International Journal of Financial Engineering
(2026) 2650025 (50 pages)

DOI: 10.1142/S2424786326500258

Calibration of the rating transition model for high- and low- default portfolios

Jian He [*,‡], Asma Khedher [*] and Peter Spreij [*,†]

[*]*Korteweg-de Vries Institute, University of Amsterdam, Amsterdam, The Netherlands*

[†]*IMAPP, Radboud University, Nijmegen, The Netherlands*

Received: 11 July 2025; Accepted: 31 October 2025
Published: 9 May 2026

Abstract

In this paper, we develop maximum likelihood (ML)-based algorithms to calibrate the model parameters in credit rating transition models. Since the credit rating transition models are not Gaussian linear models, the celebrated Kalman filter is not suitable to compute the likelihood of observed migrations. Therefore, we develop a Laplace approximation of the likelihood function and as a result the Kalman filter can be used in the end to compute the likelihood function. This approach is applied to so-called high-default portfolios, in which the number of migrations (defaults) is large enough to obtain high accuracy of the Laplace approximation. By contrast, low-default portfolios have a limited number of observed migrations (defaults). Therefore, in order to calibrate low-default portfolios, we develop an ML algorithm using a particle filter (PF) and Gaussian process regression. Experiments show that both algorithms are efficient and produce accurate approximations of the likelihood function and the ML estimates of the model parameters.

Keywords: Credit risk; transition model; parameter calibration; Kalman filter; particle filter; Laplace method.

1. Introduction

1.1. Problem description and background

Credit transitions, also referred to as credit migrations, constitute one of the most important building blocks of credit risk management. It concerns the change of the

[‡]Corresponding author.
Email addresses: [‡]j.he2@uva.nl

creditworthiness of a firm or a particular debt issue, which leads to potential credit losses and hence results in credit risk. The creditworthiness is usually measured by credit ratings (or scores). For instance, Standard & Poor's Rating Services, Moody's Investor Services, and Fitch Ratings assign corporate bond issuers with credit ratings to reflect their creditworthiness. Once the ratings are defined, the objective is to determine the probability with which the credit risk rating of a borrower decreases or increases by a certain degree, from one level to another, lower or higher, one. The probabilities, that a credit risk rating of a borrower decreases or increases from one period to the next one, are usually collected in a *transition matrix*. Namely, the transition matrix lists the probabilities of the migrations between different credit ratings over specific time intervals. A temporal analysis of credit migrations requires a dynamic model of the transition matrix, the transition model. One of the focuses of the transition model is to address the *cross-sectional dependence* in default rates within a time period due to common economic conditions, i.e., systematic risk factors. This cross-sectional dependence especially describes how likely the obligors default together in a stress scenario and is hence essential in credit risk measuring. The classic examples of modeling the cross-sectional dependence are the Merton model (Merton, 1974) and its extensions such as RiskMetrics-Group (1997) and Crosbie and Bohn (2003) from Moody's. These models belong to the class of so-called structural models, in which the systematic risk factors together with the obligor-specific idiosyncratic factors reflect the obligor's asset value.

A default occurs when the asset value falls below a certain threshold, the default barrier. In these models, the sensitivity of the asset value to the systematic risk factors determines the asset value correlations, and hence also default correlations between different obligors. This asset value correlation is usually referred to as the *asset correlation*. Structural models can also be generalized to describe rating migrations. This requires a definition of rating barriers, and rating changes occur when the asset value crosses a rating barrier value. The dynamics of a systematic risk factor are specified by a model. The model can be a latent variable model, such as the Merton model (Merton, 1974), Albanese *et al.* (2003), or a mixture model which uses both latent and observed (Macroeconomic) factors to form the systematic risk factors, see Credit Suisse First Boston International (1997) and McNeil and Wendin (2007).

The risk factor model parameters (if there are any), the asset correlation, and the rating barriers can be calibrated to time series of historical migration or market data. The calibration serves as the essential link between theoretical model structures and real-world financial data, and hence plays a central role in ensuring the model's practical relevance. Due to the presence of the latent variables, the likelihood of the joint default and migration events depends on an integral, which

in general lacks a closed-form formula. Therefore, in practice, the latent variables at different periods are usually assumed to be independent to simplify the integral in the likelihood function, so that numerical integration and maximum likelihood (ML) methods can be relatively easily implemented, see, for instance, Gordy and Heitfield (2002), Frey and McNeil (2002) and Demey *et al.* (2007). On the other hand, since it is well observed that the default (migration) intensity varies over time according to an economic cycle (see, for instance, McNeil and Wendin (2007)), a good risk factor model is required to capture *serial dependence* caused by the cyclical behavior of economy. Neglecting the fact that credit events tend to cluster in time and are strongly influenced by cyclical economic conditions, such as recessions or periods of financial stress, may lead to biased parameter estimates when calibrated to historical data. However, including serial dependence significantly increases the complexity of the integral in the likelihood function, and consequently makes ML estimation very difficult to perform by using numerical integration.

There are few attempts in the literature to fit such a model with serial dependent latent processes, which belong to the category of *state space models*, to default or migration data. Koopman *et al.* (2005) considered a model where default event is driven by continuous latent factors. But they model the default rates (i.e., the ratio of default obligors) instead of the default counts. This simplification leads to a much less complex likelihood function, but it requires more model assumptions and may show undesirable features, especially when the number of obligors or the number of defaults is small. In order to calibrate the transition model directly to default counts rather than to default rates, McNeil and Wendin (2007) implemented an MCMC approach, more specifically Gibbs sampling, to calibrate the mixture transition model. Although the MCMC approach has its advantages, in general, it is slow. It requires tuning the burn-in period and simulating nested Monte Carlo samples, i.e., first simulating the model parameters and then simulating the latent states given each simulated model parameter. The running time of an MCMC algorithm is acceptable when only calibrating a few transition models. However, when dealing with large portfolios which contain a lot of different rating systems, the MCMC approach is impractical.

1.2. Contributions

The main contributions of this paper are two new calibration algorithms designed for high- and low-default portfolios, respectively. A higher default portfolio, usually a retail portfolio, is a portfolio with a relatively large number of observed defaults, which means that either the probability of default is relatively high or the number of obligors (observations) is large. By contrast, a low-default portfolio,

such as the portfolio of large corporate or financial institutions, means a portfolio with few observed defaults, due to a small default probability and a small amount of obligors in the portfolio.

The two new calibration methodologies aim to calibrate the transition model to the observed default or migration counts. For a high-default portfolio, we propose a calibration approach which approximates the likelihood function by using the Laplace method. We show that, as a consequence, the Kalman filter can be used to compute the approximated likelihood functions. We also show that mode estimation for the Laplace approximation and the log-likelihood function can be calculated by the same Kalman filter iterations. Therefore numerical ML estimation is extremely fast based on the proposed Laplace approach. For the low-default portfolio, due to the small number of observations, the Laplace approximation is less accurate and hence can introduce bias in the parameter estimates. In this case, we propose a Monte Carlo approach, more precisely, a particle filter (PF), to estimate the likelihood function. This proposed Monte Carlo approach leverages on the mode estimation algorithm used in the Laplace approach, and uses its outcome to construct the importance density for the Monte Carlo sampling. The Monte Carlo approach is very efficient and provides very accurate estimation of the likelihood function, with small amounts of Monte Carlo samples. Moreover, since the likelihood function estimated by the Monte Carlo approach is not continuous with respect to the model parameter, we also implement a Gaussian process regression (GPR), a grid-based approach, to smooth the likelihood function so that the ML algorithm can be easily applied. Since the proposed algorithm does not require nested Monte Carlo simulations, it is much faster than a conventional MCMC approach.

The proposed calibration methodologies are generic. They can not only deal with default models, which most literature usually focuses on, but also with migration models. They can also be applied to various structural or mixture transition models with different response functions, such as probit or logistic functions. For instance, they can be applied to calibrate the transition model in He *et al.* (2023). The numerical analysis presented in this paper provides comprehensive validation of the accuracy, robustness, and computational efficiency of the proposed calibration methodologies. Through a series of controlled simulation experiments and real-data applications, the experiments demonstrate that the proposed calibration methodologies achieve highly accurate parameter estimation while offering substantial computational efficiency gains over conventional methods. The application to the S&P credit rating transition data further confirms the empirical effectiveness and robustness of the proposed framework, with estimated asset correlations of about 4.3% and autocorrelation of the latent factors near 20%, which is consistent

with the findings reported in the recent literature (e.g., Batema (2024) and Jacobs (2025)).

Beyond their methodological contribution, the proposed approaches have important practical implications for financial institutions. First, by explicitly modeling *serial dependence*, they better capture the cyclical nature of credit risk, ensuring that stress testing and economic capital frameworks reflect realistic clustering of defaults and migrations during downturns. Neglecting such dependence may lead to underestimation of losses in adverse economic scenarios, and consequently to insufficient capital buffers. Second, the ability to calibrate directly on observed default and migration counts rather than derived rates provides more reliable parameter estimates, especially in portfolios with few obligors or low-default frequencies, where rate-based approaches become unstable. This improvement directly strengthens the implementation of accounting standards such as IFRS 9, where accurate estimation of transition dynamics is critical for expected credit loss calculations. Finally, compared to MCMC-based methods, the proposed algorithms are substantially faster and more resource-efficient, avoiding costly simulation and burn-in procedures. This computational tractability allows financial institutions to run large-scale scenario analyses and regulatory capital calculations on an operational basis, without requiring excessive infrastructure or computation time. These features make the proposed calibration methods practically deployable, thereby closing an important gap between academic modeling and industry application. In particular, they enable financial institutions to integrate theoretically sound models into day-to-day risk management processes, regulatory compliance frameworks, and strategic decision-making. By reducing reliance on *ad-hoc* simplifications and computationally prohibitive techniques, the methods facilitate more accurate stress testing, more consistent IFRS 9 provisioning, and more reliable regulatory capital calculations. As a result, the proposed approaches not only advance the state of the literature but also deliver tangible benefits for practitioners tasked with managing credit risk in complex and dynamic financial environments.

1.3. Organization of the paper

Section 2 provides the background knowledge of the migration models in credit risk modeling and presents the migration models studied in this paper. The proposed ML methodologies are presented in Sec. 3. In Sec. 4, we provide numerical experiments to show the performance of the proposed approaches. Finally, Sec. 5 is devoted to the conclusions. In Appendix A, we summarize some results on the state space model and on Bayesian filters.

1.4. Notation and conventions

Here follows some notations and other conventions used in this paper. All random processes are defined on a fixed probability space $(\Omega, \mathbb{F}, \mathbb{P})$. We will use discrete time, a typical period denoted by k. The discrete time filtration is $\{\mathcal{F}_k, k = 0, 1, \ldots, n\}$, where $\mathcal{F}_k$ summarizes the information up to time k and n is the final time. All processes are assumed to be adapted to this filtration. We denote by $\mathbb{N}^+$ the strictly positive integers. By $\cdot^\top$ we denote the transpose of a vector or a matrix.

We consider a migration matrix with $R \in \mathbb{N}$ different ratings, including default as the last rating. We denote by $T_{ij,k}$ the probability of the transition from a credit state $i = 1, \ldots, R$ at time k to a credit state $j = 1, \ldots, R$ at time $k + 1$. Similarly, the number of (observed) migrations from rating i at period k to rating j at period $k + 1$ is denoted as $m_{ij,k}$. Therefore, the migration matrix at period k, denoted by M_k, is defined as $M_k = (m_{ij,k}, i = 1, \ldots, R, j = 1, \ldots, R) = (m_{1,k}^\top, \ldots, m_{R,k}^\top)^\top$, where $m_{i,k} = m_{i,k} = (m_{i1,k}, \ldots, m_{iR,k})$ denotes row i of the migration matrix M_k. We denote by $M = (M_1, \ldots, M_n)$ the (observed) sequence of migration matrices of n periods. We use similar notations for other matrices. We denote by $x_{1:n}$ a sequence $x_1, \ldots, x_n$.

Finally, we collect all unknown model parameters in a vector ψ.

2. Transition Model

2.1. Transition model review

The probabilities that a credit risk rating of a borrower decreases or increases by a certain degree, from one period to the next one, are described by a transition matrix. Namely, the transition matrix presents the probabilities of the migrations between different credit ratings over specific time intervals. In the transition matrix, the element located at the ith row and jth column represents the probability of a borrower migrating from the ith rating to the jth rating. These are called transition probabilities and are often related to macroeconomic variables such as interest rates, inflation, gross domestic product (GDP), unemployment, etc. Alternatively, transition probabilities can also be modeled by using certain abstract latent processes.

The transition models can, in general, be divided into two main classes, structural and reduced form models. Structural models link the up- or down-grade (or default) probabilities of a firm to the value of its assets and liabilities. The most popular link functions are probits and logistic functions. The structural models were pioneered by Merton (1974), usually referred to as the Merton model, which applies principles of Black and Scholes option pricing in corporate debt valuation.

This approach models equity as a call option on the underlying assets of a firm with strike value equal to the book value of the firm's liabilities. In particular, a default event is triggered if the firm's asset value drops below its liabilities. In this case, the correlations between different firms in a default event are a result of the correlated asset values. Therefore, the so-called asset correlation is usually used to describe the correlation in the default event. The Merton model is further developed by, for instance, Hull and White (2001), Avellaneda and Zhu (2001), Duffie and Singleton (2004) and Albanese *et al.* (2003).

In these structural models, a latent credit process X_t, called a credit index, is used to reflect unobservable credit quality over time which is driven by firm-specific variables such as the asset values. A default occurs if the credit index X_t crosses a default barrier, which is often interpreted as the value of the firms liabilities. The dynamics of the credit index are specified by the model and the default barriers can be calibrated to historical migration or market data. Structural models can be generalized to describe not only defaults, but also rating migrations. This requires a mapping from the latent credit index to the rating states at each time, which corresponds to specifying an interval for the credit index that corresponds to each rating class. The intervals are defined by rating barriers, and rating changes occur when the credit index passes a barrier value.

Reduced form models, by contrast, treat a default as an unexpected event whose probability is governed by a default intensity process. Reduced form models originated with Jarrow (1992), and later studied by, for example, Jarrow and Turnbull (1995), Duffie and Singleton (2004), Frey and McNeil (2002), Wendin and McNeil (2006) and Koopman *et al.* (2008). Unlike the structural model, in which the default relies on a firm's capital structure, the reduced form model describes the default as an event of a jump process driven by exogenous factors. In such models, default correlation is captured by co-movement in default intensities. The reduced form model may be easy to implement and calibrate as no estimation of a firm's asset or liabilities value is required. However, these models suffer from the lack of economic interpretation about default behavior. Whilst, structural models are particularly useful in credit risk management field because they provide intuitive economic interpretations of default events. In fact, this approach, for example, is used in the Vasicek formula, see Vasicek (2002), an example of an asymptotic single factor model, which is the cornerstone of the IRB (internal rating-based approach) formula of a regulatory credit risk weighted assets (RWA) calculation. A structural approach also facilitates the calibration of the model using various data sources, such as default rates, rating transitions, equity returns (as a proxy for asset returns) and CDS spreads. The latter is commonly applied when calibrating sovereign portfolios where the numbers of observations are extremely limited.

The key role of the transition model is to describe the dependence of the migrations (defaults). One important dependence is the cross-sectional dependence, which models the correlations of the credit rating movements among or within different (homogeneous groups of) obligors (e.g., obligors could be grouped by country, industry, rating, etc.). These dependencies are typically caused by exposure to common systemic risk factors. Articles such as Nickell *et al.* (2000) and Bangia *et al.* (2002) demonstrated that besides cross-sectional correlations, default (migration) rates also exhibit serial dependence. Serial correlation is a result of cyclical behavior of economic factors. In particular, it implies that a poor (good) economic state is more likely to be followed by a poor (good) economic year and decreased (increased) asset values, and in the end increase (decrease) the chance of default or downward migrations. Factors used to explain variation in migration rates can be classified into observed (macroeconomic) and unobserved (latent) factors. The use of macroeconomic factors is motivated by the observation that default rates in the financial, corporate, and household sectors increase during recessions. This observation leads to the implementation of credit models that attempt to explain default indicators using economic variables. For example, Simons and Rolwes (2018) used GDP, interest rates, exchange rates and oil price to explain default rates. An advantage of a macroeconomic factor model is that this type of model is very suitable for designing stress scenarios. However, macroeconomic variables are business cycle indicators and may not be an optimal proxy for a credit cycle, see for example Gorton and He (2008) and Koopman and Lucas (2005). Furthermore, macroeconomic variables have to be modeled if used for simulation of migration rates. An alternative approach is to employ unobserved factors. In this case the dynamics of any underlying systematic component are estimated directly from the data. Literature which allows for unobserved factors includes, for instance, Koopman *et al.* (2008), Gagliardini and Gouriéroux (2005) and McNeil and Wendin (2007). While latent factor models are less prone to misspecification of a credit cycle, more advanced mathematical methods, such as the Kalman filter or a PF, are required to estimate them.

