes is significant. Interestingly, the same does not apply in the case of the stochastic volatility model, where the computational time for the two methods are very similar, attesting to the Neural Network's ability for efficient generalization. This is further validated when observing that the DNN for the PD under the generalized model requires approximately 194,208 seconds, a rather insignificant difference compared to the stochastic volatility case, whereas the corresponding FD scheme is practically intractable.

5.6 Conclusion

We have seen in this Chapter that the problem of estimating PDs under various stochastic asset value models can be efficiently tackled using Machine Learning approaches and particularly Neural Networks. We can train DNN models to produce the PD functions not only under all the models analyzed in Chapter 4, but also under more sophisticated dynamics, such as the generalized model (3.7). Furthermore, we can harness the DNN's capabilities for generalization and handling high-dimensional data to train models where the parameters of the stochastic processes are handled as inputs, thereby creating a framework that can estimate PD functions under a family of asset value model, as described in section 5.10.

The DNN framework for PD estimation can therefore be extremely useful for practitioners for a wide range of tasks, including provisioning and scenario analysis, which become more demanding under IFRS 9. Their applicability lies not only in the aforementioned capabilities for generalization, but also in the advantages compared to FD difference methods, such as the fact that they can produce PD values for every (x, t), not only at points on a pre-defined grid. Therefore, it is expected that such novel approaches will be not only applicable but necessary as the complexity of credit risk modelling, and the corresponding regulations it must adhere to, increase in the years to come.

Despite their usefulness, many questions relating the development of DNNs remain open. One of the most imminent issues is the selection of model architecture, since, as seen, the best choices may be counter-intuitive, in the sense that more trainable parameters do not always result in better models. It is probable that this observation is closely related to the overfitting problem exhibited by Neural Networks in various settings (see e.g., Caruana, Lawrence, and Giles, 2000; Bilbao and Bilbao, 2017). Note that new approaches have been suggested to rectify this problem; for example, in Srivastava et al., 2014, the authors develop the "Dropout" method for training NNs which is applied to supervised learning problems. Future research could consider such overfitting-prevention techniques in the context of the unsupervised learning problem of solving PDEs and PIDEs. Furthermore, comparisons of the DNN models with the standard FD approaches indicate that relying solely on DNN models for credit risk modelling purposes might still be premature. Results show that large discrepancies between the two approaches can arise near the boundaries of the spatial and temporal domains, which are still difficult to analyze due to the lack of rigorous explainability techniques for complex ML models with a large number of trainable parameters. To this end, given the importance of the tasks entrusted to these models, future research should focus on bridging the numerical, as well as conceptual gaps between the novel ML and established FD methods, in order to safely deploy such models to solve a multitude of credit risk modelling tasks.

Appendix A

Discrete time Markov Chain Models

A.1 Lumpability of Rating Agency transition matrices

Below we display the key tables required for the credit agency ratings presented in Example 2.5.5. These consist of the transition matrices for sovereigns from the three agencies (Moody's, Fitch and S&P, respectively) and the table mapping the internal ratings with the aggregated ratings (for brevity we omit the row ratings, with the understanding that they are in the same order as those in the columns).

M =

Aaa1	Aa1	Aa2	Aa3	A1	A2	A3	Baa1	Baa2	Baa3	Ba1	Ba2	Ba3	B1	B2	B3	Caa1	Caa2	Caa3	Ca	C	WR
0.9282	0.0602	0.0095	0.0020	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.1798	0.5654	0.2088	0.0438	0.0014	0.0007	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0481	0.0748	0.8390	0.0364	0.0011	0.0006	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0299	0.0070	0.1258	0.7582	0.0499	0.0260	0.0014	0.0001	0.0017	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0017	0.0003	0.0076	0.0945	0.7895	0.0368	0.0333	0.0025	0.0326	0.0011	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0005	0.0001	0.0021	0.0273	0.0707	0.8211	0.0258	0.0033	0.0475	0.0015	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0004	0.0001	0.0019	0.0248	0.0278	0.1143	0.8041	0.0218	0.0038	0.0005	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0004
0.0000	0.0000	0.0001	0.0025	0.0035	0.0414	0.1297	0.7527	0.0031	0.0310	0.0022	0.0007	0.0000	0.0000	0.0001	0.0011	0.0001	0.0000	0.0000	0.0000	0.0000	0.0316
0.0000	0.0000	0.0000	0.0009	0.0017	0.0301	0.0348	0.1002	0.7759	0.0515	0.0016	0.0011	0.0000	0.0000	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0021
0.0000	0.0000	0.0000	0.0003	0.0004	0.0037	0.0205	0.0381	0.0945	0.7588	0.0481	0.0316	0.0010	0.0007	0.0011	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000	0.0008
0.0000	0.0000	0.0000	0.0001	0.0001	0.0012	0.0042	0.0326	0.0220	0.1179	0.7552	0.0321	0.0141	0.0034	0.0142	0.0016	0.0006	0.0000	0.0000	0.0000	0.0000	0.0009
0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0007	0.0028	0.0038	0.0495	0.0878	0.7420	0.0287	0.0337	0.0309	0.0155	0.0027	0.0004	0.0004	0.0001	0.0000	0.0009
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0011	0.0008	0.0068	0.0524	0.0971	0.6316	0.1427	0.0532	0.0101	0.0024	0.0002	0.0002	0.0001	0.0000	0.0010
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0001	0.0001	0.0014	0.0048	0.0409	0.0684	0.7246	0.0910	0.0536	0.0092	0.0014	0.0013	0.0004	0.0000	0.0028
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0010	0.0000	0.0000	0.0002	0.0020	0.0051	0.0262	0.1322	0.6266	0.1164	0.0548	0.0042	0.0039	0.0019	0.0000	0.0254
0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0014	0.0000	0.0000	0.0001	0.0015	0.0013	0.0044	0.0452	0.1516	0.5347	0.1542	0.0321	0.0281	0.0111	0.0000	0.0342
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0000	0.0000	0.0000	0.0002	0.0003	0.0021	0.0134	0.1117	0.1038	0.6536	0.0390	0.0343	0.0369	0.0000	0.0045
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0000	0.0000	0.0000	0.0001	0.0001	0.0004	0.0042	0.0232	0.1068	0.1417	0.6124	0.0147	0.0933	0.0000	0.0031
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0018	0.0221	0.0222	0.2634	0.0285	0.4742	0.1869	0.0000	0.0006
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0020	0.0230	0.0304	0.2602	0.1173	0.1018	0.4641	0.0000	0.0008
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
0.0000	0.0000	0.0000	0.0009	0.0010	0.0045	0.0633	0.0021	0.0009	0.0047	0.0612	0.0013	0.0007	0.0018	0.0071	0.0514	0.0066	0.0013	0.0012	0.0003	0.0000	0.7896

