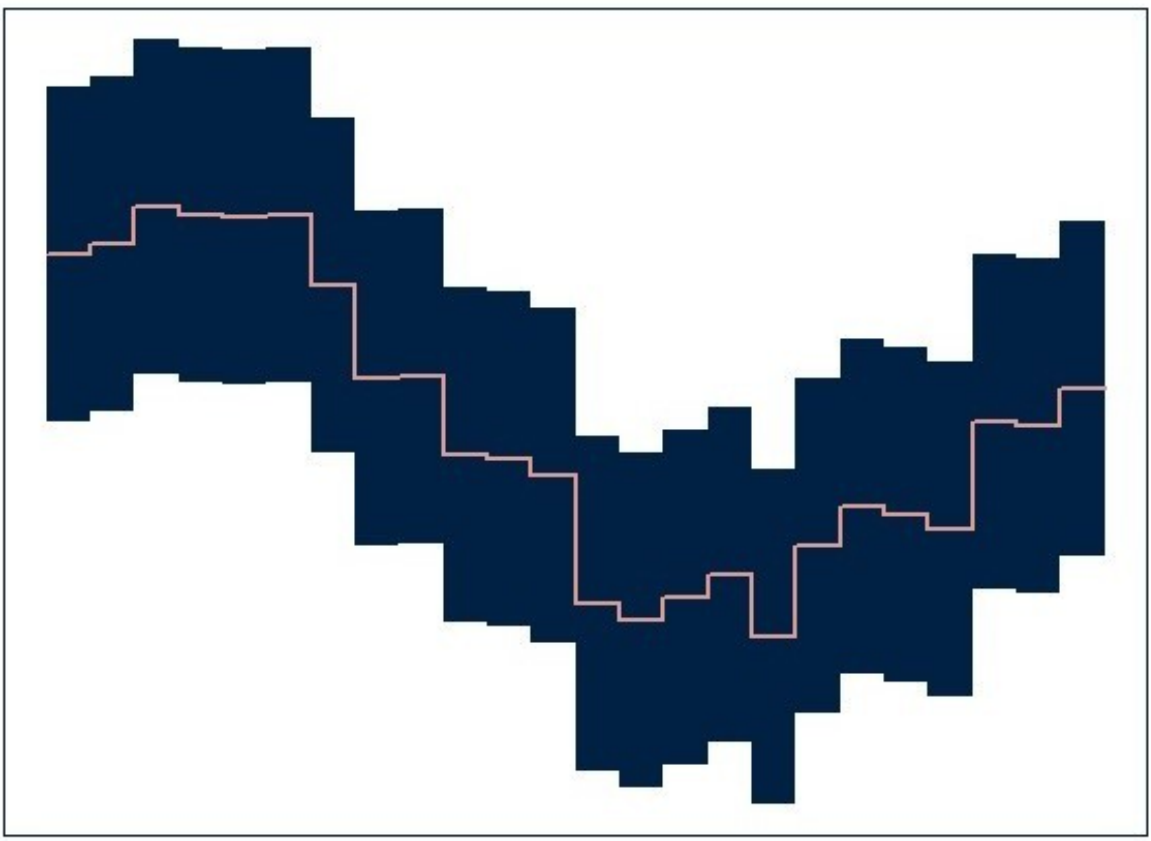
Stockholm University

Bayesian Sequential Inference for Dynamic Regression Models

Parfait Munezero



Doctoral Thesis in Statistics at Stockholm University, Sweden 2020

Bayesian Sequential Inference for Dynamic Regression Models

Parfait Munezero

Academic dissertation for the Degree of Doctor of Philosophy in Statistics at Stockholm University to be publicly defended on Friday 11 December 2020 at 10.00 in hörsal 6, hus C, Universitetsvägen 10 C, and digitally via Zoom. A link will be published at https://www.statistics.su.se/.

Abstract

Many processes evolve over time and statistical models need to be adaptive to change. This thesis proposes flexible models and statistical methods for inference about a data generating process that varies over time. The models considered are quite general dynamic predictive models with parameters linked to a set of covariates via link functions. The dynamics can arise from time-varying regression coefficients and from changes in the link function over time. The covariates can be time-varying and may also have incomplete information.

An efficient Bayesian inference methodology is developed for analyzing the posterior of dynamic regression models sequentially, with a particular focus on online learning and real-time prediction. The core inferential algorithm belongs to a family of sequential Monte Carlo methods commonly known as particle filters, and a key contribution is the development of a tailored proposal distribution. The algorithm is shown to outperform a state-of-the-art Markov Chain Monte Carlo method and is also extended to mixture-of-experts models.