2.2. Specified transition model

Structural models are usually used in measuring the credit risk since such models have an economic interpretation, and parts of the models can be calibrated to external data. A mixture model for transitions including both observed and latent systematic factors is introduced in this paper. This type of the model is an example of a *general linear mixture model* (GLMM), see McNeil and Wendin (2007). The precise specification is as follows.

Suppose we have $R \geq 2$ credit states, in which the last state R is default. Recall $T_{ij,k}$, $i,j = 1,\ldots,R$ and $k = 1,\ldots,n$ as defined in Sec. 1.4. We model the transition probability $T_{ij,k}$, $i,j = 1\ldots,R$ at period k as[a]

$$T_{ij,k} = g(\theta_{ij,k}). \tag{1}$$

Here $g : \mathbb{R} \to [0,1]$ is a so-called response function. The $\theta_{ij,k}$, $i,j = 1,\ldots,n$ are often referred to as the signals that describe the dynamics of the transition probabilities at period k. We assume the signals are linearly driven by the *latent* common factors $(x_k)_{k=1,\ldots,n}$ and the *observed* common factors $(u_k)_{k=1,\ldots,n}$ as

$$\theta_{ij,k} = d_{i,j} + K_{i,j}^{\top} x_k + L_{i,j}^{\top} u_k, \quad i,j = 1,\ldots,R\,. \tag{2}$$

Here the constants $d_{i,j}$ indicate the level of the (cumulative) transition probabilities from rating i to rating j, the latent common factors x_k at period k are $s \times 1$ random vectors, the observed common factors u_k at period k are $l \times 1$ vectors which present certain predetermined macro-economic variables or market indices that can be directly observed in the market. The $K_{i,j}$ and $L_{i,j}$, for every $i,j = 1,\ldots,R$ are the factor loadings with dimension $s \times 1$ and $l \times 1$, respectively. They describe the sensitivity of the signal to the common factors. The latent process x is assumed to be an autoregressive AR(1) process (see, e.g., Brockwell and Davis (2006, Chap. 3) for an introduction to such models) which evolves linearly over time with a Gaussian error η. Note that the migrations $\theta_{ij,k}$, $i,j = 1,\ldots,R$ at period k as defined in (2) are correlated.

Recall $m_{i,k}$, $i = 1,\ldots,R$, $k = 1,\ldots,n$ as introduced in Sec. 1.4. It is obvious that the random vector $m_{i,k}$ with dimension $R \times 1$ follows a multinomial distribution when $T_{ij,k}$ is given. Since $T_{ij,k}$ is assumed to be linked with the signal $\theta_{ij,k}$ through the response function g, our transition model in the end is given by, for $k = 1,\ldots,n$, $i,j = 1,\ldots,R$,

$$\begin{aligned} m_{i,k} &\sim mn\big(\cdot|N_{i,k}, g(\theta_{ij,k})\big), \\ \theta_{ij,k} &= d_{ij} + K_{i,j}^{\top} x_k + L_{i,j}^{\top} u_k, \\ x_k &= A x_{k-1} + \eta_k. \end{aligned} \tag{3}$$

In (3), mn stands for multinomial distribution, $N_{i,k} = \sum_{j=1}^{R} m_{ij,k}$ is the number of independent trials (i.e., the number of clients) in period k with rating i. The initial distribution of x_k at a period k is assumed to be Gaussian with mean a_0 and covariance matrix P_0, i.e., $x_0 \sim \mathcal{N}(a_0, P_0)$. The η_k are assumed to be independent with a common $\mathcal{N}(0,Q)$ distribution, A and Q are $s \times s$-matrices. Define $d = (d_{i,j}, i,j = 1,\ldots,R)$, $K = (K_{i,j}, i,j = 1,\ldots,R)$ and $L = (L_{i,j}, i,j = 1,\ldots,R)$.

[a]In some cases, the cumulative transition probability, instead of the transition probability, is modeled through the response function. One of the examples can be found in Sec. 2.4.2.

The parameters d, K, L, A and Q are referred to as the parameters of the transition model and collectively denoted ψ. The random vectors $m_{1,k}, \ldots, m_{R,k}$ are assumed to be independent given the signals $\theta = (\theta_{ij,k}, i,j = 1, \ldots, R)$.

The transition model (3) above belongs to the class of state space models, see Appendix A. We are interested in estimating the (latent) state process $(x_k)_{k=1,\ldots,n}$, but only have access to the process $(M_k)_{k=1,\ldots,n}$ as introduced in Sec. 1.4 which represents the observations. Because of the existence of the white noise in the data, estimating the value of the latent states $(x_k)_{k=1,\ldots,n}$ by the observations $(M_k)_{k=1,\ldots,n}$ is not trivial. There are different methods available in the literature to estimate the latent process, see, e.g., Press (2003), Chui and Chen (2017) and Arulampalam *et al.* (2002). The Bayesian filters and their extensions and the PF are the most used approaches in practice, see Appendices B.1 and B.2 for an introduction and references to Kalman and PFs. In particular, the Bayesian filters are also widely used to compute, or estimate, the likelihood function of the observations. In this paper, the Kalman filter and the PF are used as the building blocks in our proposed MLE-based methodologies to estimate the unknown parameters in the model.

2.3. Likelihood function of the observed migrations

Similar to the notation of the migration matrices, we denote all the signals of the n periods by $\theta = \{\theta_1, \ldots, \theta_n\}$, in which $\theta_k = \{\theta_{ij,k}, i,j = 1, \ldots, R\}$ is the signal at period k. Note that the likelihood of the observed migrations $p(M|\psi)$ with respect to the unknown model parameter ψ is obtained by computing the integral

$$\begin{aligned} p(M|\psi) &= \int p(M|\psi,\theta)p(\theta|\psi)\,\mathrm{d}\theta \\ &= \int p(M|\theta)p(\theta|\psi)\,\mathrm{d}\theta, \end{aligned} \tag{4}$$

where we used the generic symbol p to denote probabilities and densities. Since the distribution of the migrations is assumed to be independent given the signal θ, one obtains

$$\begin{aligned} p(M|\theta) &= \prod_{k=1}^{n} p(M_k|\theta_k) \\ &= \prod_{k=1}^{n}\prod_{i=1}^{R} p(m_{i,k}|\theta_{i,k}). \end{aligned} \tag{5}$$

Plugging Eq. (5) into Eq. (4), one obtains the likelihood

$$p(M|\psi) = \int \prod_{k=1}^{n}\prod_{i=1}^{R} p(m_{i,k}|\theta_{i,k})p(\theta|\psi)\,\mathrm{d}\theta. \tag{6}$$

If the $m_{i,k}$ would be linear in $\theta_{i,k}$, for $i = 1, \ldots, R$, $k = 1, \ldots, n$ and follow a Gaussian distribution, then the integrals in Eq. (6) have an analytic expression and can be calculated using the Kalman filter algorithm. However, the migrations $m_{i,k}$, for $i = 1, \ldots, R$, $k = 1, \ldots, n$, follow a multinomial distribution given the transition probabilities $T_{ij,k}, j = 1, \ldots, R$. Therefore, there is no analytical expression for the integrals in (6). Approximations or a Monte Carlo-based approach are needed to estimate the likelihood function. In Sec. 3, we propose two approaches, a Laplace approximation approach and a Monte Carlo-based approach, to approximate the likelihood function. In the Laplace approximation approach, the Laplace method is applied to Eq. (4) so that a Kalman filter can be applied, whereas the Monte Carlo approach uses the PF with importance sampling. Details about Kalman filter and PF can be found in Appendix B. We finish this section by giving three examples of the structural mixed transition model.

2.4. Examples of transition models

For illustration, we list below three examples for the two transition models with probit and logistic response functions. Especially the transition model with a probit response function will be used in a numerical analysis in Sec. 4. Note that, in these examples, no cure event is modeled.[b] Namely the default rating is absorbing and hence there are no transitions from default to the (nondefault) performing ratings. Therefore, the last row of the transition matrix is trivial and does not need to be modeled.

2.4.1. *One-factor default-only model with probit response function*

The default-only model describes the migrations by only separating default and nondefault events for each rating (group), and omits the migrations between different performing (i.e., nondefault) ratings. Therefore in such a model only default events, and hence the probability of default (PD), of different performing ratings are modeled. This type of model is usually used to calibrate to the defaults on rating level and then extended to be able to simulate the rating migrations when necessary. The default behavior is modeled using a one-dimensional latent driving factor x_k at period k with probit response function, i.e., $g = \Phi$, for Φ being the cumulative distribution function of the standard normal distribution. In order to keep the PD monotone with the ratings, i.e., a worse rating always has a higher PD, all different PD (signals) have the same sensitivity to the common factors. Hence the parameters K and L in (3) reduce to scalars. In this case the one-factor

[b]In practice, the cure event is usually included in the Loss-given-Default (LGD) modeling.

default-only model, for $k = 1, \ldots, n$, and $i = 1, \ldots, R-1$, is given by

$$\begin{aligned} \theta_{i,k} &= d_i + Kx_k + Lu_k, \\ T_{iR,k} &= \mathrm{PD}_{i,k} = \Phi(\theta_{i,k}), \\ \log p(m_{i,k}|\theta_{i,k}) &= m_{iR,k} \log \mathrm{PD}_{i,k} + (N_{i,k} - m_{iR,k}) \log(1 - \mathrm{PD}_{i,k}) + \log \binom{N_{i,k}}{m_{iR,k}}, \end{aligned} \tag{7}$$

where $\binom{N_{i,k}}{m_{iR,k}}$ is the binomial coefficient. When $R = 2$, i.e., the whole portfolio is only separated by default and nondefault states, this model belongs to the framework of the famous Merton model, see Merton (1974).

2.4.2. *Two-factor model with probit response function*

This model is an extension of the one-factor default-only model in Sec. 2.4.1. In this model, the transitions between the nondefault (performing) ratings are also modeled. All the transitions are divided into two parts, default migrations and performing migrations. The default migrations describe the rating transitions from a performing rating to default, while the performing migrations describe the transitions among the performing ratings. Each part, i.e., the default migration or the performing migration, is modeled by a (discrete time) stochastic process. Similar to the one-factor default-only model in Sec. 2.4.1, the sensitivity parameters K and L in Eq. (3) are set to be the same for different default migrations and different performing migrations to keep feasibility (i.e., the transition probabilities should always be larger than zero) and the monotonicity of the PD. Consequently, the parameters K and L become two-by-two diagonal matrices, in which each diagonal element is the sensitivity of the default or performing signal to its corresponding common factor. The response function g is chosen to be a probit function. Note that, in order to make sure that each row of the transition matrix sums up to 1, the cumulative transition probability is actually described through the response function. Therefore, the model is described as, for $k = 1, \ldots, n$ and for $i, j = 1, \ldots, R-1$,

$$\begin{aligned} x_k &= [x_k^{(D)}, x_k^{(P)}]^\top, \\ u_k &= [u_k^{(D)}, u_k^{(P)}]^\top, \\ \theta_k &= [\theta_k^{(D)}, \theta_k^{(P)}]^\top = Kx_k + Lu_k, \quad K = \mathrm{diag}(k_d, k_p), \quad L = \mathrm{diag}(l_d, l_p), \\ T_{iR,k} &= \Phi(d_{i,R} + \theta_k^{(D)}), \end{aligned}$$

$$
\begin{aligned}
T_{ij,k} &= (1 - T_{iR,k})T^{ND}_{ij,k}\,, \\
T^{\mathrm{ND}}_{ij,k} &= ((\Phi(d_{i,j} + \theta^{(P)}_k) - \Phi(d_{i,j+1} + \theta^{(P)}_k)), \quad d_{i,R} = -\infty, \\
\log p(m_{i,k}|\theta_{i,k}) &= \sum_{j=1}^{R} m_{ij,k} \log T_{ij,k} + \log \frac{N_{i,k}!}{\prod_{j=1}^{R} m_{ij,k}!}\,,
\end{aligned}
\tag{8}
$$

where the $T^{\mathrm{ND}}_{ij,k}$ are the migration probabilities given no default at period k. The two-factor model can be extended to multi-factor models, see the following section.

2.4.3. *Multi-factor model with logistic response function*

The standard logistic function is given by $p(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{1+e^x}$. Extending this function to the multi-variate case and applying it as the response function g in model (3), one can build a multi-factor model based on the logistic function as follows. For $k = 1, \ldots, n$, $i = 1, \ldots R-1$, $j = 1, \ldots R$,

$$
\begin{aligned}
T_{ij,k} &= \frac{\exp(\theta_{ij,k})}{\sum_{j=1}^{R} \exp(\theta_{ij,k})}, \\
\log p(m_{i,k}|\theta_{i,k}) &= \sum_{j=1}^{R} m_{ij,k}(\theta_{ij,k}) - N_{i,k} \log \left(\sum_{j=1}^{R} \exp(\theta_{ij,k}) \right) \\
&\quad + \log \frac{N_{i,k}!}{\prod_{j=1}^{R} m_{ij,k}!}.
\end{aligned}
\tag{9}
$$

3. Calibration of the Transition Model

In this section, we present the ML approach to estimate the parameters (to be specified) in the transition model. Note that analytical formulae for the likelihood function of the observed migrations with respect to the model parameters, see Eq. (6), are not available due to the non-Gaussian and nonlinear nature of the transition model. Therefore, approximations are needed. We first propose a Laplace approximation of the likelihood. This approximation is very accurate when modeling high-default portfolios with a relatively large number of observations, such as retail portfolios. The advantage of the Laplace approximation is that it produces a likelihood function that can be analytically computed by using the Kalman filter. Therefore, ML estimation can be easily conducted by using numerical optimization. However, for low-default portfolios with a smaller number

of clients, such as nonretail portfolios, the accuracy of the Laplace approximation is not assured. Consequently, the resulting ML estimator can be severely biased. To circumvent this problem, for a low-default portfolio, we design a PF algorithm to estimate the likelihood function. This algorithm involves an importance sampling approach which makes the PF estimation very efficient. One of the biggest problems of using a PF to approximate the likelihood is that the likelihood function is not continuous with respect to the model parameters. Consequently, usual numerical optimization is not feasible in this case to obtain the ML estimates. As a remedy, we propose a grid-based approach using GPR.