F =

AAA	AA+	AA	AA-	A+	A	A-	BBB+	BBB	BBB-	BB+	BB	BB-	B+	B	B-	CCC+	CCC	CCC-	CC	C	SD
0.9924	0.0074	0.0001	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.1490	0.8095	0.0211	0.0199	0.0005	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0083	0.0947	0.8676	0.0287	0.0007	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0003	0.0060	0.1088	0.8416	0.0409	0.0022	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0003	0.0086	0.1322	0.7709	0.0857	0.0000	0.0002	0.0019	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0003	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0005	0.0123	0.1453	0.8002	0.0001	0.0004	0.0047	0.0336	0.0021	0.0004	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0005	0.0098	0.1078	0.8516	0.0276	0.0002	0.0023	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0002	0.0040	0.0487	0.2975	0.6485	0.0001	0.0009	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0000	0.0004	0.0067	0.0572	0.1785	0.7016	0.0325	0.0021	0.0005	0.0027	0.0151	0.0015	0.0011	0.0001	0.0000	0.0081	0.0077	0.0022	0.0005
0.0000	0.0000	0.0000	0.0000	0.0000	0.0006	0.0060	0.0249	0.1911	0.6531	0.0860	0.0171	0.0018	0.0032	0.0016	0.0129	0.0004	0.0003	0.0007	0.0000	0.0000	0.0001
0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0007	0.0030	0.0293	0.1252	0.7594	0.0512	0.0155	0.0120	0.0013	0.0019	0.0001	0.0000	0.0001	0.0000	0.0000	0.0002
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0005	0.0058	0.0378	0.1106	0.7618	0.0720	0.0075	0.0014	0.0016	0.0001	0.0000	0.0001	0.0000	0.0000	0.0009
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0000	0.0006	0.0044	0.0267	0.1218	0.6518	0.1207	0.0280	0.0269	0.0015	0.0005	0.0013	0.0001	0.0000	0.0156
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0005	0.0049	0.0200	0.2127	0.5717	0.1128	0.0635	0.0054	0.0015	0.0033	0.0003	0.0001	0.0030
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0032	0.0041	0.0542	0.1806	0.5568	0.1387	0.0391	0.0081	0.0077	0.0022	0.0005	0.0045
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0004	0.0020	0.0234	0.0013	0.0091	0.0421	0.1394	0.6280	0.0400	0.0272	0.0731	0.0064	0.0008	0.0068
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0035	0.0001	0.0008	0.0054	0.0222	0.2058	0.4376	0.1522	0.0108	0.0370	0.0072	0.1170
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0051	0.0002	0.0013	0.0079	0.0319	0.2801	0.0091	0.6464	0.0156	0.0010	0.0001	0.0008
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0000	0.0000	0.0002	0.0008	0.0143	0.0002	0.0001	0.8194	0.0767	0.0052	0.0828
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0024	0.0001	0.0005	0.0035	0.0153	0.1849	0.0044	0.0026	0.0073	0.1402	0.0113	0.6275
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0050	0.0002	0.0012	0.0076	0.0313	0.2917	0.0090	0.0055	0.0152	0.0010	0.0001	0.6320
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0003	0.0056	0.0002	0.0014	0.0087	0.0349	0.3040	0.0100	0.0062	0.0171	0.0011	0.0002	0.6102

S&P =

AAA	AA+	AA	AA-	A+	A	A-	BBB+	BBB	BBB-	BB+	BB	BB-	B+	B	B-	CCC+	CCC	CCC-	CC	C	SD
0.9789	0.0207	0.0003	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0795	0.8822	0.0279	0.0102	0.0002	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0049	0.1091	0.8179	0.0664	0.0016	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0002	0.0080	0.1220	0.8274	0.0391	0.0027	0.0006	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0003	0.0079	0.1047	0.7566	0.1067	0.0229	0.0005	0.0002	0.0002	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0004	0.0077	0.1140	0.8399	0.0368	0.0008	0.0003	0.0003	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0001	0.0012	0.0197	0.1137	0.8047	0.0344	0.0129	0.0128	0.0004	0.0001	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0002	0.0046	0.0416	0.2638	0.6402	0.0243	0.0242	0.0009	0.0002	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0000	0.0006	0.0062	0.0525	0.1997	0.6709	0.0667	0.0024	0.0005	0.0001	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0000	0.0000	0.0004	0.0043	0.0222	0.1503	0.7464	0.0536	0.0111	0.0019	0.0081	0.0013	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0015	0.0149	0.1443	0.7455	0.0616	0.0146	0.0038	0.0123	0.0010	0.0001	0.0001	0.0000	0.0000	0.0000	0.0002
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0020	0.0236	0.1211	0.7461	0.0712	0.0191	0.0145	0.0016	0.0001	0.0001	0.0000	0.0000	0.0000	0.0004
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0037	0.0327	0.0969	0.6931	0.1386	0.0262	0.0072	0.0004	0.0003	0.0001	0.0001	0.0000	0.0037
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0016	0.0158	0.0229	0.1546	0.6033	0.1295	0.0589	0.0048	0.0030	0.0010	0.0007	0.0000	0.0160
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0016	0.0021	0.0175	0.1410	0.6915	0.1053	0.0095	0.0089	0.0039	0.0025	0.0000	0.0160
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0002	0.0015	0.0185	0.1722	0.5725	0.0942	0.0388	0.0208	0.0151	0.0000	0.0660
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0035	0.0491	0.2426	0.4649	0.0339	0.0175	0.0314	0.0000	0.1568
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0048	0.0667	0.2283	0.0491	0.0756	0.1540	0.0850	0.0000	0.3361
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0044	0.0626	0.1754	0.0494	0.0072	0.2260	0.0846	0.0000	0.3901
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0001	0.0006	0.0091	0.1098	0.2901	0.0833	0.0135	0.0041	0.0480	0.0000	0.4415
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0001	0.0012	0.0153	0.1489	0.3560	0.1055	0.0196	0.0072	0.0075	0.0000	0.3385