The performance of the inference methodology is assessed through various simulation experiments and real data from clinical and social-demographic studies, as well as from an industrial software development project.

Keywords: *Bayesian sequential inference, Dynamic regression models, Particle filter, Online prediction, Particle smoothing, Linear Bayes.*

Stockholm 2020
http://urn.kb.se/resolve?urn=urn:nbn:se:su:diva-186121

ISBN 978-91-7911-336-0
ISBN 978-91-7911-337-7

Department of Statistics

Stockholm University, 106 91 Stockholm

BAYESIAN SEQUENTIAL INFERENCE FOR DYNAMIC REGRESSION MODELS

Parfait Munezero

Stockholm
University

Bayesian Sequential Inference for Dynamic Regression Models

Parfait Munezero

ISBN print 978-91-7911-336-0
ISBN PDF 978-91-7911-337-7

Printed in Sweden by Universitetsservice US-AB, Stockholm 2020

To Grace and Paris

Abstract

Many processes evolve over time and statistical models need to be adaptive to change. This thesis proposes flexible models and statistical methods for inference about a data generating process that varies over time. The models considered are quite general dynamic predictive models with parameters linked to a set of covariates via link functions. The dynamics can arise from time-varying regression coefficients and from changes in the link function over time. The covariates can be time-varying and may also have incomplete information.

An efficient Bayesian inference methodology is developed for analyzing the posterior of dynamic regression models sequentially, with a particular focus on online learning and real-time prediction. The core inferential algorithm belongs to a family of sequential Monte Carlo methods commonly known as particle filters, and a key contribution is the development of a tailored proposal distribution. The algorithm is shown to outperform a state-of-the-art Markov Chain Monte Carlo method and is also extended to mixture-of-experts models.

The performance of the inference methodology is assessed through various simulation experiments and real data from clinical and social-demographic studies, as well as from an industrial software development project.

Keywords: *Bayesian sequential inference, Dynamic regression models, Particle filter, Online prediction, Particle smoothing, Linear Bayes.*

Acknowledgments

I take this opportunity to express my gratitude and appreciation to everyone who supported me in any ways during this PhD journey.

I am grateful to my supervisor, Mattias Villani. Thank you for the support and guidance you provided me throughout this journey. I highly value your advises, the time you put into my work, and the great research discussions. I thank you from the bottom of my heart for encouraging and being patient with me especially in the last time.

I would like to express my heart-felt gratitude to my co-supervisor Gebrenegus Ghilagaber. I am grateful for the assistance you provided to make the collaboration with Ericsson possible. This collaboration turned out to be very fruitful to the thesis as well as to my private life. Thanks for collaborating with me, taking time to read and edit my papers despite your heavy duty, and most importantly thanks for the "injera". I would like also to thank the staff of statistics department at Stockholm University. It was an honor to work with you.

I am grateful to Data Insights support team which provided some of the data used in the thesis. I thank Håkan Abrander and Paul Stewart, in particular, for being extremely supportive and for providing me all facilities I needed to write my thesis in comfortable conditions. Thank you Vasilis Manikas for connecting me with Data Insights support team.

This PhD journey would not have been possible without the support of my lovely family. Grace, my wife, there is no value I can attribute to the efforts you made especially during the last time. There is no way I could finish this PhD without you. All I can say now is that: "we made it, my love!". I thank my parents for being my source of motivation since day one, always encouraging and standing by my side. Dad and Mum, you made my dream possible.

Parfait Munezero

Contents

Bibliography **53**

1. Introduction

1.1. Background

Predictive models allow us to describe the uncertainty about unknown events and, therefore, they play an important role in decision making. For instance, in our daily life, we often check the weather forecast in order to plan our day. If the forecast indicates rain, we may decide to carry an umbrella or drive to work instead of taking public transport.

Such prediction-dependent decisions are also essential in many business or application areas. This thesis contains several examples in clinical, social-demographic, and industrial software development applications where predictions and decisions are essential. In clinical applications, modeling the effect of medical treatments for a disease can help to understand the likely course of the medical condition of patients and to prescribe personalized treatments. In industrial applications, predictive models can be used to describe the reliability of a product, which can in turn support decisions related to the production process. For example, scheduling the release time of a product.