3.1. Approach for high-default portfolios: Laplace approximation for the likelihood function

This subsection describes the Laplace approximation of the likelihood function and the corresponding algorithm to obtain the ML estimates of the unknown model parameters ψ representing d, K, L, A and Q in the transition model (3) given the observed migrations M. Recall Eq. (4), we apply Laplace's method to the integrals, i.e., we use a second order Taylor expansion of $p(M|\theta)$ around the mode $\tilde{\theta}$ of $p(M|\theta)$. The estimation of the mode uses the approach in Durbin and Koopman (2012). We will shortly explain this in Sec. 3.1.1. Here, for the time being, we suppose the mode $\tilde{\theta}$ is known to us. Denote

$$\hat{D}(\theta_0) = \frac{\partial \log p(M|\theta)}{\partial \theta}\bigg|_{\theta=\theta_0}, \quad \hat{H}(\theta_0) = \frac{\partial^2 \log p(M|\theta)}{\partial \theta \partial \theta^T}\bigg|_{\theta=\theta_0}. \tag{10}$$

Note that, using (5),

$$\begin{aligned}\log p(M|\theta) &= \sum_{k=1}^{n} \log p(M_k|\theta_k) \\ &= \sum_{k=1}^{n}\sum_{i=1}^{R-1} \log p(m_{i,k}|\theta_{i,k}).\end{aligned}$$

Therefore, we have

$$\begin{aligned}\hat{D}(\theta) &=: \frac{\partial \log p(M|\theta)}{\partial \theta} = \left[\frac{\partial \log p(m_{1,1}|\theta_{1,1})}{\partial \theta_{1,1}}, \ldots, \frac{\partial \log p(m_{R-1,n}|\theta_{R-1,n})}{\partial \theta_{R-1,n}}\right], \\ \hat{H}(\theta) &=: \frac{\partial^2 \log p(M|\theta)}{\partial \theta \partial \theta^T} = \text{blkdiag}\left(\frac{\partial^2 \log p(m_{1,1}|\theta_{1,1})}{\partial \theta_{1,1} \partial \theta_{1,1}^T}, \ldots, \frac{\partial^2 \log p(m_{R-1,n}|\theta_{R-1,n})}{\partial \theta_{R-1,n} \partial \theta_{R-1,n}^T}\right),\end{aligned} \tag{11}$$

where blkdiag refers to a block diagonal matrix. Then using a second order Taylor expansion around the model $\tilde{\theta}$, one obtains, recalling (4),

$$\begin{aligned} p(M|\psi) &= \int p(M|\theta)p(\theta|\psi)\,\mathrm{d}\theta \\ &= \int e^{\log p(M|\theta)}p(\theta|\psi)\,\mathrm{d}\theta \\ &\approx \int \exp\big(\log p(M|\tilde{\theta}) + \hat{D}(\tilde{\theta})^{\top}(\theta-\tilde{\theta}) + \frac{1}{2}(\theta-\tilde{\theta})^{\top}\hat{H}(\tilde{\theta})(\theta-\tilde{\theta})\big)p(\theta|\psi)\,\mathrm{d}\theta \\ &= p(M|\tilde{\theta})\int \exp\Big\{\frac{1}{2}\big(\theta-\tilde{\theta}+\hat{H}(\tilde{\theta})^{-1}\hat{D}(\tilde{\theta})\big)^{\top}\hat{H}(\tilde{\theta})\big(\theta-\tilde{\theta}+\hat{H}(\tilde{\theta})^{-1}\hat{D}(\tilde{\theta})\big) \\ &\quad -\frac{1}{2}\hat{D}(\tilde{\theta})^{\top}(\hat{H}(\tilde{\theta})^{-1})^{\top}\hat{D}(\tilde{\theta})\Big\}p(\theta|\psi)\,\mathrm{d}\theta. \end{aligned} \tag{12}$$

Let

$$C = (2\pi)^{n(R-1)R/2}(\det(-\hat{H}(\tilde{\theta})^{-1}))^{1/2}p(M|\tilde{\theta})\exp(-\frac{1}{2}\hat{D}(\tilde{\theta})^{\top}(\hat{H}(\tilde{\theta})^{-1})^{\top}\hat{D}(\tilde{\theta}))$$

and

$$\hat{M}(\tilde{\theta}) = \tilde{\theta} - \hat{H}(\tilde{\theta})^{-1}\hat{D}(\tilde{\theta}). \tag{13}$$

With

$$\begin{aligned} p(\hat{M}(\tilde{\theta})|\theta) &= (2\pi)^{-n(R-1)R/2}(\det(-\hat{H}(\tilde{\theta})^{-1}))^{-1/2} \\ &\quad \times \exp\left(\frac{1}{2}(\theta-\hat{M}(\tilde{\theta}))^{\top}\hat{H}(\tilde{\theta})(\theta-\hat{M}(\tilde{\theta}))\right) \end{aligned} \tag{14}$$

and in view of (4) for $\hat{M}(\tilde{\theta})$ instead of θ, it then holds

$$\begin{aligned} p(M|\psi) &\approx C\int p(\hat{M}(\tilde{\theta})|\theta)p(\theta|\psi)\,\mathrm{d}\theta \\ &= C\,p(\hat{M}(\tilde{\theta})|\psi). \end{aligned} \tag{15}$$

Note that (14) expresses that, conditional on θ, $\hat{M}(\tilde{\theta})$ has a $N(\theta, -\hat{H}(\tilde{\theta})^{-1})$ distribution, using that $\hat{H}(\tilde{\theta})$ is negative definite since $\tilde{\theta}$ is the mode. According to Eq. (11), denote $\hat{D}(\tilde{\theta}) = [\hat{D}_{1,1}(\tilde{\theta}_{1,1}), \ldots, \hat{D}_{R-1,n}(\tilde{\theta})]$ and $\hat{H}(\tilde{\theta}) = \text{blkdiag}(\hat{H}_{1,1}(\tilde{\theta}_{1,1}), \ldots, \hat{H}_{R-1,n}(\tilde{\theta}_{R-1,n}))$. One can formulate a linear Gaussian state space model for $\hat{M}(\tilde{\theta}) = \{\hat{M}_1(\tilde{\theta}_1), \ldots, \hat{M}_n(\tilde{\theta}_n)\}$, with the matrix $\hat{M}_k(\tilde{\theta}_k) = (\hat{m}_{ij,k}, i = 1, \ldots, R-1, j = 1, \ldots, R), k = 1, \ldots, n$:

$$\begin{aligned} \hat{m}_{ij,k} &= \theta_{ij,k} + \epsilon_k = d_{i,j} + K_{i,j}^{\top}x_k + L_{i,j}^{\top}u_k + \epsilon_{ij,k}, \\ x_k &= Ax_{k-1} + \eta_k, \quad \eta_k \sim N(0, Q), \end{aligned} \tag{16}$$

with $\epsilon_{i,k} = [\epsilon_{i1,k}, \ldots, \epsilon_{iR,k}] \sim N(0, -\hat{H}_{i,k}(\tilde{\theta}_{i,k})^{-1})$ and the likelihood $p(\hat{M}(\tilde{\theta})|\psi)$ can be computed by Kalman filtering as described in Sec. B.1.

The error introduced by the approximation (15) in the likelihood function originates from the Laplace method, i.e., Eq. (12). It is well known that the Laplace approximation error is asymptotically of order $\mathcal{O}(n^{-1})$; see de Bruijn (1981, Chap. 4), Tierney and Kadane (1986), Miller (2006, Chap. 3) and Łapiński (2019), where n denotes the *sample size*, i.e., the number of observations used to construct the likelihood function. As n becomes large, the likelihood function becomes increasingly concentrated around the mode $\tilde{\theta}$, making the approximation progressively more accurate. This asymptotic behavior justifies the application of the proposed approach to portfolios with a large number of migration or default observations at each time point (i.e., high-default portfolios), where the relatively large sample size ensures that the Laplace approximation provides a reliable estimate of the likelihood function, and consequently leads to an accurate estimate of the parameters.

3.1.1. *Mode estimation*

The Taylor expansion applied to the density $p(M|\theta)$ is around the mode $\tilde{\theta}$ of the posterior density $p(\theta|M)$, i.e.,

$$\tilde{\theta} = \arg\max_{\theta} p(\theta|M).$$

The posterior density $p(\theta|M)$ does not have an explicit expression from which we can obtain the mode analytically. Therefore, a Newton–Raphson method is applied to numerically maximize $p(\theta|M)$ with respect to θ, see Durbin and Koopman (2012). Suppose the Newton–Raphson estimate at the current step is $\hat{\theta}$, the Newton–Raphson update of the current estimate $\hat{\theta}$, denoted by $\hat{\theta}^{+}$, is expressed by

$$\hat{\theta}^{+} = \hat{\theta} - [\dot{p}(\theta|M)|_{\theta=\hat{\theta}}]^{-1}\dot{p}(\theta|M)|_{\theta=\hat{\theta}}, \tag{17}$$

where the dots denote differentiation with respect to θ. Note that (in self-evident notation for densities)

$$\log p(\theta|M) = \log p(M|\theta) + \log p(\theta) - \log p(M). \tag{18}$$

The explicit expression for the gradients $\dot{p}(\theta|M)$ and Hessian $\dot{p}(\theta|M)$ can be derived according to Eq. (18), since the analytic formulas for the conditional likelihood $p(M|\theta)$ and the marginal density $p(\theta)$ are available. Specially, the gradient and Hessian of $p(M|\theta)$ are computed as $\hat{D}(\hat{\theta})$ and $\hat{H}(\hat{\theta})$ in Eq. (11), respectively. Moreover, Durbin and Koopman (2012) showed that the Newton–Raphson update $\hat{\theta}^{+}$ in Eq. (17) is equal to the posterior estimates of $\theta_1, \ldots, \theta_n$ in the linear Gaussian state space model (16). These posterior estimates can be computed through the Kalman filter algorithm in Sec. B.1.

3.1.2. *Algorithms for mode estimation and MLE based on Laplace approximation*

In this section, we present the algorithm for the mode estimation and MLE based on the Laplace approximation.

3.2. Approach for low-default portfolios: particle filter for the likelihood function

The proposed Laplace approach applied to the likelihood requires a sufficiently large number of observed migrations to ensure a satisfactory degree of accuracy. However, this property is not always met. For instance, when modeling the default events of large corporate and financial institutions, the observed numbers of defaults are very limited since these companies rarely default. In this case, we propose a PF-based algorithm, see, e.g., Arulampalam *et al.* (2002), Cappé *et al.* (2007), Doucet and Johansen (2011) or He *et al.* (2021), to estimate the likelihood function. We design an importance sampling approach that leverages the Laplace approximation, so that the PF becomes much more efficient and can deal with the outliers in the data.

3.2.1. *Importance sampling*

In Appendix B.2, the basic (bootstrap) PF is presented. This filter will be used to estimate the likelihood of the observations $p(M|\psi)$. However, the PF is very inefficient when outliers in the data are present. Experiments, see for instance Sec. 4.2, show that the likelihood function obtained from the PF is very sensitive to outliers. Therefore, a large number of Monte Carlo samples need to be generated to obtain an accurate estimate of the posterior distribution of the outlier and consequently to obtain also an accurate estimate of the likelihood. In this case, importance sampling is used to enhance the performance of the PF. The idea of importance sampling is to sample the Monte Carlo particles from a proposed density that is as close as possible to the posterior density $p(x_k|M_{1:k})$, where $(x_k)_{k=1,\dots,n}$ is the latent process and $(M_k)_{k1,\dots,n}$ is the migration matrix sequence. This proposed density is the so-called importance density. Denote the importance density by $q(x_k|M_{1:k}, x_{k-1})$. In the present context, Eq. (B.5) takes the form

$$\begin{aligned} p(x_k|M_{1:k},\psi) &= \frac{p(M_k|x_k,M_{1:k-1},\psi)\int p(x_k|x_{k-1},\psi)p(x_{k-1}|M_{1:k-1},\psi)\,\mathrm{d}x_{k-1}}{p(M_k|M_{1:k-1},\psi)} \\ &= \frac{\int \alpha(x_k,x_{k-1},M_{1:k}|\psi)q(x_k|M_{1:k},x_{k-1})p(x_{k-1}|M_{1:k-1},\psi)\,\mathrm{d}x_{k-1}}{p(M_k|M_{1:k-1},\psi)}, \end{aligned} \tag{19}$$

where

$$\alpha(x_k, x_{k-1}, M_{1:k}|\psi) = \frac{p(M_k|x_k, M_{1:k-1}, \psi)p(x_k|x_{k-1}, \psi)}{q(x_k|M_{1:k}, x_{k-1})} \tag{20}$$

is the so-called importance weight. Consequently, the conditional likelihood in Eq. (B.6) is reformulated as

$$p(M_k|M_{1:k-1}, \psi) = \int\int \alpha(x_k, x_{k-1}, M_{1:k}|\psi)q(x_k|M_{1:k}, x_{k-1})p(x_{k-1}|M_{1:k-1}, \psi)\,\mathrm{d}x_{k-1}\mathrm{d}x_k. \tag{21}$$

When implementing importance sampling in the PF, according to Eqs. (19) and (21), the particles of x_k are generated from the importance density $q(x_k|M_{1:k}, x_{k-1})$, instead of the transition density $p(x_k|x_{k-1})$ as described in Appendix B.2. As a result, according to Eq. (B.11), the formula for the weights in Eq. (B.12) now becomes

$$w_k^{(i)} = \frac{w_{k-1}^{(i)}\alpha(x_k^{(i)}, x_{k-1}^{(i)}, M_{1:k}|\psi)}{\sum_{i=1}^N w_{k-1}^{(i)}\alpha(x_k^{(i)}, x_{k-1}^{(i)}, M_{1:k}|\psi)}. \tag{22}$$

The biggest difficulty when implementing importance sampling is to find a proper importance density. To this end, we propose an importance density by leveraging on the Laplace approximation described in Sec. 3.1. Once the approximation $\tilde{\theta}$ of the mode is obtained, the Laplace approximation is applied to the density $p(M|\theta)$. The approximated density $p(\hat{M}(\tilde{\theta})|\theta)$ results in a linear Gaussian state space model as in Eq. (16). Therefore, by running the Kalman filter algorithm on the model (16) one obtains an estimate of the posterior distribution of x_k.

To summarize, we present here an alternative algorithm to Algorithm 2. This alternative algorithm is specially designed for the low-default portfolio, in which the number of observations is not large enough to support a good approximation using Laplace approach in Algorithm 2. Note that we will need the mode estimation procedure as in Algorithm 1. Recall the notation $M_{1:k} = M_1, \ldots, M_k$. When $k = 0$, then this is an empty sequence.

3.2.2. *MLE for the particle filter*

The results in Sec. 3.2.1 show that for a given parameter ψ, the (conditional) likelihood $p(M_k|M_{1:k-1}, \psi)$ can be estimated according to Eq. (??). However, the PF estimate of the conditional likelihood is not a continuous function of ψ. This is due to the resampling steps. At each time k, in multinomial resampling, a piecewise constant and hence discontinuous cumulative distribution function is defined

Algorithm 3.1 (Mode estimation (Newton-Raphson)). Given are the parameters d, K, L, and the process u_i, $i = 1, \ldots, n$, see (2).

Initialization: Define the initial guess $\hat{\theta}^+$ for the mode of $p(\theta|M)$. Define the threshold C_T as the condition for judging the convergence of Newton–Raphson algorithm. Define the initial $\hat{\theta}$ such that $\|\hat{\theta}^+ - \hat{\theta}\| > C_T$

Iteration: While $\|\hat{\theta}^+ - \hat{\theta}\| > C_T$,

1. Reset $\hat{\theta} = \hat{\theta}^+$.
2. Compute the gradient $\hat{D}(\hat{\theta})$ and Hessian $\hat{H}(\hat{\theta})$ in Eq. (11).
3. Run the Kalman filter algorithm (Algorithm B.1) on model (16). The output of the Kalman filter is $\hat{x}_k$, $k = 1, \ldots, n$.
4. Update $\hat{\theta}^+$ to $(d + K\hat{x}_1 + Lu_1, \ldots, d + K\hat{x}_n + Lu_n)$.

Convergence: Once $\|\hat{\theta}^+ - \hat{\theta}\| \leq C_T$, the mode estimation is given by $\hat{\theta}^+$.

Algorithm 3.2 (MLE (Laplace approximation)).

Likelihood function: Use the parameter value ψ as input, build the likelihood function $p(M|\psi)$ by:

1. Run the mode estimation Algorithm 3.1 and obtain the mode estimate $\tilde{\theta}_\psi$.
2. Given the mode estimate $\tilde{\theta}_\psi$, compute the gradient $\hat{D}(\tilde{\theta}_\psi)$ and Hessian $\hat{H}(\tilde{\theta}_\psi)$.
3. Compute the likelihood $p(\hat{M}(\tilde{\theta}_\psi)|\psi)$ in Eq. (15) by applying the Kalman filter to model (16).
4. Approximate the likelihood function by the Laplace approximation of (15), i.e., $p(M|\psi) \approx Cp(\hat{M}(\tilde{\theta}_\psi)|\psi)$.

Maximization: Maximize the likelihood function $p(\hat{M}(\tilde{\theta}_\psi)|\psi)$ and obtain the optimal parameter $\tilde{\psi}$, i.e.,

$$\tilde{\psi} = \underset{\psi}{\operatorname{argmax}}\, Cp(\hat{M}(\tilde{\theta}_\psi)|\psi)\,.$$

Latent states estimation: Given $\tilde{\psi}$, run the mode estimation algorithm (Algorithm 3.1) and obtain the estimates $\hat{x}_k$ of the latent process x_k, $k = 1, \ldots, n$.

by the weights $\{w_k^{(i)}, i = 1, \ldots, N\}$ and particles $\{x_k^{(i)}, i = 1, \ldots, N\}$. A small change in the parameter ψ will cause a small change in the importance weights $\{w_k^{(i)}\}$. But, due to the discontinuous character of the multinomial cumulative distribution function, a small change in the importance weights will potentially generate a different set of resampled particles $\{x_k^{(i)}\}$ and hence the likelihood function estimate will not be continuous in ψ. This discontinuity problem in the static parameter estimation using a PF has generated a lot of interest over the past few years and many techniques have been proposed to solve it, for instance gradient approach of Poyiadjis *et al.* (2005), expectation maximization (EM) in Andrieu *et al.* (2004) and Wills *et al.* (2008), smoothing likelihood in Pitt (2002), and MCMC in Chopin *et al.* (2013). However, these approaches have difficulties in

Algorithm 3.2 (**MLE (Laplace approximation)**).