The following table displays the mapping between each agency's ratings and the consolidated (broad) ratings, as shown in Hill, Brooks, and Faff, 2009:

Moodys	**Fitch**	**Standard and Poor's**	**Broad Rating**
Aaa	AAA	AAA	8
Aa1	AA+	AA+	7
Aa2	AA	AA	7
Aa3	AA-	AA-	7
A1	A+	A+	6
A2	A	A	6
A3	A-	A-	6
Baa1	BBB+	BBB+	5
Baa2	BBB	BBB	5
Baa3	BBB-	BBB-	5
Ba1	BB+	BB+	4
Ba2	BB	BB	4
Ba3	BB-	B-	4
B1	B+	B+	3
B2	B	B	3
B3	B-	B-	3
Caa1	CCC+	CCC+	2
Caa2	CCC	CCC	2
Caa3	CCC-	CCC-	2
Ca	CC	CC	1
C	C	C	1
WR	DDD/DD/D	DDD/DD/D	Default

Appendix B

Probability of Default modelling under stochastic processes

B.1 Kolmogorov equations for regime switching and stochastic volatility models

Below we recall the Kolmogorov backward equations under the continuous regime switching, stochastic volatility and generalized models, respectively. These depend on the generators of the processes, which now include terms to capture the evolution of the regime and/or volatility processes.

$$\frac{\partial f}{\partial t}(x,\rho,t) = \mathcal{L}_1 Q(x,\rho,t) := \\ k_\rho(\theta_\rho - x)\frac{\partial f}{\partial x}(x,\rho,t) + \frac{1}{2}\sigma_\rho^2\frac{\partial^2 f}{\partial x^2}(x,\rho,t) + \sum_{j\neq\rho} q_{\rho j}\Big(Q(x,j,t) - f(x,\rho,t)\Big) \tag{B.1}$$

$$\frac{\partial f}{\partial t}(x,y,t) = \mathcal{L}_2 f(x,y,t) := \\ k(\theta - x)\frac{\partial f}{\partial x}(x,y,t) + \kappa(\mu - y)\frac{\partial f}{\partial y}(x,y,t) + \frac{1}{2}y\frac{\partial^2 f}{\partial x^2}(x,y,t) + \frac{1}{2}\xi^2 y\frac{\partial^2 Q}{\partial y^2}(x,y,t) \tag{B.2}$$

$$\frac{\partial f}{\partial t}(x,\rho,y,t) = \mathcal{L}_3 f(x,\rho,y,u) := k_\rho(\theta_\rho - x)\frac{\partial f}{\partial x}(x,\rho,y,t) + \kappa(\mu - y)\frac{\partial f}{\partial y}(x,\rho,y,t) \\ + \frac{1}{2}\sigma_\rho^2 y\frac{\partial^2 f}{\partial x^2}(x,\rho,y,t) + \frac{1}{2}\xi^2 y\frac{\partial^2 f}{\partial y^2}(x,\rho,y,t) + \sum_{j\neq\rho} q_{\rho j}\Big(f(x,j,y,t) - f(x,\rho,y,t)\Big), \tag{B.3}$$

where we define separately the generator operators $\mathcal{L}_1, \mathcal{L}_2$ and $\mathcal{L}_3$, for notational convenience. For more details on the generators of regime switching and stochastic volatility models see e.g., Hainaut, 2011; Zhu, Yin, and Baran, 2015.

B.2 PIDEs for the PD functions in Sobolev spaces

For completeness, we first recall some basic definitions pertaining to the theory of weak derivatives in Sobolev spaces.

Definition B.2.1. (**Weak derivative**) Consider an open subset $\Omega \subset \mathbb{R}^n$ and the space of continuous functions which are k times continuously differentiable, for

$k = 1, 2, \ldots$, denoted by $C^k(\Omega)$ and $L^1_{loc}(\Omega)$ the space of locally integrable functions. Furthermore, let $\alpha = (\alpha_1, \ldots, \alpha_n)$ be a multi-index, with order $|\alpha| := \sum_i \alpha_i$, and denote $D^a u$ by:

$$D^\alpha u = \frac{\partial^{|\alpha|} u}{\partial x_1^{\alpha_1} \ldots \partial x_n^{\alpha_n}} = \frac{\partial^{\alpha_1}}{\partial x_1^{\alpha_1}} \cdots \frac{\partial^{\alpha_n}}{\partial x_n^{\alpha_n}} u.$$

Then, for $f \in L^1_{\text{loc}}(\Omega)$ we define $u \in L^1_{\text{loc}}(\Omega)$ to be the αth weak derivative of f, $D^\alpha f = u$, if:

$$\int_\Omega f D^\alpha \varphi dx = (-1)^{|\alpha|} \int_\Omega u \varphi dx,$$

for every smooth test functon with compact support, φ.