The most commonly used predictive models are statistical models which express the probability distribution of a response variable as function of a set of explanatory input variables. The explanatory variables - also known as covariates, regressors, or features - document the background information essential for describing the response variable. The models considered here are *dynamic regression* models in which the predictive density function of the response variable changes over time. The dynamics can arise from the following sources:

- *Time-varying covariate*s: Covariates may be measured at different time points or naturally change over time. An example is educational level, commonly used in social-demographic studies to explain the life course of some events such as marriage. In Section 4.2, we describe models that incorporate a dynamic covariate and propose a methodology for adjusting incomplete information in the covariate.
- *Time-varying parameter*s: The models developed in the thesis are dynamic regression models with parameters linked to covariates via link functions. Both the regression coefficients and the form of the link may change smoothly over time. One example provided in Section 4.1 is a clinical application where

the effect of some treatment of Gastric cancer improves over time. See also additional examples in the papers presented in the thesis.

- *Time-varying functional form of the response's probability distribution*: A typical example is the application on predicting the number of software faults in a continuously upgraded software project. The distribution of software faults changes because the developers in the project, the user behavior and technologies change over time. More details are provided in Section 4.3.

The thesis proposes a Bayesian methodology for inference in dynamic predictive models. The Bayesian paradigm treats the parameter as a random quantity and describes the uncertainty about the parameter before any data is seen through the *prior* distribution. This provides the flexibility to incorporate relevant expert knowledge in the analysis, information collected from other data sources and subjective beliefs about the parameter. After observing evidence from the data, the prior distribution is updated into a *posterior* distribution on which the predictive inference is based; see Section 1.4.

The Bayesian inference methodology allows naturally for constructing the posterior recursively through time where "yesterday's posterior becomes today's prior". This sequential inference has some advantages. It simplifies *online* (real time) predictions. Online predictive models are constructed recursively through time in a way that does not require a complete scan of the entire data set. This enables us to make predictions as new data arrive by simple updating; a common situation in applications with massive streaming data. It can also be useful in other situations where sequential analysis is not necessarily adopted for online predictions, but it is rather used for computational convenience. An example is the class of models for analyzing survival data, where the response is the survival time – a time period between some starting point and the occurrence of an event of interest; see Section 4.1 and Paper I for details. Other examples include the models with static parameters but analyzed sequentially by annealing the likelihood to gain computational efficiency (Chopin, 2002).

1.2. Objectives and main contributions

The main objective of the thesis is to develop a Bayesian framework for efficient online predictions, more specifically to: (i) extend some flexible predictive models to the dynamic setting, and (ii) develop fast and efficient inference methodology for such models. In the following, we detail each separately; emphasizing on the contributions proposed in the thesis.

Dynamic link functions. The link functions connecting the covariates to the parameter in the model are predefined by the researcher, typically without any reference to the data. Paper IV extends the family of parametric link functions in

Czado 1997 to class of flexible data-driven link functions that can change through time.

Dynamic mixture of experts. Mixture of experts are flexible predictive models for modeling the response variable. They extend the usual finite mixture models to allow for covariate-dependent component models and mixture weights (Jordan and Jacobs, 1994). The component models can be any density belonging to the exponential family; see Section 2.2 for a definition. This sets up a wide spectrum of applications since mixture models have the flexibility to capture complex distributional shapes of the data and the exponential family includes density functions for both discrete and continuous response variables. The standard setting of mixture of experts models use static regression models in the component and mixture weights models, see for instance Villani et al. (2012). Paper III extends this class of models to the dynamic setting by allowing both the parameters in the mixture components and mixture weights to vary over time. Furthermore, it provides an inference methodology that accommodates density functions of the response variable outside the exponential family.

Fast and efficient online inference methodology. The state-of-the-art methods for doing Bayesian inference use Markov Chain Monte Carlo (MCMC) sampling algorithms. These algorithms have their own merits, but they are generally computationally expensive. Furthermore, they are not suitable for online predictions. To provide fast and efficient online inference, we use Sequential Monte Carlo (SMC) methods commonly known as *particle filters* (Gordon et al., 1993). SMC methods were initially developed for filtering problems encountered in engineering applications, where the main task is to extract information from noisy and partially observed data. The thesis focuses on adapting these algorithms to general dynamic predictive models to make fast and efficient inference using a new proposal density tailored to the sequential posterior distribution. Paper I, III and IV contain the details.

1.3. Outline of the thesis

The thesis is organized in two parts. The first part introduces relevant concepts used in the thesis. Chapter 2 describes the dynamic regression models considered in the thesis. Further, a concise introduction to sequential Bayesian analysis for dynamic models is provided towards the end of the chapter. Chapter 3 introduces Bayesian inference methods in dynamic models. Chapter 4 presents an overview of some applications of the dynamic predictive models considered in the thesis. The first part is then concluded with a summary of the research papers. The second part provides all research papers in the thesis.