Likelihood function: Use the parameter value ψ as input, build the likelihood function $p(M|\psi)$ by:

1. Run the mode estimation Algorithm 3.1 and obtain the mode estimate $\tilde{\theta}_\psi$.
2. Given the mode estimate $\tilde{\theta}_\psi$, compute the gradient $\hat{D}(\tilde{\theta}_\psi)$ and Hessian $\hat{H}(\tilde{\theta}_\psi)$.
3. Compute the likelihood $p(\hat{M}(\tilde{\theta}_\psi)|\psi)$ in Eq. (15) by applying the Kalman filter to model (16).
4. Approximate the likelihood function by the Laplace approximation of (15), i.e., $p(M|\psi) \approx Cp(\hat{M}(\tilde{\theta}_\psi)|\psi)$.

Maximization: Maximize the likelihood function $p(\hat{M}(\tilde{\theta}_\psi)|\psi)$ and obtain the optimal parameter $\tilde{\psi}$, i.e.,

$$\tilde{\psi} = \underset{\psi}{\operatorname{argmax}}\, Cp(\hat{M}(\tilde{\theta}_\psi)|\psi)\,.$$

Latent states estimation: Given $\tilde{\psi}$, run the mode estimation algorithm (Algorithm 3.1) and obtain the estimates $\hat{x}_k$ of the latent process x_k, $k = 1, \ldots, n$.

terms of implementation, see, for instance, Kantas *et al.* (2009). For instance, in the gradient approach the step size is difficult to tune; the EM method requires a large number of particles to ensure the likelihood in the expectation step increases monotonically; and the MCMC approach is usually time-consuming. Therefore, in this paper, we propose a smoothing likelihood method based on GPR to calibrate the model parameters, and the experiments in Sec. 4.4 show that the GPR-based MLE approach is robust and fast.

3.2.3. *Smooth likelihood function by GPR*

The idea of the smoothing likelihood approach is to approximate the likelihood function by a smooth function, see Pitt (2002). In this paper, we choose to use GPR to approximate the likelihood function. A short introduction to GPR is presented in Appendix C. Before applying the GPR methodology to the ML estimation, it needs to be trained to evaluate the likelihood. Given a set of pre-selected model parameters, the likelihoods are approximated by using the proposed PF algorithm (Algorithm 3). The pre-selected model parameters and the corresponding likelihoods are used to construct the grid for the model parameters. Then the GPR is applied to fit the grid. Once the GPR is trained, the ML algorithm can be applied and one obtains an approximation of the ML estimator. The algorithm is summarized as follows.

Algorithm 3.3 (Particle filter with importance sampling for given parameter ψ). Given is the parameter vector ψ.

Initialization: Assume an initial distribution $p(x_0)$ for the latent states x_0, and sample from this initial distribution to get the N particles $\{x_0^{(i)}, i = 1, \ldots, N\}$ and define $\{w_0^{(i)} = \frac{1}{N}, i = 1, \ldots, N\}$.

Importance density: Run Algorithm 3.1 and obtain the mode estimation of θ. Given the mode $\tilde{\theta}$, run the Kalman filter algorithm (Algorithm B.1) on model (16) to obtain the posterior distribution of the latent process $\{x_k, k = 1, \ldots, n\}$. These posterior distributions, denoted by $q(x_k|M_{1:k}, x_{k-1}^{(i)}, \psi), k = 1, \ldots, n$, are used as the importance densities.

Particle filter recursion: For $k = 1, \ldots, n$, suppose the estimated posterior distribution at time $k-1$ is given by $p(x_{k-1}|M_{1:k-1}, \psi) \approx \frac{1}{N}\sum_{i=1}^{N} \delta_{x_{k-1}^{(i)}}(x_{k-1})$.

1. Sample N particles from the importance density, i.e., $\tilde{x}_k^{(i)} \sim q(x_k|M_{1:k}, x_{k-1}^{(i)}, \psi)$, for $i = 1, \ldots, N$.
2. Compute the weights according to Eqs. (20) and (22), namely

$$w_k^{(i)} = \frac{w_{k-1}^{(i)} \alpha(\tilde{x}_k^{(i)}, x_{k-1}^{(i)}, M_{1:k} \mid \psi)}{\sum_{i=1}^{N} w_{k-1}^{(i)} \alpha(\tilde{x}_k^{(i)}, x_{k-1}^{(i)}, M_{1:k} \mid \psi)},$$

   with the initial w_0 as specified in the Initialization step, and the other w_k are calculated from the iteration, using the equation above with

$$\alpha(\tilde{x}_k^{(i)}, x_{k-1}^{(i)}, M_{1:k} \mid \psi) = \frac{p(M_k \mid \tilde{x}_k^{(i)}, M_{1:k-1}, \psi) p(\tilde{x}_k^{(i)} \mid x_{k-1}^{(i)}, \psi)}{q(\tilde{x}_k^{(i)}|M_{1:k}, x_{k-1}^{(i)}, \psi)}),$$

3. Resample $\{x_k^{(1)}, \ldots, x_k^{(N)}\}$ from $\{\tilde{x}_k^{(1)}, \ldots, \tilde{x}_k^{(N)}\}$ with probabilities $\{w_k^{(1)}, \ldots, w_k^{(N)}\}$.
4. The posterior distribution and the conditional likelihood at time k are estimated as

$$\begin{aligned} p(x_k|M_{1:k}, \psi) &\approx \frac{1}{N}\sum_{i=1}^{N} \delta_{\tilde{x}_k^{(i)}}(x_k), \\ p(M_k|M_{1:k-1}, \psi) &\approx \frac{1}{N}\sum_{i=1}^{N} p(M_k|x_k^{(i)}, \psi). \end{aligned} \tag{23}$$

Likelihood: Compute the log-likelihood of the observation as

$$\log p(M|\psi) \approx \sum_{k=1}^{n} \log p(M_k|M_{1:k-1}, \psi)\,.$$

4. Numerical Results

This section presents a comprehensive evaluation of the proposed calibration methodologies through a series of simulations and empirical studies. In Sec. 4.1, we first outline the experimental setup for the numerical experiments, including the one-factor default-only model and the two-factor transition model, which serve as representative examples for assessing the accuracy and efficiency of the

proposed methods. Section 4.2 examines the approximation quality of the likelihood function obtained from the Laplace method and the PF with importance sampling. Section 4.3 applies the Laplace approximation to calibrate the two-factor transition model, while Sec. 4.4 focuses on the PF calibration with GPR, providing a detailed comparison of its performance against existing MCMC and default-rate-based approaches in terms of estimation accuracy and computational cost. Finally, Sec. 4.5 demonstrates the practical applicability of the proposed method using the empirical S&P credit rating transition data.

4.1. Setup for the transition model

We use the one-factor default-only model and the two-factor transition model with the probit response function, see Eqs. (3), (7) and (8), as the examples to illustrate the performance of the proposed calibration methodologies. Specially, the default-only model is used to show the likelihood function approximation of the proposed Laplace and the PF (with importance sampling) approximation, see Sec. 4.2. It is also used in Sec. 4.4 to illustrate the calibration performance of the PF combined with the GPR approach (Algorithm 4), and in Sec. 4.5 to demonstrate the performance of the proposed methods on real-world data. The two-factor transition model is used to test the calibrations based on Laplace approximation approach, see Sec. 4.3. In the experiments of this paper, we omit the observed process $(u_k)_{k=1,\dots,n}$, i.e., $u_k = 0$, for all $k = 1, \dots, n$ in Eq. (3), as it does not increase the complexity from methodological perspective but just adds a few unknown model parameters of the calibration. In practice, the choice of the observed factors u_k, or whether to include u_k, depends on the objective and data environment of the

Algorithm 3.4 (PF with GPR).

1. Define the grid for the value of model parameters.
2. Given each point (a vector of model parameter values) on the grid, run Algorithm 3.3 with the proposed importance sampling to obtain the corresponding log-likelihood.
3. Train the GPR on the grid with the corresponding log-likelihood. The trained GPR can be defined as a function between the model parameters and the log-likelihood, denoted by $f_{GPR}(\psi)$.
4. Find the estimator $\hat{\psi}$ such that

$$\hat{\psi} = \underset{\psi}{\operatorname{argmax}}\, f_{GPR}(\psi).$$

This $\hat{\psi}$ is the approximate ML estimator.

application. For instance, in stress testing or scenario analysis, incorporating observed common factors can be highly valuable. On the other hand, if the data points are too limited to capture stable relationships, it may be preferable to omit the observed factors.

We use simulated migration data for the experiments in Secs. 4.2–4.4, so that the "true" values of the model parameters are known and can be used as a benchmark. The simulation is done according to Eqs. (3) and (7) or (9), by assuming a one- or two-factor model for the latent process $(x_k)_{k\in\mathbb{N}^+}$. The process $(x_k)_{k\in\mathbb{N}^+}$ is simulated using the Monte Carlo method based on the autoregressive model in Eq. (3). Consequently, the migrations are simulated according to the model with Eq. (7) or (9). The simulated data are supposed to have 150 time points, i.e., one assumes to observe 150 periods of migrations. We are supposed to have four different ratings with three performance ratings (P1, P2 and P3) and one default rating (D). We further assume that the default state is absorbing, which means that the probabilities of the transition from D to P1, P2 or P3 are zero. The PF algorithm (Algorithm 3) with importance sampling is designed for low-default portfolios. We assume, for the PF assessments, the long-term average of the probability of defaults is $\overline{\text{PD}} = [0.001, 0.004, 0.01]$. The number of clients in these three ratings is assumed to be $[5000, 1000, 500]$. By contrast, the Laplace approximation is for portfolios with higher default probabilities and more number of clients. Therefore, for Laplace approximation assessments, we assume the long-term average $\overline{\text{PD}} = [0.01, 0.04, 0.1]$. The value of the d parameter defined after Eq. (3) is derived from the long-term average $\overline{\text{PD}}$. The number of clients in these three ratings is assumed to be $[100000, 10000, 5000]$.

Strictly speaking, the models (7) and (8) have three parameters: d, A and K. Specially the parameter d contains quite some elements. To reduce the number of parameters, the elements of d, in the calibration, we will use the auxiliary fact that

$$\mathbb{E}[\Phi(X)] = \Phi\left(\frac{\mu}{\sqrt{1+\sigma^2}}\right), \tag{24}$$

if $X \sim N(\mu, \sigma)$. To see that this relation holds, we argue as follows. We write the expectation as, with ϕ the standard normal density,

$$\mathbb{E}[\Phi(X)] = \int_{-\infty}^{\infty} \Phi(\mu + \sigma z)\phi(z)\,\mathrm{d}z = \int_{-\infty}^{\infty}\int_{-\infty}^{\mu+\sigma z} \phi(x)\phi(z)\,\mathrm{d}x\mathrm{d}z,$$

where we recognize that the double integral equals $\mathbb{P}(X \le \mu + \sigma Z)$ with X and Z independent standard normals. This probability trivially equals $\mathbb{P}(X - \sigma Z \le \mu)$. As $X - \sigma Z$ has a $N(0, 1+\sigma^2)$ distribution, $\mathbb{P}(X - \sigma Z \le \mu) = \Phi(\frac{\mu}{\sqrt{1+\sigma^2}})$, and we obtain (24).

Coming back to the calibration, to reduce the number of parameters to be estimated, the elements of d are estimated directly by using the average, denoted $\bar{r}$, of the observed PDs (for the default-only model (7)) or cumulative migration probabilities (for the migration model (8)) and the K parameter. So, we approximate the theoretical PD, according to the law of large numbers, by the sample average, i.e., $\bar{r}$, and invert the relation (24) with $\mu = d$ and $\sigma = K$ to get the approximation (estimate)

$$d \approx \sqrt{1 + K^2}\, \Phi^{-1}(\bar{r}).$$

Consequently, the model parameters left to be estimated are only A and K.

4.2. Laplace approximation and importance sampling

This section assesses the accuracy of the proposed Laplace and the PF approximation of the likelihood function. The likelihood profile is generated based on the simulated migration data using the one-factor default-only model (7). The one-dimensional latent process $(x_k)_{k=1,\ldots,n}$ is as described in Eq. (3) with $A = 0.7$ and $Q = \sqrt{1 - A^2} \approx 0.71414$. The sensitivity parameter K is set to be 0.6 for low-default portfolio and 0.3 for high-default portfolio. The performance of the PF is first assessed on a low-default portfolio as described in Sec. 4.1. The assessment of the PF consists of two parts: the assessment of the importance sampling and the assessment of the likelihood function. The Laplace approximation is tested on a portfolio with higher default probabilities and a higher number of clients, and the PF is used as a benchmark.

4.2.1. *Particle filter with importance sampling*

The importance density used in the proposed PF algorithm is a Gaussian density with mean and covariance approximated using the Laplace approach. The approximations are obtained by running the Kalman filter algorithm on model (16). Figures 1 and 2 present the comparison of the estimated latent process $(x_k)_{k=1,\ldots,n}$ using Laplace approximation and PF without importance sampling. In Fig. 1, the "true" parameters are used, i.e., $A = 0.7$ and $K = 0.6$. One observes that the Laplace approximation aligns with the PF approximations. In contrast, in Fig. 2, the K parameter is set to be 0.3. In this case, one observes that the Laplace approximation still aligns with the PF approximation with enough Monte Carlo points (100,000). When the number of Monte Carlo points is not sufficiently large, one observes that the PF without importance sampling gives a biased estimation of the x_k. The results indicate that the Laplace approach provides a good estimate of the x_k and hence can be used for importance sampling.

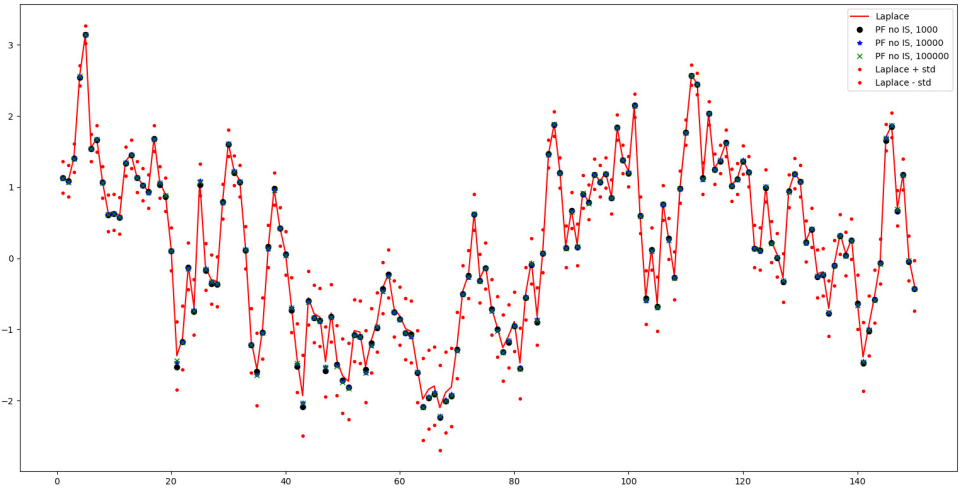


Fig. 1. Estimated x_k: Laplace versus PF without importance sampling. We use $A = 0.7$ and $K = 0.6$. The Laplace approximation and the 68% confidence interval (i.e., mean $\pm$ one standard deviation) are shown in red line and dots, respectively. The PF without importance sampling uses 1,000, 10,000, 100,000 Monte Carlo samples and is shown in black dots, blue stars and green crosses, respectively. Because of the high accuracy of the procedure, the blue stars and green crosses nearly coincide with the black dots and are barely visible.