With $\mathcal{Q} := \mathcal{D} \times \mathcal{V} \times [0, T]$, we are interested in functions which are twice weakly differentiable with respect to the initial condition and once with respect to the time until maturity. Hence, we can therefore work in the Sobolev space containing all such functions $W^{2,1}(\mathcal{Q}) = \left\{ f \in L^p(\mathcal{Q}) : D^\alpha f \in L^1(\mathcal{Q}), |\alpha| \leqslant 2 \right\}$.

To work in this space, we also need an appropriate weak version of the Itô formula, pertaining to cases where the underlying function may not enjoy the required regularity properties; these results are given by Theorems B.2.2 and B.2.3, due to Krylov, 2008 and Okhrati and Schmock, 2015, respectively. We include the results below, for completeness:

Theorem B.2.2. *Consider the stochastic process*

$$dX_t = a(x,t)dt + \sigma(x,t)dB_t$$

and a region $\mathcal{Q}$, where B_t is a standard Brownian motion, with function f such that function $f \in W^{2,1}(\mathcal{Q})$. Moreover, let τ be some Markov time such that $\tau < \tau_{\mathcal{Q}}$, where $t_{\mathcal{Q}}$ is the exit time of the process from the region $\mathcal{Q}$. Then, if there exists some constant K such that $|\sigma(x,t)| + |a(x,t)| \leq K$, for some fixed time s we have that:

$$\begin{aligned} f(X_\tau, s+\tau) - f(X_t, s+t) = &\int_t^\tau \frac{\partial f}{\partial u}(X_u, s+u)du \\ &+ \int_t^\tau \frac{\partial f}{\partial x}(X_u, s+u)\sigma(X_u, u)dB(t) + \frac{1}{2}\int_t^\tau \frac{\partial^2 f(X_u, s+u)}{\partial x^2}\sigma^2(X_u, u)du, \end{aligned} \tag{B.4}$$

almost surely.

Theorem B.2.3. *Consider the stochastic process with representation*

$$X_t = \gamma t + \int_0^t \int_{\mathbb{R}} z N(dz, du),$$

where $\gamma \in \mathbb{R}$. Assume $f : \mathcal{Q} \to \mathbb{R}$ is a continuous function on $U\mathcal{Q}$ such that $f \in L^1_{loc}(\mathcal{Q})$, i.e., f is locally integrable. Furthermore, assume the existence of locally bounded weak first order derivatives, as defined in B.2.1. Then:

$$\begin{aligned} f(X_t, t) = f(X_0, 0) + \int_0^t \frac{\partial f}{\partial s}(X_u, u)du + \gamma \int_0^t \frac{\partial f}{\partial x}(X_u, u)du + \\ + \int_0^t \int_{\mathbb{R}} f(X_{u-} + z, u) - f(X_{u-}, u) N(du, dz), \end{aligned} \tag{B.5}$$

where all derivatives are understood in the weak sense.

In the Sobolev setting, we can now derive a PIDE using the common approach of appropriate martingale arguments (similar analysis has been given in e.g., Møller, 1995).

Lemma B.2.4. *The survival probability with a variable starting time and fixed maturity T, $\Phi(x,\rho,y,s;T)$ satisfies the partial integro-differential equation, almost surely:*

$$\frac{\partial\Phi}{\partial s}(x,\rho,y,s;T)+\mathcal{L}_3\Phi(x,\rho,y,s;T)+\int_{\mathbb{R}}\Big(\Phi(x+z,\rho,y,s;T)-\Phi(x,\rho,y,s;T)\Big)\nu(dz)=0, \tag{B.6}$$

for $(x,\rho,y,s)\in\mathcal{D}\times\mathcal{R}\times\mathcal{V}\times[0,T]$, with initial and boundary conditions:

$$\begin{aligned}
\Phi(x,\rho,y,T;T)&=\mathbb{1}_{\{x>0\}},\quad (x,\rho,y)\in\mathcal{D}\times\mathcal{R}\times\mathcal{V},\\
\Phi(0,\rho,y,s;T)&=0,\quad (\rho,y,s)\in\mathcal{R}\times\mathcal{V}\times[0,T],\\
\Phi(x,\rho,y,s;T)&\to 1,\ \text{as } x\to\infty,\quad (\rho,y,s)\in\mathcal{R}\times\mathcal{V}\times[0,T],\\
\frac{\partial\Phi}{\partial y}(x,\rho,y,s;T)&=0,\ \text{as } y\to\infty\quad (x,\rho,s)\in\mathcal{D}\times\mathcal{R}\times[0,T],
\end{aligned} \tag{B.7}$$

with the generator operator $\mathcal{L}_3$ as given in (B.3).

Proof. We begin by considering the dynamics of the survival probability. As Φ is differentiable in $W^{2,1}(Q)$, we will employ Theorems B.2.2 and B.2.3 above. We then obtain:

$$\begin{aligned}
\Phi(G_w,R_w,Y_w,w)-\Phi(G_s,\rho,y,s)&=\int_s^w\Big(\frac{\partial\Phi}{\partial r}(G_r,R_r,Y_r,r)+\mathcal{L}_3\Phi(x,\rho,y,s)\Big)dr\\
&+\int_s^w\sigma(R_r)\sqrt{Y_r}\frac{\partial\Phi}{\partial x}(G_r,R_r,Y_r,r)dB_r+\int_s^w\xi\sqrt{Y_r}\frac{\partial\Phi}{\partial x}(G_r,R_r,Y_r,r)dW_r\\
&+\int_s^w\int_{\mathbb{R}}\Big(\Phi(G_r+z,R_r,Y_r,r)-\Phi(G_r,R_r,Y_r,r)\Big)N(dr,dz),
\end{aligned} \tag{B.8}$$

where the derivatives are understood in the weak sense in accordance to definition B.2.1. Also note that we omit the dependence on the t parameter, for brevity. We are now able to formulate the following result regarding the survival probability. We write the dynamics above in terms of the compensated Poisson measure $\tilde{N}(dt,dz)=N(dt,dz)-\nu(dz)dt$. The last term then becomes:

$$\int_s^w\int_{\mathbb{R}}\Big(\Phi(G_r+z,R_r,Y_r,r)-\Phi(G_r,R_r,Y_r,r)\Big)(\tilde{N}(dr,dz)+\nu(dz)dr).$$