1.4. Introduction to Bayesian learning

This section introduces the key concepts for Bayesian learning such as the prior distribution, likelihood function and posterior distribution in order to set up some foundation for models discussed in subsequent chapters.

We initially disregard the information from covariates for reasons of simplicity; models with covariates are considered in Chapter 2. We assume that we have data consisting of n observations $y_1, \ldots, y_n$ on the response variable; for instance the survival times of n individuals in the survival analysis example, in Section 4.1, or the number of faults in a given software release in the software development application in Section 4.3.

The observed data are random since different data may be observed if we followed a different sample of individuals in our study. We therefore model the data as realizations from some unknown data generating process (DGP). Throughout this section, we provide illustrations using the Poisson distribution because of its simplicity and its important role as basis for more flexible models in the thesis.

1.4.1. Bayesian parameter updating and prediction

The observed data set is used to model the DGP by constructing a conditional probabilistic model $f(y_1, \ldots, y_n|\lambda)$. This model describes the uncertainty about the observations given an unknown parameter λ, for example, the mean or variance of the distribution. Such a model necessitates some form of assumptions on the dependence of the observations. The classical assumption requires the observations to be mutually independent and identically distributed (IID) conditional on λ:

$$f\left(y_1, \ldots, y_n|\lambda\right) = \prod_{i=1}^{n} f\left(y_i|\lambda\right), \tag{1.1}$$

where $f\left(y|\lambda\right)$ is any parametric density function parameterized by λ. When viewed as function of the parameter, for a fixed sample, the expression (1.1) is the *likelihood* function. Example 1.1 provides an illustration of the likelihood function for Poisson distributed data.

Example 1.1. Poisson distributed data.

Assuming that the data y_i , $i = 1, \ldots, n$ are IID Poisson distributed with mean λ, the likelihood function is

$$f\left(y_1, \ldots, y_n|\lambda\right) = \prod_{i=1}^{n} \frac{\lambda^{y_i} \exp\left(-\lambda\right)}{y_i!} = \frac{\lambda^{\sum_{i=1}^{n} y_i} \exp\left(-n\lambda\right)}{\prod_{i=1}^{n} y_i!}. \tag{1.2}$$

The likelihood function plays a important role in statistical inference. We can estimate the unknown parameter by the maximum likelihood estimator (MLE)

$$\hat{\lambda}_{MLE} = \arg\max_{\lambda} \log f\left(y_1, \ldots, y_n|\lambda\right). \tag{1.3}$$

The MLE is used, in the frequentist approach, for inference where focus is on the repeated sampling variability of the MLE. Given $\hat{\lambda}_{MLE}$ a common frequentist solution to the prediction problem is the predictive density for future outcomes $y_{n+1}, \ldots, y_{n+m}$ conditional on $\hat{\lambda}_{MLE}$:

$$f_{\text{MLE}}\left(y_{n+1}, \ldots, y_{n+m} | y_1, \ldots, y_n\right) = f\left(y_{n+1}, \ldots, y_{n+m} | \hat{\lambda}_{MLE}\right), \tag{1.4}$$

under the IID assumption.

On the other hand, the Bayesian paradigm treats the parameter λ as a random quantity. Different analyses may involve similar subjects which differ only in their contextual interpretation. For instance, we may have two data sets of count data: One consisting of the number of accidents on a highway, and the other consisting of the number of injuries that occurred during different football matches. The Bayesian approach gives us the flexibility to treat the two analyses differently by incorporating our belief and knowledge about the subject under study. Our uncertainty about the parameter before seeing evidence from the data is described through the *prior* $p(\lambda)$. After observing evidence from the data, Bayes theorem,

$$p\left(\lambda | y_1, \ldots, y_n\right) = \frac{f\left(y_1, \ldots, y_n | \lambda\right) p(\lambda)}{\int_{\Omega} f\left(y_1, \ldots, y_n | \lambda\right) p(\lambda) d\lambda}, \tag{1.5}$$

is used to update the prior information we had about the parameter to a *posterior* density $p(\lambda | y_1, \ldots, y_n)$. Here Ω is the parameter space for λ, and the integral in the denominator of Eq. 1.5 is a constant of normalization well known as the *marginal likelihood* of the data. This constant is usually omitted to obtain a simpler expression of the unnormalized posterior, ´

$$p\left(\lambda | y_1, \ldots, y_n\right) \propto f\left(y_1, \ldots, y_n | \lambda\right) p(\lambda). \tag{1.6}$$

The posterior is a compromise between the data and the prior. To put this clearly, we analyze the posterior of IID Poisson distributed data in Example 1.2 and illustrate it graphically in Figure 1.1.