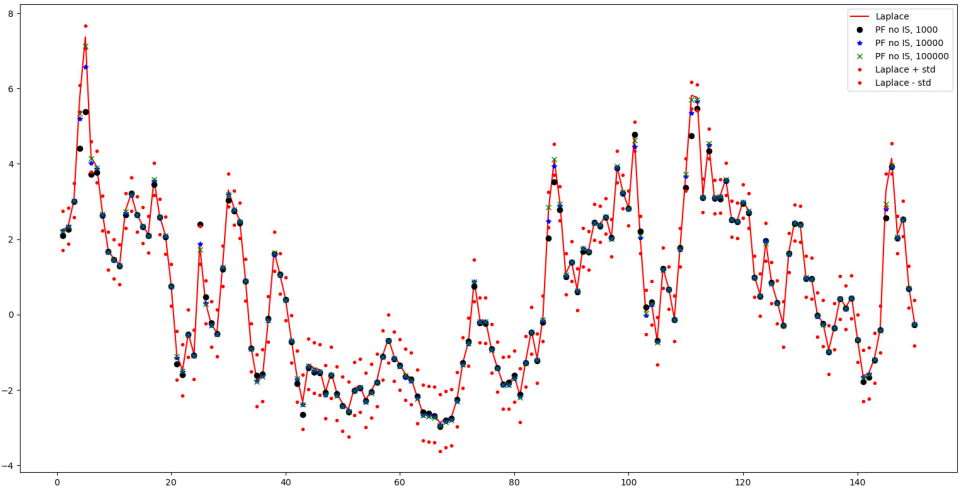


Fig. 2. Estimated x_k: Laplace versus PF without importance sampling. We use $A = 0.7$ and $K = 0.3$. The Laplace approximation and the one standard deviation confidence interval are shown in red line and dots, respectively. The PF without importance sampling uses 1,000, 10,000 and 100,000 Monte Carlo samples and is shown in black dots, blue stars and green crosses, respectively.

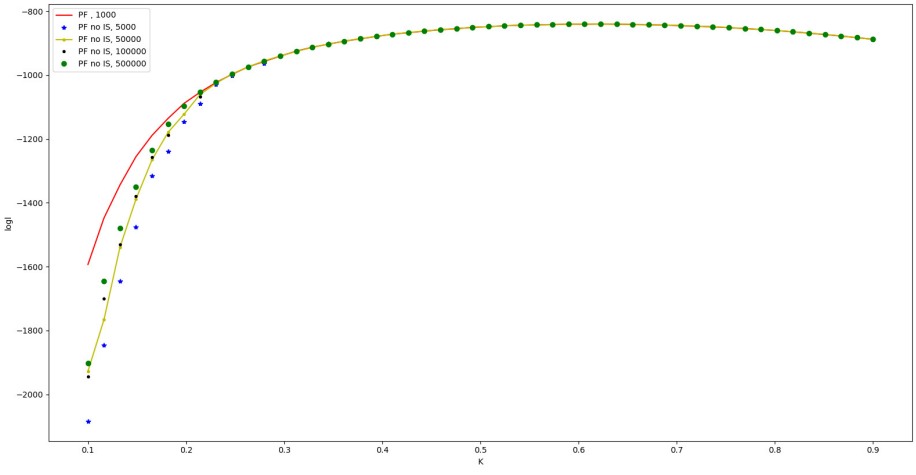


Fig. 3. Log-likelihood profile: K versus log-likelihood. We use $A = 0.7$. The log-likelihood profile using PF with the proposed importance sampling is shown as the red line. The log-likelihood profiles based on PF without importance sampling are used with 5,000, 50,000, 100,000 and 500,000 Monte Carlo samples and are shown in blue stars, yellow dot line, black dots and green dots, respectively.

Figures 3 and 4 show the comparison of the log-likelihood profile, with different K and A, approximated by the PF with and without the proposed importance sampling. One observes that the PF without importance sampling converges to the PF with the importance sampling, but much slower. Especially, Fig. 3 shows that

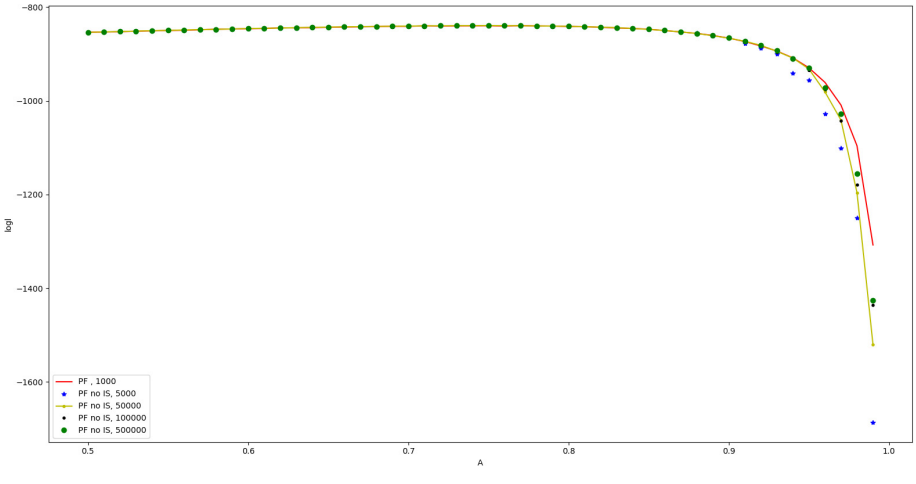


Fig. 4. Log-likelihood profile: A versus log-likelihood. We use $K = 0.6$. The log-likelihood profile using PF with the proposed importance sampling is shown in red. The log-likelihood profiles based on PF without importance sampling used 5,000, 50,000, 100,000 and 500,000 Monte Carlo samples and are shown in blue stars, yellow dot line, black dots and green dots, respectively.

the PF without importance sampling converges very slowly when the K value is significantly smaller than the optimal (where the ML is obtained) value. That is because the volatility of the *realized*x_k needs to be much larger to compensate for the low value of K and hence match the volatility of the signal θ_k implied from the data. Since the volatility of x_k is assumed to be $Q = \sqrt{1 - A^2}$, the high volatility of x_k requires the PF without importance sampling to generate extreme values for η_k in the simulation and consequently requires a large number of Monte Carlo samples. Same reasoning applies to the cases with high value of A parameters, see Fig. 4. When A is large, the volatility of x_k is small. Therefore, extreme values of η_k need to be sampled to reach the volatility of θ_k implied from the data. This will significantly slow down the convergence speed when an appropriate importance sampling is not used. These results suggest that the PF with the proposed importance sampling significantly increases the efficiency of the normal PF algorithm.

4.2.2. *Laplace approximation*

The Laplace approximation is designed for high-default portfolios with a relatively large number of observations. Figures 5 and 6 present the comparison between the

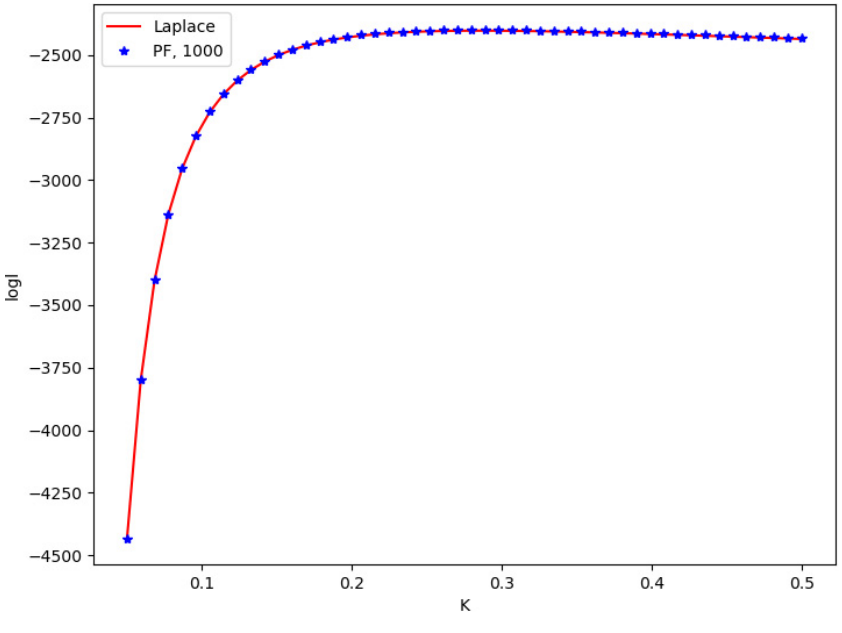


Fig. 5. Log-likelihood profile: K versus log-likelihood. $A = 0.7$. The log-likelihood profiles using Laplace approximation and PF with the proposed importance sampling are shown in red line and blue stars, respectively.

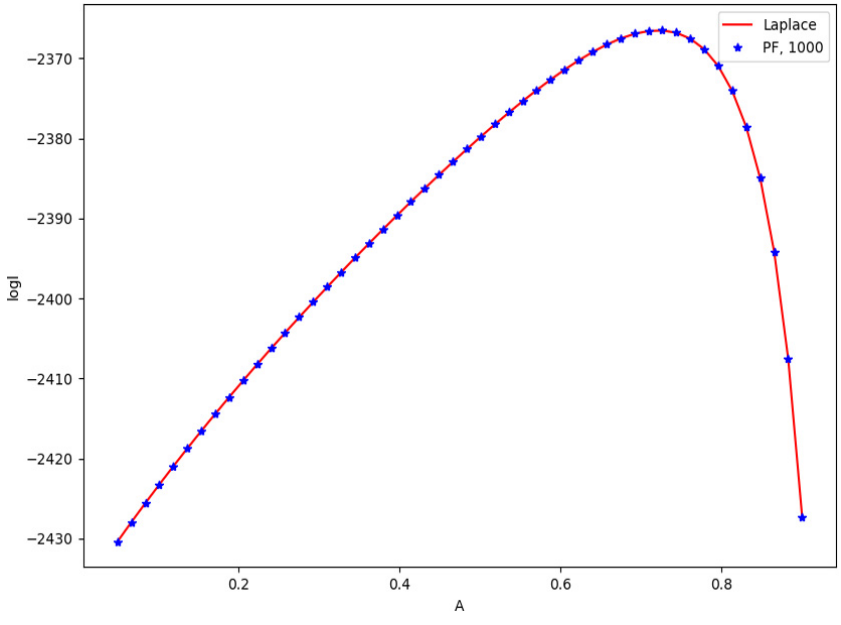


Fig. 6. Log-likelihood profile: A versus log-likelihood. $K = 0.3$. The log-likelihood profiles using Laplace approximation and PF with the proposed importance sampling are shown in red line and blue stars, respectively.

Laplace approximation and the PF with importance sampling. One observes that the log-likelihood profile of the Laplace approximation matches the profile of the PF, which indicates a good precision of the Laplace approximation of the likelihood function.

4.3. Calibration using laplace approximation

This section presents the calibration analysis using the Laplace approximation method. The migration model used is the two-factor transition model as in Eq. (8). The model parameters used for the simulation are specified as follows:

$$A = \mathrm{diag}([a_d, a_p]) = \mathrm{diag}([0.7, 0.8]),$$

$$Q = \begin{pmatrix} \sqrt{1-A^2(1,1)} & 0 \\ 0 & \sqrt{1-A^2(2,2)} \end{pmatrix} * \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} * \begin{pmatrix} \sqrt{1-A^2(1,1)} & 0 \\ 0 & \sqrt{1-A^2(2,2)} \end{pmatrix}$$

$$= \begin{pmatrix} 0.7141 & 0 \\ 0 & 0.7141 \end{pmatrix} * \begin{pmatrix} 1 & 0.4 \\ 0.4 & 1 \end{pmatrix} * \begin{pmatrix} 0.6 & 0 \\ 0 & 0.6 \end{pmatrix},$$

$$K = \mathrm{diag}([k_d, k_p]) = \mathrm{diag}([0.3, 0.2]).$$

The d parameter is set such that the long-term average migration probabilities are

$$[\bar{T}_{1R}, \bar{T}_{2R}, \bar{T}_{3R}] = [0.01, 0.04, 0.1],$$

$$\bar{T}^{ND} = \begin{pmatrix} T_{11}^{ND} & T_{12}^{ND} & T_{13}^{ND} \\ T_{21}^{ND} & T_{22}^{ND} & T_{23}^{ND} \\ T_{31}^{ND} & T_{32}^{ND} & T_{33}^{ND} \end{pmatrix} = \begin{pmatrix} 0.85 & 0.1 & 0.05 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.2 & 0.7 \end{pmatrix}.$$

To test the calibration method, 1,000 different scenarios of migrations are simulated based on the parameter values above. The Laplace method is applied to calibrate the model parameters A and K using each scenario of simulated migrations. The d parameters are derived beforehand according to the average of the observed cumulative migration probabilities of each scenario, using Eq. (24), similar to the procedure in Sec. 4.1

4.3.1. *Calibration results*

Table 1 presents the statistics of the estimates in the 1,000 scenarios. One observes a very good match between the average of the estimates and the "true" parameters defined at the beginning of Sec. 4.3. The variance of the estimates mainly comes from the randomness of the Monte Carlo sampling. For example, the model assumes a unit variance of each of the elements of the x_k, i.e., the equilibrium distribution of x_k is such that the variance of the elements is equal to one. Hence, the covariance matrix of the idiosyncratic variable η_k must then be $I - A^2$ as A is diagonal. But in the simulation, the realized variance of the simulated η_k could deviate from this assumption and hence impact the estimation of the K parameters. To illustrate this, in the simulation, we re-normalize the Monte Carlo samples of η_k so that its realized covariance matrix is always equal to $I - A^2$. The re-calibration results are presented in Table 2. One observes that the average of the K estimates is closer to the "true" values and the standard deviation is much smaller. Other factors which could impact the variance of the estimates are the values of the model parameters. For example, in a one-dimensional model for the x_k, the bigger the autocorrelation parameter A is, the bigger the autocovariance of the AR(1) process is. Consequently it creates more uncertainty in the estimation of the K

Table 1. Average and the standard deviation of the estimates based on the Laplace method.

	a_d	a_p	k_d	k_p	ρ
Average	0.6768	0.7732	0.2962	0.1976	0.3998
Std	0.0550	0.0493	0.0264	0.0217	0.0705

Table 2. Average and the standard deviation of the estimates based on the Laplace method with re-normalized η_k.

	a_d	a_p	k_d	k_p	ρ
Average	0.6767	0.7732	0.2985	0.2002	0.3994
Std	0.0551	0.0494	0.0086	0.0080	0.0705

Table 3. Average and the standard deviation of the estimates based on the Laplace method with resetting $a_d = 0.3$ and $a_p = 0.4$

	a_d	a_p	k_d	k_p	ρ
Average	0.2887	0.3998	0.2962	0.1976	0.3998
Std	0.0685	0.0703	0.0182	0.0133	0.0702

parameters. On the other hand, the bigger A is, the smaller the variance of η_k is (keeping the variance of the x_k fixed), and hence the smaller the Monte Carlo error of the estimation of A parameters is. Table 3 presents the calibration results of the model with A reset to $\mathrm{diag}([0.3, 0.4])$ and the other parameters kept the same. Compared to Table 1, one observes that with a smaller A value the variance of the K estimates decreases while the variance of the A estimates increases.

4.3.2. *Stepwise calibration results*

An alternative procedure to calibrate the transition model is a stepwise approach, aimed at reducing the parameter dimensionality. Note that the conditional likelihood of the observed migrations of row i of the two-factor transition model, see Eq. (8), can be rewritten as, where we use $C_{i,k} = \frac{N_{i,k}!}{\prod_{j=1}^{R} m_{ij,k}!}$,

$$\begin{aligned}
&\log p(m_{i,k}|\theta_k) \\
&\quad = \sum_{j=1}^{R} m_{ij,k} \log T_{ij,k} + \log C_{i,k} \\
&\quad = m_{iR,k} \log T_{iR,k} + \sum_{j=1}^{R-1} m_{ij,k} \log T_{ij,k} + \log C_{i,k} \\
&\quad = m_{iR,k} \log T_{iR,k} + \sum_{j=1}^{R-1} m_{ij,k} [\log(1 - T_{iR,k}) + \log (T_{ij,k})^{ND}] + \log C_{i,k}
\end{aligned}$$

$$= \underbrace{m_{iR,k} \log T_{iR,k} + (N_{i,k} - m_{iR,k}) \log(1 - T_{iR,k})}_{\text{default only}}$$
$$+ \underbrace{\sum_{j=1}^{R-1} m_{ij,k} \log(T_{ij,k}^{ND})}_{\text{migration given no default}} + \log C_{i,k},$$

with $N_{i,k}$ the number of clients in rating i at time k. Therefore, one can separate the calibration of the default and nondefault migrations of the transition model. This stepwise approach works as follows:

(1) Calibrate the default-only model, with the observed number of defaults, number of clients, and PD, at time k for rating i, equal to $m_{iR,k}$, $N_{i,k}$ and $T_{iR,k}$, respectively. The outputs of the calibration are the a_d and k_d parameters for the default part of the model and the latent process $x_k^{(D)}$.
(2) Calibrate the migration model, with observed number of migrations and the migration probabilities, at time k for rating $i = 1, \ldots, R-1$ and $j = 1, \ldots, R-1$, equal to $m_{ij,k}$ and $T_{ij,k}^{ND}$, respectively. The outputs of the calibration are the a_p and k_p parameters for the nondefault part of the model and the latent process $x_k^{(P)}$.
(3) Empirically estimate the correlation parameter ρ by using the calibrated parameter A and the realized process $[x_k^{(D)}, x_k^{(P)}]$ obtained from steps 1 and 2.