Combining with the dynamics of Φ above and using the fact that the sum of the non-martingale quantities must be identically zero, we obtain PIDE (B.6), as required. The boundary conditions follow by definition of the survival probability. □

B.3 Existence and continuity of the PD function

Theorem B.3.1. Arzelà-Ascoli. *Let (X,d) be a compact metric space and $C(X)$ the space of continuous functions on X. Then, if a sequence of continuous functions $\{f\}_{n=1}^{\infty}$ in $C(X)$ is bounded and equicontinuous it has a uniformly convergent subsequence.*

Theorem B.3.2. Schauder Fixed Point *Let $(X,\|\cdot\|)$ be a Banach space and $S\subset X$ is compact, convex, and nonempty. Any continuous operator $A:S\to S$ has at least one fixed point.*

B.4 Regularity of solutions to parabolic PDEs

We will require the following results pertaining to the regularity of solutions of the second order parabolic PDE (1.15). The following are due to Garroni, Menaldi, et al., 1992.

Theorem B.4.1. *Consider a bounded domain Ω, the operator $L := \frac{\partial f}{\partial t}(x,t) - \mathcal{L}f(x,t)$, where $\mathcal{L}$ is the generator operator, and the PDE:*

$$\begin{cases} Lf = g(x,t) & \text{for } (x,t) \in Q_T \\ f(x,0) = \varphi(x) & \text{for } x \in \Omega \\ f(x,t) = \psi(x,t) & \text{for } x \in \Sigma_T := \partial\Omega \times [0,T]. \end{cases} \tag{B.9}$$

Then, for any $g \in C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)$, $\varphi \in C^{2+\alpha}(\bar{\Omega})$, $\psi \in C^{2+\alpha,\frac{2+\alpha}{2}}(\Sigma_T)$, with $0 < a < 1$, (B.9) has a unique solution from the class $C^{2+\alpha,\frac{2+\alpha}{2}}(\bar{Q}_T)$ and satisfies the inequality:

$$\|f\|_{2+\alpha,\bar{Q}_T} \leq C\left(\|g\|_{\alpha,\bar{Q}_T} + \|\varphi\|_{2+\alpha,\bar{\Omega}} + \|\psi\|_{2+\alpha,\Sigma_T}\right),$$

with the constant C independent of f, φ and ψ.

When extending to Lévy models and the corresponding integro-differential equations, we will need the following result.

Lemma B.4.2. *Consider $f \in C^{\alpha+2,\frac{2+\alpha}{2}}(\bar{Q}_T)$ and the differential operator:*

$$If(x,t) = \int_\Omega [f(x+z,t) - f(x,t)]\nu(dz).$$

Then, for $0 < a < 1$, we have that:

$$\|If\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)} \leq \varepsilon\|\nabla f\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)} + C(\varepsilon)\|f\|_{C^{\alpha,\frac{\alpha}{2}}(\bar{Q}_T)}.$$

Note that Lemma B.4.2 is a simplified version of the corresponding results in Garroni, Menaldi, et al., 1992, where the authors consider additional integral operators of higher orders.

Bibliography

Abadi, Martin (2016). "TensorFlow: learning functions at scale". In: *Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming*, pp. 1–1.

Acharya, Viral V., Sanjiv Ranjan Das, and Rangarajan K. Sundaram (2019). "Pricing Credit Derivatives with Rating Transitions". In: *Financial Analysts Journal* 58.3, pp. 28–44.

Ackerer, Damien, Damir Filipović, and Sergio Pulido (2018). "The Jacobi stochastic volatility model". In: *Finance and Stochastics* 22, pp. 667–700.

Applebaum, David (2009). *Lévy processes and stochastic calculus*. Cambridge university press.

Aristidou, Chrystalleni (2018). "The meta-Phillips Curve: Modelling US inflation in the presence of regime change". In: *Journal of Macroeconomics* 57, pp. 367–379.

Asanga, Sujith et al. (2014). "Portfolio optimization under solvency constraints: a dynamical approach". In: *North American Actuarial Journal* 18.3, pp. 394–416.

Ballotta, Laura and Efrem Bonfiglioli (2016). "Multivariate asset models using Lévy processes and applications". In: *The European Journal of Finance* 22.13, pp. 1320–1350.

Ballotta, Laura and Gianluca Fusai (2015). "Counterparty credit risk in a multivariate structural model with jumps". In: *Finance* 36.1, pp. 39–74.

Barles, Guy and Cyril Imbert (2008). "Second-order elliptic integro-differential equations: viscosity solutions' theory revisited". In: *Annales de l'IHP Analyse non linéaire*. Vol. 25. 3, pp. 567–585.

Barndorff-Nielsen, Ole E and Neil Shephard (2001a). "Modelling by Lévy processess for financial econometrics". In: *Lévy processes*, pp. 283–318.

— (2001b). "Non-Gaussian Ornstein–Uhlenbeck-based models and some of their uses in financial economics". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63.2, pp. 167–241.

Barr, DR and MU Thomas (1977). "An eigenvector condition for Markov chain lumpability". In: *Operations Research* 25.6, pp. 1028–1031.

Bates, David S (1996). "Jumps and stochastic volatility: Exchange rate processes implicit in deutsche mark options". In: *The Review of Financial Studies* 9.1, pp. 69–107.

Bauschke, Heinz H and Jonathan M Borwein (1994). "Dykstras alternating projection algorithm for two sets". In: *Journal of Approximation Theory* 79.3, pp. 418–443.

Bauschke, Heinz H et al. (2011). *Fixed-point algorithms for inverse problems in science and engineering*. Vol. 49. Springer Science & Business Media.

Beck, Christian et al. (2021). "Solving the Kolmogorov PDE by means of deep learning". In: *Journal of Scientific Computing* 88, pp. 1–28.

Beerbaum, Dirk (2015). "Significant increase in credit risk according to IFRS 9: Implications for financial institutions". In: *International Journal of Economics & Management Sciences* 4.9, pp. 1–3.

Beerbaum, Dirk and Sammar Ahmad (2015). "Credit risk according to IFRS 9: Significant increase in credit risk and implications for financial institutions". In: *Available at SSRN 2654120*.

Belkina, T and Yu Kabanov (2016). "Viscosity solutions of integro-differential equations for nonruin probabilities". In: *Theory of Probability & Its Applications* 60.4, pp. 671–679.

Bellotti, Tony and Jonathan Crook (2012). "Loss given default models incorporating macroeconomic variables for credit cards". In: *International Journal of Forecasting* 28.1, pp. 171–182.

Ben-Israel, Adi and Thomas NE Greville (2003). *Generalized inverses: theory and applications*. Vol. 15. Springer Science & Business Media.

Benhamou, Eric, Emmanuel Gobet, and Mohammed Miri (2010). "Time dependent Heston model". In: *SIAM Journal on Financial Mathematics* 1.1, pp. 289–325.

Berglund, Jonas (2016). *Estimating expected lifetime of revolving credit facilities in an IFRS 9 framework*. Preprint, Lund University.

Bilbao, Imanol and Javier Bilbao (2017). "Overfitting problem and the over-training in the era of data: Particularly for Artificial Neural Networks". In: *2017 eighth international conference on intelligent computing and information systems (ICICIS)*. IEEE, pp. 173–177.

Blechschmidt, Jan and Oliver G Ernst (2021). "Three ways to solve partial differential equations with neural networks—A review". In: *GAMM-Mitteilungen* 44.2, e202100006.

Brattka, Vasco (2007). *From Hilbert's 13th Problem to the theory of neural networks: constructive aspects of Kolmogorov's Superposition Theorem*. Springer.

Braun, Alexander, Hato Schmeiser, and Florian Schreiber (2015). "Solvency II's Market Risk Standard Formula: How Credible is the Proclaimed Ruin Probability?" In: *Journal of insurance issues*, pp. 1–30.

Brouste, Alexandre et al. (2014). "The yuima project: A computational framework for simulation and inference of stochastic differential equations". In: American statistical association.

Bruche, Max (2005). "Estimating structural bond pricing models via simulated maximum likelihood". In.

Buchholz, Peter (1994). "Exact and ordinary lumpability in finite Markov chains". In: *Journal of applied probability* 31.1, pp. 59–75.

Cai, Shengze et al. (2021). "Physics-informed neural networks for heat transfer problems". In: *Journal of Heat Transfer* 143.6.

Campbell, Stephen L and Carl D Meyer (2009). *Generalized inverses of linear transformations*. SIAM.

Cantor, Richard and Frank Packer (1997). "Differences of opinion and selection bias in the credit rating industry". In: *Journal of Banking & Finance* 21.10, pp. 1395–1417.

Cariboni, Jessica and Wim Schoutens (2007). "Pricing credit default swaps under Lévy models". In: *Journal of Computational Finance* 10.4, p. 71.

Carr, Peter and Anita Mayo (2007). "On the numerical evaluation of option prices in jump diffusion processes". In: *European Journal of Finance* 13.4, pp. 353–372.

Caruana, Rich, Steve Lawrence, and C Giles (2000). "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping". In: *Advances in neural information processing systems* 13.

Chatterjee, Somnath (2015). "Centre for Central Banking Studies". In: *Modelling credit risk. Bank of England*.

Chen, Tianping and Hong Chen (1995). "Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems". In: *IEEE Transactions on Neural Networks* 6.4, pp. 911–917.

Cont, Rama and Ekaterina Voltchkova (2005a). "A finite difference scheme for option pricing in jump diffusion and exponential Lévy models". In: *SIAM Journal on Numerical Analysis* 43.4, pp. 1596–1626.

— (2005b). "Integro-differential equations for option prices in exponential Lévy models". In: *Finance and Stochastics* 9.3, pp. 299–325.

Cox, John C, Jonathan E Ingersoll Jr, and Stephen A Ross (2005). "A theory of the term structure of interest rates". In: *Theory of valuation*. World Scientific, pp. 129–164.

Crandall, Michael G and Pierre-Louis Lions (1983). "Viscosity solutions of Hamilton-Jacobi equations". In: *Transactions of the American mathematical society* 277.1, pp. 1–42.

Cuomo, Salvatore et al. (2022). "Scientific machine learning through physics–informed neural networks: where we are and what's next". In: *Journal of Scientific Computing* 92.3, p. 88.

d'Halluin, Yann, Peter A Forsyth, and Kenneth R Vetzal (2005). "Robust numerical methods for contingent claims under jump diffusion processes". In: *IMA Journal of Numerical Analysis* 25.1, pp. 87–112.

Duan, Jin-Chuan, Ivilina Popova, and Peter Ritchken (2002). "Option pricing under regime switching". In: *Quantitative Finance* 2.2, p. 116.

Eggensperger, Katharina et al. (2013). "Towards an empirical foundation for assessing bayesian optimization of hyperparameters". In: *NIPS workshop on Bayesian Optimization in Theory and Practice*. Vol. 10. 3.

Escalante, René and Marcos Raydan (2011). *Alternating projection methods*. SIAM.

Fadlalla, Adam and Chien-Hua Lin (2001). "An analysis of the applications of neural networks in finance". In: *Interfaces* 31.4, pp. 112–122.

Fourneau, Jean-Michel, M Lecoz, and F Quessette (2004). "Algorithms for an irreducible and lumpable strong stochastic bound". In: *Linear Algebra and its Applications* 386, pp. 167–185.