Tables 4 and 5 present the calibration results of the stepwise calibration and the comparison with the standard two-factor model calibration. One observes that the stepwise approach produces very similar results as the standard approach. The advantage of the stepwise approach is that it reduces the parameter dimensionality in the calibration. This can be very helpful to improve the efficiency of the calibration.

4.4. Calibration using the particle filter

In this section, we test the calibration performance of the proposed PF with the GPR approach. Since the calibration is assessed using 1,000 different scenarios, as

Table 4. Average and the standard deviation of the estimates based on the Laplace method with the stepwise approach.

	a_d	a_p	k_d	k_p	ρ
Average	0.6775	0.7709	0.2901	0.1897	0.3947
Std	0.0585	0.0528	0.0277	0.0226	0.0703

Table 5. Average absolute error of the stepwise approach compared with the standard two-factor model calibration.

	a_d	a_p	k_d	k_p	ρ
Average error	0.0058	0.0065	0.0153	0.0174	0.0049

in Sec. 4.3, for the sake of a moderate running time, we consider the default-only case, the model of (7). But note that this calibration exercise can be rather straightforwardly extended to the cases of full migration matrix calibration. The set-up of the default-only model in this experiment is the same as in Sec. 4.2, namely, a high-default case and a low-default case. Note that although the PF calibration approach is specially designed for the low-default portfolio, it is in fact a generic method and can be used for both high- and low-default cases. In this assessment, the purpose of the calibration to a high-default portfolio is to compare it with the calibration using the Laplace approach of Sec. 4.3. We build a Cartesian grid for the training GPR. The Cartesian grid is two-dimensional, with one dimension for parameter a and another dimension for parameter k. Each coordinate ranges on a set of 20 evenly spaced points between 0.1 and 0.9. Therefore, in total there are 400 pairs of points (A, K) on the Cartesian grid. For each pair of points, the proposed PF algorithm (Algorithm 4) with 1,000 Monte Carlo points is run to obtain estimates of the corresponding log-likelihood function. Then the GPR is trained on the Cartesian grid with inputs (A, K) and outputs the corresponding log-likelihood. The training (and prediction) of the GPR uses the `scikit-learn` Python package, with a radial basis function (RBF) kernel (i.e., a squared-exponential kernel) plus a WhiteNoise kernel. The RBF kernel is generally regarded as the standard or default choice in GPR applications, see Rasmussen and Williams (2005) and Seeger (2004). It is one of the most widely used kernels, due to its favorable balance of smoothness, simplicity, and flexibility. By imposing infinitely differentiable sample paths, the RBF kernel enforces smooth functions, which aligns well with the requirements of likelihood function approximations (Schölkopf and Smola, 2001). Its simple parametric form, with only a variance and a length-scale as hyperparameters, makes it both interpretable and computationally convenient (Bishop, 2006). Despite this simplicity, the RBF kernel is a universal kernel, capable of approximating any continuous function on a compact set to arbitrary accuracy, see Steinwart (2001). Furthermore, the addition of a white noise kernel accounts explicitly for the noise from the training set (Monte Carlo error) and ensures numerical stability of the covariance matrix, preventing overconfident interpolation of noisy data.

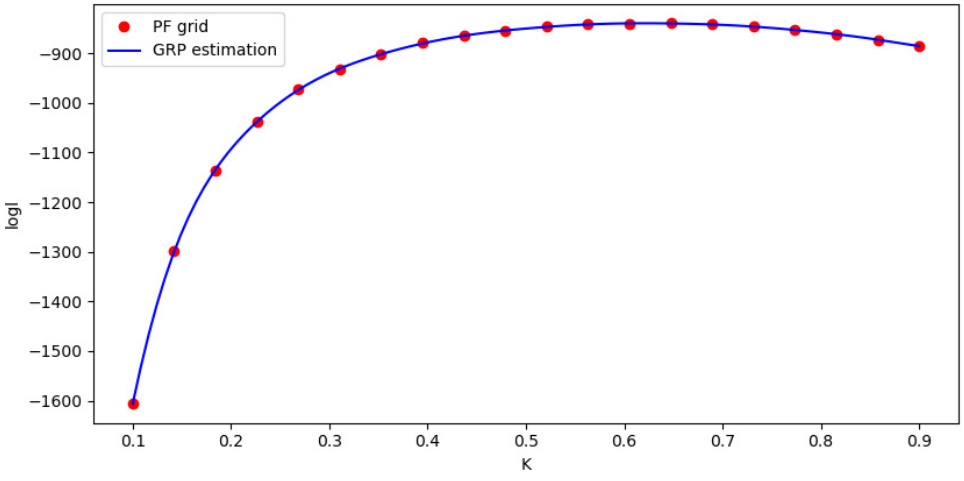


Fig. 7. Log-likelihood profile for k when $a = 0.73$. The red dots are the log-likelihoods of the training grid. The blue line is the predicted log-likelihood using the trained GRP, when $a = 0.73$.

Figure 7 shows an instance of the prediction of the trained GPR compared with the training samples. One can see a very good prediction of the log-likelihood from the trained GPR.

Once the GPR is trained, MLE is applied to the trained GPR function to find the estimator for the model parameters a and k. As in the experiments in Sec. 4.3, 1,000 scenarios of migrations are generated based on the proposed "true" parameters. For each scenario, the PF with the GPR method is applied to calibrate the model parameters.

4.4.1. *Calibration results*

Tables 6 and 7 present the averages and standard deviations of the estimated parameter values based on the PF with GPR approach, for both high- and low-default cases. One observes that, for the high-default case, the PF approach produces very similar results as the Laplace approach, which justifies the accuracy of the PF approach. It is also interesting to see that the standard deviations in the PF case are slightly higher, due to the Monte Carlo error in PF and the bias of the GPR approximation. For the low-default case, the results also suggest a good performance of the PF calibration. One observes the bias in the k parameter estimation and a higher standard deviation. This is mainly because of the low-default nature. Namely, there is only a limited number of default cases observed and hence these introduce more estimation errors.

Table 6. Average and the standard deviation of the estimators from PF approach. "True" value of k_d is 0.3.

	a_d	k_d
Average	0.6720	0.2903
Std	0.0634	0.0290

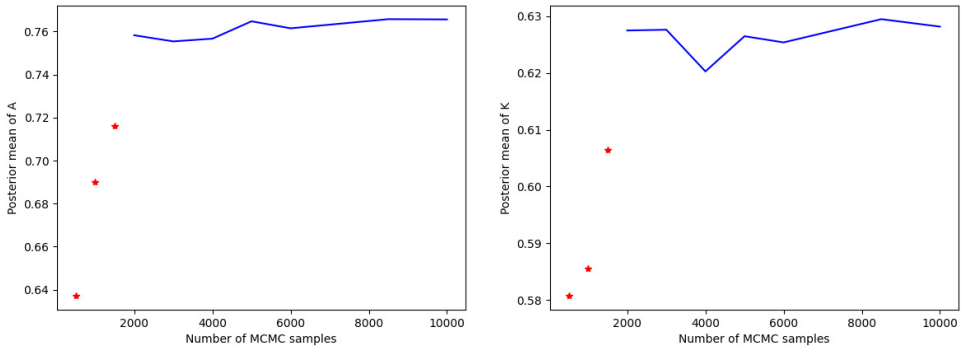
Table 7. Average and the standard deviation of the estimators of the PF approach for the low-default case. "True" value of k_d is 0.6.

	a_d	k_d
Average	0.7211	0.5518
std	0.0714	0.0993

4.4.2. *Comparison with the existing methods*

We further evaluate the performance of the proposed calibration method by comparing it with two widely used benchmark approaches, namely the default-rates-based approach (Koopman *et al.*, 2005) and the MCMC-based approach (McNeil and Wendin, 2007). The MCMC method, in particular, requires specifying a burn-in period to exclude the initial, nonstationary samples produced before the Markov chain reaches its equilibrium distribution. The length of the burn-in period depends on the specific setting and convergence speed of the chain. In our implementation, it is set to 1,500 iterations, meaning that the first 1,500 simulated draws are discarded, and only the subsequent samples are retained as effective observations. Figure 8 illustrates the convergence behavior of the posterior means of the two estimated parameter distributions obtained from the MCMC procedure. The red stars correspond to posterior means computed from samples before the burn-in period, while the blue line shows those based on the effective MCMC samples after the burn-in phase (therefore the burn-in period is excluded). One observes that the posterior means stabilize after approximately 2,500 iterations, indicating satisfactory convergence of the MCMC implementation with 1,500 burn-in iterations.

Table 8 presents a detailed comparison among the proposed PF approach, the MCMC approach with 10,000 samples, and the default-rates-based approach in



(a) Posterior mean of parameter *ad*

(b) Posterior mean of parameter *kd*

Fig. 8. Convergence of the posterior mean of the parameters in MCMC approach. The burn-in period is 1,500 iterations. The red stars are the means of the MCMC samples before the burn-in period. The blue line shows the means of the *effective* MCMC samples after the burn-in period.

Table 8. Comparison between the proposed approach and the existing approaches. The values of a_d and k_d used to simulate the data are 0.7 and 0.6, respectively.

	a_d	k_d
Proposed PF	0.7644	0.6326
MCMC 10000	0.7657	0.6281
Default-rate-based	0.0714	0.0993

terms of the estimated parameters. As shown, the proposed PF and the MCMC methods yield parameter estimates that are highly consistent with each other and closely aligned with the true values used to generate the simulated data. In contrast, the estimates obtained from the default-rates-based approach deviate noticeably from the true values, due to the biased estimation of the default rates when the number of observations is limited.

In addition to assessing the accuracy of the estimation, the computational efficiency of the proposed method is also examined by comparing the proposed PF approach with the MCMC approach. For a fair comparison on the running time, the MCMC procedure is configured to produce 2,500 total samples, corresponding to 1,000 effective samples after discarding the 1,500 burn-in iterations. Both algorithms were executed on a machine equipped with an AMD Ryzen 5 PRO 7545U CPU (6 cores, 3.20 GHz base clock), 32 GB DDR5 RAM. All implementations were conducted in Python 3.12.8 using NumPy 1.26.4 and SciPy 1.14.1, and the runtime is measured with the "*time*" module. The results clearly

Table 9. Comparison between the proposed approach and the MCMC approach with 2,500 samples.

	Runtime (s)
Proposed PF	42.23
MCMC 2500	873.26

demonstrate that the proposed PF approach is more than 20 times faster than the MCMC approach, while achieving comparable accuracy. These findings highlight both the robustness and computational efficiency of the proposed calibration procedure (Table 9).

4.5. Calibration on the empirical data

In this section, the proposed calibration method is applied to the empirical S&P data. The S&P rating migration data is from January 1999 to August 2025 with quarterly frequency, yielding a total of 106 time points. The dataset contains credit rating transition records for 9,804 firms, primarily from the United States and the United Kingdom, with additional representation from other European and Asian economies. The dataset therefore reflects a broad and diversified sample across industries and geographical regions, capturing several economic cycles, including partly the dot-com recession, the 2008 global financial crisis, and the COVID-19 pandemic period.

We focus on the default-only version of the model. The model is calibrated using the proposed Laplace approximation approach and the proposed PF with GPR approach, and their results are benchmarked against those obtained from the MCMC method with 10,000 samples, including 1,500 burn-in iterations and 8,500 effective samples. Table 10 summarizes the estimated parameters.

The results show that the parameter estimates obtained from the proposed PF and the MCMC approaches are highly consistent, confirming the reliability of the proposed calibration method on real-world data. In particular, the proposed PF methods and the MCMC approaches produce almost identical estimates on

Table 10. Calibration results on the S&P data.

	a_d	k_d
Proposed Laplace approximation	0.1822	0.2068
Proposed PF with GPR	0.1936	0.2084
MCMC 10000	0.1952	0.2086

the parameters, demonstrating that the PF effectively replicates the performance of the MCMC benchmark while being computationally much more efficient. The proposed Laplace approximation method also yields an estimate of k_d that is nearly identical to those obtained from the other two approaches, although it shows a slight bias in the autocorrelation parameter. This bias is due to the relatively low-default frequency in the dataset (firms in the sample typically constitute a low-default portfolio) and the finite sample size of 9,804 observations, which limits the precision of the Laplace-based approximation. Nevertheless, the Laplace approximation produces an estimated autocorrelation of 18%, which is very close to the 19% obtained from both the PF and MCMC methods.

Overall, all three approaches produce consistent results, with an estimated asset correlation of $\rho = k_d^2 = 4.34\%$, and an autocorrelation parameter of approximately 20%. The asset correlation reflects the degree of dependence between firms' latent asset returns, representing the extent to which their default risks move together due to shared macroeconomic or systemic factors. An estimated correlation of 4.34% indicates a relatively low, yet nonnegligible, level of comovement among firms' credit qualities. This magnitude is consistent with empirical evidence from recent studies such as Batema (2024) and Jacobs (2025), which report asset correlations in the range of 3–6% for similarly diversified corporate portfolios.

In summary, the results demonstrate that the proposed PF and Laplace approximation methods both provide accurate and robust parameter estimates. While the Laplace approximation exhibits a minor bias in autocorrelation due to data sparsity, it remains computationally efficient and sufficiently accurate. The PF approach, in particular, achieves performance comparable to the MCMC benchmark while offering significant improvements in speed and scalability, making it a practical tool for large-scale empirical credit risk applications.

5. Conclusions and Future Study

This paper proposes two calibration algorithms, tailor-made for the credit rating transition models of high- and low-default portfolios, respectively. The algorithm for the high-default portfolio uses the Laplace method, incorporated with the Kalman filter algorithm, which is used to estimate the mode of the posterior distribution of the signals and compute the (approximated) likelihood function of the observed transition matrices. We show that the same Kalman filter algorithm can be applied to both mode estimation and likelihood calculation. Therefore, the numerical ML algorithm is very fast. For the low-default portfolio, since the number of the observed migrations (defaults) is limited, the performance of the Laplace approximation is not assured and hence might introduce biased in the likelihood approximations and ML estimates. As an alternative, we develop an

algorithm based on a PF with importance sampling. The importance density is obtained from the mode estimation in the Laplace algorithm. Experiments show that mode estimation produces very accurate estimates of the posterior of the latent states and as a result, the designed importance sampling significantly improves the efficiency of the PF. Moreover, the GPR is used to smooth the likelihood function approximated by the PF, so that a numerical optimization algorithm can be easily applied to obtain the ML estimates. The efficiency and the accuracy of these two proposed calibration algorithms are substantiated by numerical experiments.

This paper has focused primarily on the development and evaluation of calibration methodologies. Future research should extend the analysis by examining the robustness of the proposed approaches to deviations from the assumed AR(1) dynamics of the latent factors, for instance by considering alternative specifications such as ARMA processes, stochastic volatility structures, or regime-switching models. Moreover, a systematic analysis of the effects of different response functions would provide deeper insight into the sensitivity of the results to modeling assumptions and enhance the general applicability of the proposed framework. In addition, it would be valuable to conduct comprehensive case-by-case studies on the convergence and accuracy of the Laplace approximation and to investigate more systematically the critical conditions under which this approximation is reliable or may fail.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments and constructive suggestions, which have improved the quality and clarity of this paper.

ORCID

Jian He https://orcid.org/0000-0002-5372-0541
Asma Khedher https://orcid.org/0000-0003-1528-3923
Peter Spreij https://orcid.org/0000-0002-6416-6320

Appendix A. State Space Model

State-space models deal with dynamic time series problems that involve unobserved variables or parameters that describe the evolution of the state of the

underlying system. The general *state space* model, defined on some probability space $\Phi, mathcalF, mathbbP$, is as follows:

$$\begin{aligned} x_k &= f_k(x_{k-1}, u_k),\ x_0, \\ y_k &= h_k(x_k, v_k), \quad k \in \mathbb{N}^+, \end{aligned} \tag{A.1}$$

where $f_k : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^d$, $h_k : \mathbb{R}^d \times \mathbb{R}^q \to \mathbb{R}^m$ are given Borel measurable functions, $\{u_k\}_{k\in\mathbb{N}^+}$ is a p-dimensional and $\{v_k\}_{k\in\mathbb{N}^+}$ is a q-dimensional white noise processes both independent of the initial condition x_0, and mutually independent as well. Parameters in the functions f_k and h_k, together with the covariance of u_k and v_k, can be seen as the parameters of the state space model, to which we collectively refer to as ψ.

Appendix B. Bayesian Filter

The Baysian filters, see for instance Press (2003) and Robert (2007), are often used to estimate the latent process $(x_k)_{k=1,\ldots,n}$ or to compute the likelihood function of the observations y_k in a state space model (A.1) given the model parameters. We define the initial density function p_0 of x_0. The methodology in Bayesian filtering consists of two parts: prediction and update. At every time point k, the prediction part computes (estimates) the *prior distribution (density)* of x_k (a time k given the past observations up to time $k-1$),

$$p(x_k|y_{1:k-1}, \psi)$$

and the update part computes (estimates) the *posterior distribution (density)* of x_k given the past up to time k,

$$p(x_k|y_{1:k}, \psi).$$

It follows that the state space model (A.1) satisfies the properties of a stochastic system, i.e., at every (present) time $k \geq 1$ the future states and future observations (x_j, y_j), $j \geq k$, are conditionally independent from the past states and observations (x_j, y_{j-1}), $j \leq k$, given the present state x_k, see van Schuppen (1989). It then follows that $(x_k)_{k\in\mathbb{N}}$ is a Markov process, and for every $k \geq 1$ one has that y_k and $y_{1:k-1}$ are conditionally independent given x_{k-1}, in terms of densities,

$$p(x_k|x_{k-1}, y_{1:k-1}, \psi) = p(x_k|x_{k-1}, \psi). \tag{B.1}$$

Similarly, due to the Markov property, x_k and $y_{k+1:T}$ are independent given x_{k+1}, which gives

$$p(x_k|x_{k+1}, y_{1:T}, \psi) = p(x_k|x_{k+1}, y_{1:k}, \psi). \tag{B.2}$$

Moreover, one also has, for every $k \geq 1$, that y_k and $y_{1:k-1}$ are conditionally independent given x_{k-1}, in terms of densities,

$$p(y_k|y_{1:k-1}, x_k, \psi) = p(y_k|x_k, \psi), \quad \text{for} \quad k \in \mathbb{N}^+. \tag{B.3}$$

Equations (B.1)–(B.3) are used later in the derivation of the Bayesian filter and smoother recursions.

Using Bayes' rule and Eq. (B.1), we deduce that the density function of the prior distribution is given by

$$\begin{aligned} p(x_k|y_{1:k-1}, \psi) &= \int p(x_k|x_{k-1}, y_{1:k-1}, \psi) p(x_{k-1}|y_{1:k-1}, \psi)\, \mathrm{d}x_{k-1} \\ &= \int p(x_k|x_{k-1}, \psi) p(x_{k-1}|y_{1:k-1}, \psi)\, \mathrm{d}x_{k-1}\,. \end{aligned} \tag{B.4}$$

Note that when $k = 1$, the posterior distribution $p(x_{k-1}|y_{1:k-1}, \psi)$ is defined as the initial density p_0.

The purpose of the Bayesian algorithm is to sequentially compute the posterior distribution $p(x_k|y_{1:k}, \psi)$. Again using Bayes' rule, (B.3) and (B.4), we obtain the posterior

$$\begin{aligned} p(x_k|y_{1:k}, \psi) &= \frac{p(y_k|x_k, y_{1:k-1}, \psi) p(x_k|y_{1:k-1}, \psi)}{p(y_k|y_{1:k-1}, \psi)} \\ &= \frac{p(y_k|x_k, \psi) p(x_k|y_{1:k-1}, \psi)}{p(y_k|y_{1:k-1}, \psi)} \end{aligned} \tag{B.5}$$

with the conditional likelihood

$$p(y_k|y_{1:k-1}, \psi) = \int p(y_k|x_k, \psi) p(x_k|y_{1:k-1}, \psi)\, \mathrm{d}x_k. \tag{B.6}$$

If we assume the marginal likelihood function $p(y_k|x_k, \psi)$ and the transition probability $p(x_k|x_{k-1}, \psi)$ are known, then given the posterior distribution $p(x_{k-1}|y_{1:k-1}, \psi)$ at time $k-1$, we can use Eqs. (B.4) and (B.5) to compute the posterior measure $p(x_k|y_{1:k}, \psi)$ at time k. In this way the posterior distributions can be computed recursively given the initial distribution p_0.

While the mathematical derivations of the Bayesian filter and smoother are straightforward, in practice it is always a big challenge to compute the integrals in Eq. (B.4) or (B.6). Although there are some special cases where theoretical formulas are available for the integrals, such as the Kalman filter or (more general) with conjugate priors, in most cases one has to find approximations or numerical algorithms to compute these integrals. In the next two subsections, we introduce two types of the most used Bayesian filters in practice, the Kalman filter and its extensions, and the PF.

B.1. Kalman filter

Assume that the state and observations in (A.1) evolve according to a linear Gaussian model. That is the functions f_k and h_k, for $k = 1, \ldots, n,$ have to take linear forms as follows:

$$\begin{aligned} y_k &= K_k x_k + \varepsilon_k, \\ x_{k+1} &= A_k x_k + \eta_k, \end{aligned} \tag{B.7}$$

where K_k is a $m \times d$ matrix and A_k is a $d \times d$ matrix and the following distributional assumptions are made. The initial variable x_0 is assumed to be Gaussian, the ε_k and η_k are assumed to be serially independent and independent of each other at all time points, with Gaussian distributions, for $k = 1, \ldots, n$,

$$\begin{aligned} \varepsilon_k &\sim N(0, H_k), \\ \eta_k &\sim N(0, Q_k), \\ x_0 &\sim N(a_0, P_0), \end{aligned}$$

where $N(\mu, \Sigma)$ denotes the Gaussian distribution with mean μ and covariance Σ. We also use the generic notation $N(x; \mu, \Sigma)$ to denote the density at x of this normal distribution. Due to the Gaussian assumptions and the linear structure of the model in (B.7), the prior and posterior distributions defined in (B.4) and (B.5) are Gaussian and can be analytically derived. Denote the estimates for the prior and posterior density at time k by $N(x; x_{k|k-1}, P_{k|k-1})$ and $N(x; x_{k|k}, P_{k|k})$, respectively. Here is the Kalman filter algorithm for the Gaussian linear model (B.7).

Extensions and generalizations of the Kalman filter approach have also been developed, such as the extended Kalman filter, see Einicke and White (1999) or Wan and Nelson (2002), and the unscented Kalman filter, see Wan and van der Merwe (2002), which work for nonlinear systems. However, the extended Kalman filter and the unscented Kalman filter do not account for the non-Gaussian properties of the model other than their first two moments, the mean and variance functions. Durbin and Koopman (2012, Secs. 10.6–10.7) proposed a mode estimation approach which captures non-Gaussian properties of the state space model.

B.2. Particle filter

In the *PF*, the prior and posterior distributions are estimated by a Monte Carlo method. With a Monte Carlo method, a certain density function $f(x)$ is generally estimated by

$$f(x) = \sum_{i=1}^{N} w^{(i)} \delta_{x^{(i)}}(x),$$

Algorithm B.1 (Kalman filter).

Initialization: Initialize the posterior distribution $p_0(x_0) = N(x_0; x_{0|0}, P_{0|0})$.

Iteration: For $k = 1, \ldots, n$, let $x_{k-1|k-1}$ and $P_{k-1|k-1}$ be given.

1. Compute the prior distribution $p(x_k|y_{1:k-1}) = N(x_k; x_{k|k-1}, P_{k|k-1})$, with

$$\begin{aligned} x_{k|k-1} &= A_k x_{k-1|k-1}, \\ P_{k|k-1} &= A_k P_{k-1|k-1} A_k^\top + Q_k. \end{aligned} \tag{B.8}$$

2. With $S_k = K_k P_{k|k-1} K_k^\top + H_k$, compute the Kalman gain

$$G_k = P_{k|k-1} K_k^\top (S_k)^{-1}. \tag{B.9}$$

3. Compute the posterior distribution $p(x_k|y_{1:k}) = N(x_k; x_{k|k}, P_{k|k})$, with

$$\begin{aligned} x_{k|k} &= x_{k|k-1} + G_k(y_k - K_k x_{k|k-1}), \\ P_{k|k} &= P_{k|k-1} - G_k K_k P_{k|k-1}. \end{aligned} \tag{B.10}$$

4. Compute the conditional log-likelihood $p(y_k|y_{1:k-1})$ as

$$p(y_k|y_{1:k-1}) = N(y_k; H_k x_{k|k-1}, S_k).$$

Likelihood: The log-likelihood of the observations, $\log p(y_{1:n})$, can be obtained by summing up the conditional log-likelihoods,

$$\log p(y_{1:n}) = \sum_{k=1}^{n} \log p(y_k|y_{1:k-1}).$$

where $\{x^{(i)}, i = 1, \ldots, N\}$ are i.i.d. random samples from a so-called *importance density*, $\{w^{(i)}, i = 1, \ldots, N\}$ are *the importance weights*, and $\delta_{x^{(i)}}$ are Dirac distributions. The key parts of the PF are to choose the importance density and to compute the importance weights, see, e.g., Doucet (1997). For the general state space model (A.1), suppose the posterior density $p(x_{k-1}|y_{1:k-1})$ at time $k-1$ is estimated by

$$p(x_{k-1}|y_{1:k-1}, \psi) \approx \sum_{i=1}^{N} w_{k-1}^{(i)} \delta_{x_{k-1}^{(i)}(x_{k-1})}.$$

Using Eqs. (B.4) and (B.5), the prior and posterior densities are, respectively, estimated by

$$p(x_k|y_{1:k-1}, \psi) \approx \sum_{i=1}^{N} w_{k-1}^{(i)} p(x_k|x_{k-1}^{(i)}, \psi),$$

$$p(x_k|y_{1:k},\psi) \approx \frac{\sum_{i=1}^{N} w_{k-1}^{(i)} p(y_k|x_k,\psi) p(x_k|x_{k-1}^{(i)},\psi)}{\int \sum_{i=1}^{N} w_{k-1}^{(i)} p(y_k|x_k,\psi) p(x_k|x_{k-1}^{(i)},\psi) \mathrm{d}x_k}. \tag{B.11}$$

These estimated densities are mixture distributions. If the samples $\tilde{x}_k^{(i)}$ are generated from the transition density $p(x_k|x_{k-1}^{(i)},\psi)$ for $i = 1,\ldots,N$, the estimates of prior and posterior in Eq. (B.11) can be reformulated so that they can be used in recursive calculations,

$$p(x_k|y_{1:k-1},\psi) \approx \sum_{i=1}^{N} \overset{(i)}{w}_{k-1} \delta_{\underset{\tilde{x}_k}{(i)}}(x_k),$$

$$p(x_k|y_{1:k},\psi) \approx \sum_{i=1}^{N} \overset{(i)}{w}_{k} \delta_{\underset{\tilde{x}_k}{(i)}}(x_k),$$

where the weights $w_k^{(i)}$ are defined by

$$w_k^{(i)} = \frac{w_{k-1}^{(i)} p(y_k|\tilde{x}_k^{(i)},\psi)}{\sum_{i=1}^{N} w_{k-1}^{(i)} p(y_k|\tilde{x}_k^{(i)},\psi)}. \tag{B.12}$$

Consequently one obtains the conditional likelihood $p(y_k|y_{1:k-1}) \approx \sum_{i=1}^{N} w_{k-1}^{(i)} p(y_k|\tilde{x}_k^{(i)},\psi)$.

This type of PF is often referred to as sequential Monte Carlo. It is a specific member of the family termed the bootstrap PF, see Gordon *et al.* (1993). In Doucet (1997) it is shown that the variance of the importance weights decreases stochastically over time. This will lead the importance weights to be concentrated on a small amount of sampled particles. This problem is called degeneracy. To address the rapid degeneracy problem, the sampling-importance resampling (SIR) method, see, e.g., Doucet (1997) and Pitt and Shephard (1999), is introduced to eliminate the samples with a low importance weight and multiply the samples with a high importance weight. In SIR, once the approximation of the posterior $p(x_k|y_{1:k},\psi) \approx \sum_{i=1}^{N} w_k^{(i)} \delta_{\tilde{x}_k^{(i)}}(x_k)$ is obtained, new, re-sampled, particles $x_k^{(j)}$ are i.i.d. sampled from this approximate posterior distribution, i.e., every $x_k^{(j)}$ is independently chosen from the $\tilde{x}_k^{(i)}$ with probabilities $w_k^{(i)}$ for $i = 1,\ldots,N$. This step can be accomplished by sampling integers j from $\{1,\ldots,n\}$ with probabilities $w_k^{(i)}$ for $i = 1,\ldots,N$. Then the new estimation on the posterior and conditional likelihood is given by

$$p(x_k|y_{1:k},\psi) \approx \frac{1}{N} \sum_{i=1}^{N} \delta_{\underset{\tilde{x}_k}{(j)}}(x_k),$$

$$p(y_k|y_{1:k-1},\psi) \approx \frac{1}{N}\sum_{i=1}^{N} p(y_k|\overset{(j)}{x}_k).$$

Appendix C. Gaussian Process Regression

In this section, we provide a brief introduction to *GPR*. For a good overview, we refer for example to Rasmussen and Williams (2005). We start by introducing Gaussian processes.

Definition 1. For any index set χ, a *Gaussian process* (GP) on χ is a set of random variables $(f(x), x \in \chi)$ such that $\forall n \in \mathbb{N}$ and $x_1, \ldots, x_n \in \chi$, $(f(x_1), \ldots, f(x_n))$ is a multivariate Gaussian random variable.

In Definition 1, $f : \chi \times \Omega \to \mathbb{R}$ and the index set χ is the set of possible inputs. From now on, we specify χ to be $\mathbb{R}^d$.

A GP is completely specified by its *mean* and *covariance* functions. We define the mean function $m : \mathbb{R}^d \to \mathbb{R}$ and the covariance $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ of f as

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})], \\ k(\mathbf{x},\mathbf{x}') &= \mathbb{E}[f(\mathbf{x}) - m(\mathbf{x})][f(\mathbf{x}') - m(\mathbf{x}')]. \end{aligned}$$

Given n pairs of observations $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$, $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$ for $i = 1, \ldots, n$, the GPR model is a probabilistic nonparametric model with the following structure:

$$Y = \mathbf{f}(X) + \varepsilon,$$

where $Y = [y_1, y_2, \ldots, y_n]^T$ are the outputs, $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]^T$ are the inputs, ε is a Gaussian noise with a vector mean $\mathbf{0}$ and variance $\sigma^2\mathbf{I}$, and $\mathbf{f} : \mathbb{R}^d \times \cdots \times \mathbb{R}^d \to \mathbb{R}^n$ is such that $\mathbf{f}(X) = [f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots, f(\mathbf{x}_n)]^T$ follows a multivariate Gaussian distribution specified by the GP mean $m(x)$ and covariance functions $k(\mathbf{x}_i, \mathbf{x}_j)$, $1 \le i, j \le n$. Namely $\mathbf{f}(X) \sim N(\mu, K)$, where K is the $n \times n$ covariance matrix of which the (i,j)th element $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ and for simplicity, we take the mean vector $\mu = \mathbf{0}$.

To predict $Y_* = [y_{*1}, \ldots, y_{*m}]^T$ at the test locations $X_* = [\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}]^T$, the joint distribution of the training observations Y and the predictive targets Y_* is given by

$$\begin{bmatrix} Y \\ Y_* \end{bmatrix} \sim N\left(\mathbf{0}, \begin{bmatrix} K(X,X) + \sigma^2\mathbf{I} & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix}\right), \tag{C.1}$$

where $K(X,X) = K$, $K(X_*,X)$ is an $m \times n$ matrix with (i,j)th element $[K(X_*,X)]_{ij} = k(\mathbf{x}_{n+i}, \mathbf{x}_j)$. The other entries $K(X,X_*)$ and $K(X_*,X_*)$ are defined analogously.

Hence the predictive distribution is also Gaussian. Namely,

$$Y_*|Y, X, X_* \sim N(\tilde{\mathcal{Y}}, P)\,,$$

with

$$\tilde{\mathcal{Y}} = K(X_*, X)[K(X, X) + \sigma^2\mathbf{I}]^{-1}Y, \tag{C.2}$$

$$P = K(X_*, X_*) - K(X_*, X)[K(X, X) + \sigma^2\mathbf{I}]^{-1}K(X, X_*). \tag{C.3}$$

In view of (C.2) and (C.3), the *covariance/kernel* function k plays a significant role in GPR. One commonly used kernel is the *squared exponential*, i.e.,

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma_f^2 e^{-\frac{1}{2}(\mathbf{x}_i - \mathbf{x}_j)^\top \Lambda (\mathbf{x}_i - \mathbf{x}_j)}, \quad i, i = 1, \ldots, n, \tag{C.4}$$

where $\sigma_f \in \mathbb{R}$ and Λ is a $d \times d$ positive-definite matrix. For more examples of Kernel functions and an in-depth analysis on how choosing this function, we refer to Rasmussen and Williams (2005, Chap. 4). Each kernel has a number of parameters which specify the precise shape of the covariance function. These are referred to as *hyperparameters*, since they can be viewed as specifying a distribution over function parameters, instead of being parameters which specify a function directly. The hyperparameters in the squared exponential kernel are σ_f and the entries of the matrix Λ.