Frey, Rüdiger and Verena Köck (2022). "Deep neural network algorithms for parabolic PIDEs and applications in insurance mathematics". In: *Mathematical and Statistical Methods for Actuarial Sciences and Finance: MAF 2022*. Springer, pp. 272–277.

Garroni, Maria Giovanna, José Luis Menaldi, et al. (1992). *Green functions for second order parabolic integro-differential problems*. Vol. 275. Chapman and Hall/CRC.

Georgiou, K et al. (2021). "Markov chain lumpability and applications to credit risk modelling in compliance with the International Financial Reporting Standard 9 framework". In: *European Journal of Operational Research* 292.3, pp. 1146–1164.

Gibson, Gavin J and Eric Renshaw (1998). "Estimating parameters in stochastic compartmental models using Markov chain methods". In: *Mathematical Medicine and Biology: A Journal of the IMA* 15.1, pp. 19–40.

Gornjak, Mojca et al. (2017). "Comparison of IAS 39 and IFRS 9: The Analysis of Replacement". In: *International Journal of Management, Knowledge and Learning* 6.1, pp. 115–130.

Goutte, Stéphane (2013). "Pricing and hedging in stochastic volatility regime switching models". In.

Goutte, Stéphane and Armand Ngoupeyou (2011). "Conditional Laplace formula in regime switching model: Application to defaultable bond". In: *Preprint*.

Groetsch, Charles W (1977). *Generalized inverses of linear operators: representation and approximation*. Dekker.

Guo, Tianmei et al. (2017). "Simple convolutional neural network on image classification". In: *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*. IEEE, pp. 721–724.

Gurvits, Leonid and James Ledoux (2005). "Markov property for a function of a Markov chain: A linear algebra approach". In: *Linear algebra and its applications* 404, pp. 85–117.

Hainaut, Donatien (2011). "Financial modeling with switching Lévy processes". In: *ESC*.

Hainaut, Donatien and Olivier Le Courtois (2014). "An intensity model for credit risk with switching Lévy processes". In: *Quantitative Finance* 14.8, pp. 1453–1465.

Hamadène, Said and Marie-Amelie Morlais (2016). "Viscosity solutions for second order integro-differential equations without monotonicity condition: the probabilistic approach". In: *Stochastics* 88.4, pp. 632–649.

Hamilton, James D (2010). "Regime switching models". In: *Macroeconometrics and time series analysis*. Springer, pp. 202–209.

Hayou, Soufiane, Arnaud Doucet, and Judith Rousseau (2018). "On the selection of initialization and activation function for deep neural networks". In: *arXiv preprint arXiv:1805.08266*.

Hecht-Nielsen, Robert (1992). "Theory of the backpropagation neural network". In: *Neural networks for perception*. Elsevier, pp. 65–93.

Hijazi, Samer, Rishi Kumar, Chris Rowen, et al. (2015). "Using convolutional neural networks for image recognition". In: *Cadence Design Systems Inc.: San Jose, CA, USA* 9, p. 1.

Hill, Paula, Robert Brooks, and Robert Faff (2009). "Variations in sovereign credit quality assessments across rating agencies". In: *Journal of Banking & Finance* 34.6, pp. 1327–1343.

Houweling, Patrick and Ton Vorst (2005). "Pricing default swaps: Empirical evidence". In: *Journal of international Money and Finance* 24.8, pp. 1200–1225.

Hull, John, Izzy Nelken, and Alan White (2004). "Merton's model, credit risk, and volatility skews". In: *Journal of Credit Risk Volume* 1.1, p. 05.

Jacobson, Tor, Jesper Lindé, and Kasper Roszbach (2006). "Internal ratings systems, implied credit risk and the consistency of banks' risk classification policies". In: *Journal of Banking & Finance* 30.7, pp. 1899–1926.

Jwo, Dah-Jing (2020). "Investigation of Phase-Locked Loop Statistics via Numerical Implementation of the Fokker–Planck Equation". In: *Applied Sciences* 10.7, p. 2625.

Kemeny, John G and J Laurie Snell (1976). *Markov Chains*. Springer-Verlag, New York.

Kim, Yoonseong and Young Sohn So (2008). "Random effects model for credit rating transitions". In: *European Journal of Operational Research* 182.2, pp. 561–573.

Krylov, Nikolaj Vladimirovič (2008). *Controlled diffusion processes*. Vol. 14. Springer Science & Business Media.

Kumar, Koushal and Gour Sundar Mitra Thakur (2012). "Advanced applications of neural networks and artificial intelligence: A review". In: *International journal of information technology and computer science* 4.6, p. 57.

Kyprianou, Andreas E (2006). *Introductory lectures on fluctuations of Lévy processes with applications*. Springer Science & Business Media.

Li, Yibei et al. (2020). "Credit scoring by incorporating dynamic networked information". In: *European Journal of Operational Research* 286.3, pp. 1103–1112.

Liao, Lizhi et al. (2022). "An empirical study of the impact of hyperparameter tuning and model optimization on the performance properties of deep neural networks". In: *ACM Transactions on Software Engineering and Methodology (TOSEM)* 31.3, pp. 1–40.

Lipton, Alexander and Vadim Kaushansky (2018). "On the first hitting time density of an Ornstein-Uhlenbeck process". In: *arXiv preprint arXiv:1810.02390*.

Loizides, MI and Athanassios N Yannacopoulos (2012). "Lumpable Markov chains in risk management". In: *Optimization Letters* 6.3, pp. 489–501.

Lu, Lu et al. (2021). "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators". In: *Nature machine intelligence* 3.3, pp. 218–229.

Luciano, Elisa and Wim Schoutens (2006). "A multivariate jump-driven financial asset model". In: *Quantitative finance* 6.5, pp. 385–402.

Mao, Zhiping, Ameya D Jagtap, and George Em Karniadakis (2020). "Physics-informed neural networks for high-speed flows". In: *Computer Methods in Applied Mechanics and Engineering* 360, p. 112789.