C.1. Computation of the predictive targets

Denote the vector $[K(X, X) + \sigma^2\mathbf{I}]^{-1}Y$ in (C.2) by $\alpha = [\alpha_1, \ldots, \alpha_n]^T$, then we have the estimation for Y_*,

$$Y_{*j} \approx \tilde{\mathcal{Y}}_j = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_{n+i}, \mathbf{x}_j), \quad j = 1, \ldots, n\,. \tag{C.5}$$

If the Gaussian kernel is known, the value Y_* at a point X_* can be estimated by (C.5).

C.2. Training the GPR

Training the GPR means to determine the hyperparameters using the data (X, Y). The training is usually conducted by using ML estimation. Observing that $Y \sim N(\mathbf{0}, K + \sigma^2\mathbf{I})$, we deduce that the log-likelihood is given by

$$\log p(Y|X) = -\frac{1}{2}Y^T(K + \sigma^2\mathbf{I})^{-1}Y - \frac{1}{2}\log|K + \sigma^2\mathbf{I}| - \frac{n}{2}\log 2\pi. \tag{C.6}$$

C.2.1. *Train the GPR on cartesian grid*

The computational complexity for computing the marginal likelihood in Eq. C.6 is dominated by the need to invert the K matrix. Standard methods for matrix inversion of positive definite symmetric matrices require a time of order $O(n^3)$ for inversion of an n by n matrix. If the valuation set is very big, the computational time could be significant. When the covariance function is expressible as a tensor product kernel and the inputs form a multidimensional grid, it was shown in Saatçi (2011) that the costs for the GPR can be reduced (see also Xu *et al.* (2011) and Luo and Duraiswami (2013)). We briefly describe in the sequel the GPR on multidimensional grids.

We assume the kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is written in terms of kernels $k_m : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, for $m = 1, \ldots, d$ as follows:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \Pi_{m=1}^d k_m(x_i^{(m)}, x_j^{(m)}). \tag{C.7}$$

Consider the squared exponential kernel as in (C.4) with a diagonal covariance Λ, i.e., $\Lambda = \mathrm{diag}(\frac{1}{\ell_1^2}, \ldots, \frac{1}{\ell_d^2})$, for $\ell \in \mathbb{R}$. It holds

$$\begin{aligned} k(\mathbf{x}_i, \mathbf{x}_j) &= \sigma_f^2 \exp\left\{ -\sum_{m=1}^d \frac{(x_i^{(m)} - x_j^{(m)})^2}{2\ell_k^2} \right\} \\ &= \Pi_{m=1}^d k_m(x_i^{(m)}, x_j^{(m)}), \end{aligned}$$

where $k_m(x, y) = \sigma_f^{2/d} \exp\{-(x-y)^2/2\ell_m^2\}$.

We consider the data on a d-dimensional Cartesian grid. Let $\{n_s, s = 1, \ldots, d\}$ be the number of points in each dimension in the grid. Then the grid has $n = n_1 \times \cdots \times n_d$ points. Let X be the list of n points in the grid, i.e., $X = (\mathbf{x}_1, \cdots, \mathbf{x}_n)$ and the indexes $n_1, \ldots, n_d$ are such that $1 \le i_1 \le n_1$, $1 \le i_2 \le n_2$, …, $1 \le i_d \le n_d$. The points $\mathbf{x}_j$, $1 \le j \le n$, have the form $\mathbf{x}_j = (x_{j(1)}^{(1)}, x_{j(2)}^{(2)}, \ldots, x_{j(d)}^{(d)})$, with $x_{j(s)}^{(s)}$ being the $j(s)$th element in the dimension s in the grid.

We put the points of the Cartesian grid in the following order:

$$\mathbf{x}_{(i_1-1)\times n_2 \times \cdots \times n_d + \cdots + (i_d-1)\times n_d + i_d} = (x_{i_1}^{(1)}, \ldots, x_{i_d}^{(d)}). \tag{C.8}$$

Consider, for example, a three-dimensional Cartesian grid. Take $n_1 = 3$, $n_2 = n_3 = 2$. Then the grid has 12 points and Eq. (C.8) yields

$$\begin{aligned} \mathbf{x}_1 &= (x_1^{(1)}, x_1^{(2)}, x_1^{(3)}), \\ \mathbf{x}_2 &= (x_1^{(1)}, x_1^{(2)}, x_2^{(3)}), \\ \mathbf{x}_3 &= (x_1^{(1)}, x_2^{(2)}, x_1^{(3)}), \end{aligned}$$

$$\vdots$$

$$\mathbf{x}_{12} = (x_3^{(1)}, x_2^{(2)}, x_2^{(3)}).$$

For the GPR model, the Gaussian kernel on the Cartesian grid is given by $K_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$. Suppose the kernel function k has property (C.7).

Then (C.8) becomes

$$\begin{aligned} K_{(i_1-1)\times n_2\times\cdots\times n_d+\cdots+i_d,\,(j_1-1)\times n_2\times\cdots\times n_d+\cdots+j_d} &= k([x_{i_1}^{(1)}, \ldots, x_{i_d}^{(d)}], [x_{j_1}^{(1)}, \ldots, x_{j_d}^{(d)}]) \\ &= \Pi_{k=1}^{d} k_k(x_{i_k}^{(k)}, x_{j_k}^{(k)}) \\ &= k_1(x_{i_1}^{(1)}, x_{j_1}^{(1)}) \times \cdots \times k_d(x_{i_d}^{(d)}, x_{j_d}^{(d)}) . \end{aligned}$$

Let the matrix kernel $K_k = (k_k(x_i^{(k)}, x_j^{(k)}))_{1\le i,j\le n_k}$ and recall K in (C.1). We obtain

$$K = K(X, X) = \otimes_{k=1}^{d} K_k,$$

where $\otimes$ denotes the Kronecker product. It follows that

$$K^{-1} = \otimes_{k=1}^{d} K_k^{-1} .$$

So instead of inverting a high-dimensional matrix, one needs to invert d smaller matrices, and this reduces the computational time significantly.

References

Albanese, C, G Campolieti, O Chen and A Zavidonov (2003). Credit barrier models, *Risk*, 16(6), 109–113.

Andrieu, C, A Doucet, S Singh and V Tadic (2004). Particle methods for change detection, system identification, and control,*Proceedings of the IEEE*, 92, 423–438, https://doi.org/10.1109/JPROC.2003.823142.

Arulampalam, MS, S Maskell, N Gordon and T Clapp (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,*IEEE Transactions on Signal Processing*, 50(2), 174–188.

Avellaneda, M and J Zhu (2001). Distance to default, *Risk*, 14(12), 125–129.

Bangia, A, FX Diebold, A Kronimus, C Schagen and T Schuermann (2002). Ratings migration and the business cycle, with application to credit portfolio stress testing, *Journal of Banking & Finance*, 26(2–3), 445–474.

Batema, J (2024). Back testing of the Basel capital formula: Insights from 20 years of bank loss data. https://ssrn.com/abstract=4991583.

Bishop, CM (2006). *Pattern Recognition and Machine Learning*, Information Science and Statistics, Springer, New York.

Brockwell, PJ and RA Davis (2006). *Time Series: Theory and Methods*, Springer Series in Statistics, Springer, New York. Reprint of the second (1991) edition.

Cappé, O, SJ Godsill and E Moulines (2007). An overview of existing methods and recent advances in sequential Monte Carlo, *Proceedings of the IEEE*, 95(5), 899–924.

Chopin, N, PE Jacob and O Papaspiliopoulos (2013). SMC^2: An efficient algorithm for sequential analysis of state space models, *Journal of the Royal Statistical Society. Series B (Methodological)*, 75(3), 397–426, https://doi.org/10.1111/j.1467-9868.2012.01046.x.

Chui, CK and G Chen (2017). *Kalman Filtering with Real-time Applications*, 5th edn., Springer, Cham, https://doi.org/10.1007/978-3-319-47612-4.

Credit Suisse First Boston International (1997). CreditRisk+ — a credit risk management framework, London.

Crosbie, P and J Bohn (2003). Modelling default risk. https://www.moodysanalytics.com/-/media/whitepaper/before-2011/12-18-03-modeling-default-risk.pdf.

de Bruijn, NG (1981). *Asymptotic Methods in Analysis*, 3rd edn., Dover Publications, New York.

Demey, P, J Jouanin, C Roget and T Roncalli (2007). Maximum likelihood estimate of default correlations, SSRN 1032590.

Doucet, A (1997). Monte Carlo methods for Bayesian estimation of hidden Markov models. Application to radiation signals, Ph.D. thesis, Université Paris-Sud, Orsay.

Doucet, A and AM Johansen (2011). A tutorial on particle filtering and smoothing: Fifteen years later, in *The Oxford Handbook of Nonlinear Filtering*, pp. 656–704. Oxford University Press, Oxford.

Duffie, D and KJ Singleton (2004). *Credit Risk: Pricing, Measurement, and Management*, Princeton University Press.

Durbin, J and SJ Koopman (2012). *Time Series Analysis by State Space Methods*, Oxford University Press.

Einicke, GA and LB White (1999). Robust extended Kalman filtering, *IEEE Transactions on Signal Processing*, 47(9), 2596–2599.

Frey, R and AJ McNeil (2002). Dependence modelling, model risk and model calibration in models of portfolio credit risk, *Journal of Risk*, 6(2), 1–40.

Gagliardini, P and C Gouriéroux (2005). Stochastic migration models with application to corporate risk, *Journal of Financial Econometrics*, 3(2), 188–226.

Gordon, NJ, DJ Salmond and AFM Smith (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *IEE Proceedings F (Radar and Signal Processing)*, Vol. 140, pp. 107–113, IET.

Gordy, M and E Heitfield (2002). Estimating default correlations from short panels of credit rating performance data, Unpublished Working Paper.

Gorton, GB and P He (2008). Bank credit cycles, *The Review of Economic Studies*, 75(4), 1181–1214.

He, J, A Khedher and P Spreij (2021). A Kalman particle filter for online parameter estimation with applications to affine models, *Statistical Inference for Stochastic Processes*, 24, 353–403.

He, J, A Khedher and P Spreij (2023). A dimension reduction approach for loss valuation in credit risk modeling, *International Journal of Financial Engineering*, https://doi.org/10.1142/S2424786323500585.

Hull, JC and AD White (2001). Valuing credit default swaps II: Modeling default correlations, *The Journal of Derivatives*, 8(3), 12–21.

Jacobs, M (2025). An empirical study of the term structure of asset value correlations for quantifying credit risk. https://ssrn.com/abstract=5385394.

Jarrow, R (1992). Credit risk: Drawing the analogy, *Risk Magazine*, 5(9), 38–41.

Jarrow, R and S Turnbull (1995). Pricing derivatives on financial securities subject to credit risk, *The Journal of Finance*, 50(1), 53–85.

Kantas, N, A Doucet, SS Singh and JM Maciejowski (2009). An overview of sequential Monte Carlo methods for parameter estimation in general state-space models, *IFAC Proceedings Volumes*, 42(10), 774–785, https://doi.org/10.3182/20090706-3-FR-2004.00129.

Koopman, SJ and A Lucas (2005). Business and default cycles for credit risk, *Journal of Applied Econometrics*, 20(2), 311–323.

Koopman, SJ, A Lucas and P Klaassen (2005). Empirical credit cycles and capital buffer formation, *Journal of Banking & Finance*, 29(12), 3159–3179, https://doi.org/10.1016/j.jbankfin.2005.01.003.

Koopman, SJ, A Lucas and A Monteiro (2008). The multi-state latent factor intensity model for credit rating transitions, *Journal of Econometrics*, 142(1), 399–424.

Łapiński, TM (2019). Multivariate Laplace's approximation with estimated error and application to limit theorems, *Journal of Approximation Theory*, 248, 105305.

Luo, Y and R Duraiswami (2013). Fast near-grid Gaussian process regression, *Artificial Intelligence and Statistics*, pp. 424–432. PMLR.

McNeil, AJ and JP Wendin (2007). Bayesian inference for generalized linear mixed models of portfolio credit risk, *Journal of Empirical Finance*, 14(2), 131–149.

Merton, RC (1974). On the pricing of corporate debt: The risk structure of interest rates, *The Journal of Finance*, 29(2), 449–470.

Miller, PD (2006). *Applied Asymptotic Analysis*, Graduate Studies in Mathematics, Vol. 75, American Mathematical Society.

Nickell, P, W Perraudin and S Varotto (2000). Stability of rating transitions, *Journal of Banking & Finance*, 24(1–2), 203–227.

Pitt, M (2002). Smooth particle filters for likelihood evaluation and maximisation, The Warwick Economics Research Paper Series (TWERPS), Department of Economics, University of Warwick.

Pitt, MK and N Shephard (1999). Filtering via simulation: Auxiliary particle filters, *Journal of the American Statistical Association*, 94(446), 590–599, https://doi.org/10.2307/2670179.

Poyiadjis, G, A Doucet and SS Singh (2005). Particle methods for optimal filter derivative: application to parameter estimation, *Proceedings (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005*, Vol. 5, pp. v/925–v/928, doi: 10.1109/ICASSP.2005.1416456.

Press, SJ (2003). *Subjective and Objective Bayesian Statistics: Principles, Models, and Applications*, 2nd edn., Wiley Series in Probability and Statistics, Wiley-Interscience, Hoboken, NJ. With contributions by S Chib, M Clyde, G Woodworth and A Zaslavsky.

Rasmussen, CKI and CE Williams (2005). *Gaussian Processes for Machine Learning*, Adaptive Computation and Machine Learning, The MIT Press.

RiskMetrics-Group (1997). *CreditMetrics — Technical Document*, J.P. Morgan & Company, https://www.msci.com/documents/10199/93396227-d449-4229-9143-24a94dab122f.

Robert, CP (2007). *The Bayesian Choice: From Decision-theoretic Foundations to Computational Implementation*, 2nd edn., Springer Texts in Statistics, Springer, New York.

Saatçi, Y (2011). Scalable inference for structured Gaussian process models, Ph.D. thesis, Citeseer.

Schölkopf, B and AJ Smola (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, The MIT Press.

Seeger, M (2004). Gaussian processes for machine learning, *International Journal of Neural Systems*, 14(2), 69–106.

Simons, D and F Rolwes (2009). Macroeconomic default modeling and stress testing, *International Journal of Central Banking*, 5(3), 177–204.

Steinwart, I (2001). On the influence of the kernel on the consistency of support vector machines, *Journal of Machine Learning Research*, 2, 67–93.

Tierney, L and JB Kadane (1986). Accurate approximations for posterior moments and marginal densities, *Journal of the American Statistical Association*, 81(393), 82–86.

van Schuppen, JH (1989). Stochastic realization problems, in *Stochastic Realization Problems*, Lecture Notes in Control and Information Sciences, Vol. 135, pp. 480–523, Springer, Berlin, https://doi.org/10.1007/BFb0008474.

Vasicek, O (2002). The distribution of loan portfolio value, *Risk*, 15(12), 160–162.

Wan, EA and AT Nelson (2002). *Dual Extended Kalman Filter Methods*, Chap. 5, pp. 123–173, John Wiley & Sons, https://doi.org/10.1002/0471221546.ch5.

Wan, EA and R van der Merwe (2002). *The Unscented Kalman Filter*, Chap. 7, pp. 221–280, John Wiley & Sons, https://doi.org/10.1002/0471221546.ch7.

Wendin, J and AJ McNeil (2006). Dependent credit migrations, *Journal of Credit Risk*, 2(3), 87–114.

Wills, A, T Schön and B Ninness (2008). Parameter estimation for discrete-time nonlinear systems using EM, *IFAC Proceedings Volumes*, 41(2), 4012–4017, https://doi.org/10.3182/20080706-5-KR-1001.00675.

Xu, ZN, F Yan and Y Qi (2011). Infinite Tucker decomposition: nonparametric Bayesian models for multiway data analysis, preprint, arXiv:1108.6296.