Medsker, Larry R and LC Jain (2001). "Recurrent neural networks". In: *Design and Applications* 5, pp. 64–67.

Meko, M.C. So and C. Thomas Lyn (2011). "Modelling the profitability of credit cards by Markov decision processes". In: *European Journal of Operational Research* 212.1, pp. 123–130.

Mishura, Yuliya and Olena Ragulina (2016). *Ruin Probabilities: Smoothness, Bounds, Supermartingale Approach*. Elsevier.

Miu, Peter and Bogie Ozdemir (2006). "Basel requirement of downturn LGD: Modeling and estimating PD & LGD correlations". In: *Journal of Credit Risk* 2.2, pp. 43–68.

Møller, Christian Max (1995). "Stochastic differential equations for ruin probabilities". In: *Journal of applied probability* 32.1, pp. 74–89.

Moon, C.-G. and J.G. Stotsky (1993). "Testing the Differences Between the Determinants of Moody's and Standard & Poor's Ratings: An Application of Smooth Simulated Maximum Likelihood Estimation". In: *Journal of Applied Econometrics* 8.1, pp. 51–69.

Nickell, P. and Varotto S. Perraudin W. (2000). "Stability of ratings transitions". In: *Journal of Banking & Finance* 24.1-2, pp. 203–227.

Okhrati, Ramin and Uwe Schmock (2015). "Ito's formula for finite variation Levy processes: The case of non-smooth functions". In: *Journal of Mathematical Analysis and Applications* 430.2, pp. 1163–1174.

Øksendal, Bernt Karsten and Agnes Sulem (2007). *Applied stocstic control of jump diffusions*. Vol. 498. Springer.

Onalan, O (2009). "Financial modelling with Ornstein-Uhlenbeck processes driven by Lévy process". In: *Proceedings of the World Congress on Engineering*. Vol. 2, pp. 1–3.

Pang, Guofei, Lu Lu, and George Em Karniadakis (2019). "fPINNs: Fractional physics-informed neural networks". In: *SIAM Journal on Scientific Computing* 41.4, A2603–A2626.

Papik, Kornel et al. (1998). "Application of neural networks in medicine-a review". In: *Med Sci Monit* 4.3, pp. 538–546.

Pasricha, Puneet et al. (2020a). "Portfolio optimization of credit risky bonds: a semi-Markov process approach". In: *Financial Innovation* 6, pp. 1–14.

Pasricha, Puneet et al. (2020b). "Portfolio optimization of credit risky bons: a semi-Markov process approach". In: *Financial Innovation* 6.25.

Paszke, Adam et al. (2019). "Pytorch: An imperative style, high-performance deep learning library". In: *Advances in neural information processing systems* 32.

Pratiwi, Heny et al. (2020). "Sigmoid activation function in selecting the best model of artificial neural networks". In: *Journal of Physics: Conference Series*. Vol. 1471. 1. IOP Publishing, p. 012010.

Priola, Enrico and Jerzy Zabczyk (2009). "Densities for Ornstein–Uhlenbeck processes with jumps". In: *Bulletin of the London Mathematical Society* 41.1, pp. 41–50.

Raissi, Maziar, Paris Perdikaris, and George E Karniadakis (2019). "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *Journal of Computational physics* 378, pp. 686–707.

Sahli Costabal, Francisco et al. (2020). "Physics-informed neural networks for cardiac activation mapping". In: *Frontiers in Physics* 8, p. 42.

Sak, Halis and Wolfgang Hörmann (2012). "Fast simulations in credit risk". In: *Quantitative Finance* 12.10, pp. 1557–1569.

Schoutens, Wim and Jessica Cariboni (2010). *Lévy processes in credit risk*. Vol. 519. John Wiley & Sons.

Srivastava, Nitish et al. (2014). "Dropout: a simple way to prevent neural networks from overfitting". In: *The journal of machine learning research* 15.1, pp. 1929–1958.

Sutskever, Ilya, James Martens, and Geoffrey E Hinton (2011). "Generating text with recurrent neural networks". In: *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 1017–1024.

Takada, Hideyki and Ushio Sumita (2011). "Credit risk model with contagious default dependencies affected by macro-economic condition". In: *European Journal of Operational Research* 214.2, pp. 365–379.

Thomas, Marlin U. and Donald R. Barr (1977). "An approximate test of Markov chain lumpability". In: *Journal of the American Statistical Association* 72.357, pp. 175–179.

Tian, Jianjun Paul and Dan Kannan (2006). "Lumpability and commutativity of Markov processes". In: *Stochastic analysis and Applications* 24.3, pp. 685–702.

Tobback, Ellen and David Martens (2019). "Retail credit scoring using fine-grained payment data". In: *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 182.4, pp. 1227–1246.

Toivanen, Jari (2010). "A componentwise splitting method for pricing American options under the Bates model". In: *Applied and numerical partial differential equations*. Springer, pp. 213–227.

Vasicek, Oldrich (1977). "An equilibrium characterization of the term structure". In: *Journal of financial economics* 5.2, pp. 177–188.

Virolainen, Kimmo (2004). "Macro stress testing with a macroeconomic credit risk model for Finland". In: *Bank of Finland Research Discussion Paper* 18.

Witzany, Jiri (2011). "A two-factor model for PD and LGD correlation". In: *Available at SSRN 1476305*.

Xu, Xin (2016). "Estimating lifetime expected credit losses under IFRS 9". In: *Available at SSRN 2758513*.

Zhou, Chunsheng (1997). "A jump-diffusion approach to modeling credit risk and valuing defaultable securities". In: *Available at SSRN 39800*.

Zhu, Chao, George Yin, and Nicholas A Baran (2015). "Feynman–Kac formulas for regime-switching jump diffusions and their applications". In: *Stochastics An International Journal of Probability and Stochastic Processes* 87.6, pp. 1000–1032.