Example 1.2. Posterior for Poisson distributed data

Since the parameter λ for the Poisson density is defined on positive real numbers, any density that has this support can be used as the prior distribution. We assume a Gamma prior $\lambda \sim \Gamma(\alpha, \beta)$, which, using the Bayes theorem and the likelihood in Example 1.1, leads to the posterior distribution

$$\lambda | y_1, \ldots, y_n \sim \Gamma\left(\alpha + \sum_{i=1}^{n} y_i,\ \beta + n\right), \tag{1.7}$$

with mean

$$E\left(\lambda | y_1, \ldots, y_n\right) = \frac{\alpha + \sum_{i=1}^{n} y_i}{\beta + n} = \frac{\beta}{\beta + n}\left(\frac{\alpha}{\beta}\right) + \frac{n}{\beta + n}\left(\frac{\sum_{i=1}^{n} y_i}{n}\right), \tag{1.8}$$

and variance

$$V\left(\lambda|y_1,\ldots,y_n\right) = \frac{\alpha + \sum_{i=1}^{n} y_i}{(\beta + n)^2}. \tag{1.9}$$

The posterior mean is a weighted sum of the prior mean and the observations mean. The latter outweighs the prior mean as the sample size increases; see the next section. Further, the variance of the posterior is on average smaller than the variance of the prior as can be seen from the law of total variance:

$$V(\lambda) = E\left(V\left(\lambda|y_1,\ldots,y_n\right)\right) + V\left(E\left(\lambda|y_1,\ldots,y_n\right)\right),$$

where the outer expectations are taken with respect to the density of the data.

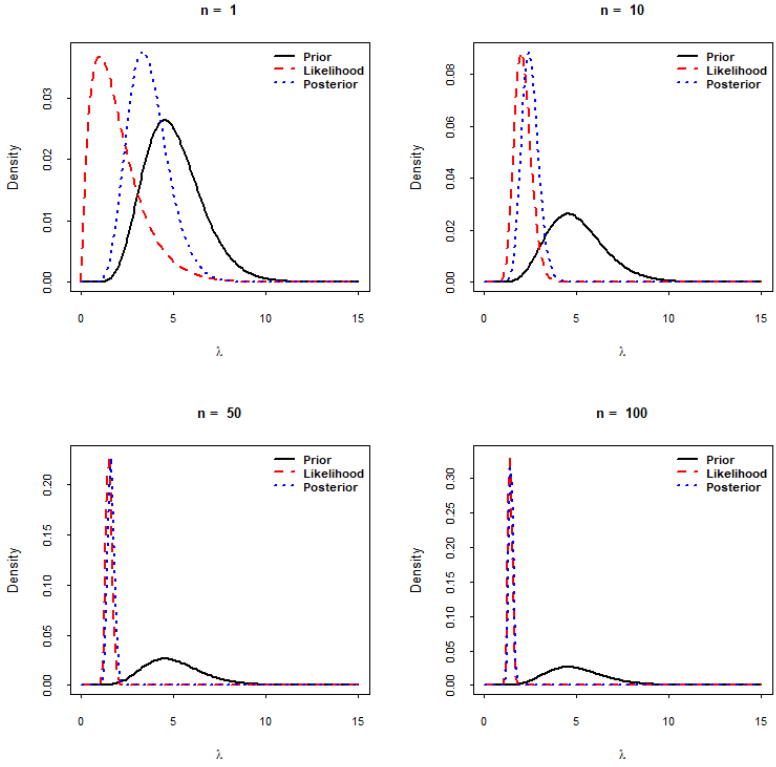


Figure 1.1.: Posterior distribution for IID Poisson data. Each panel displays the analysis with n observations simulated from the Poisson distribution with $\lambda = 1.5$. The prior distribution is set to the Gamma distribution with $\alpha = 10$ and $\beta = 2$, so that the prior mean is 5 and the prior variance is 2.5. The likelihood function displayed in all panels is normalized.

Inference is based on the posterior density in the Bayesian paradigm. The *posterior predictive density* for future outcomes $y_{n+1}, \ldots, y_{n+m}$ is given directly by

$$f\left(y_{n+1}, \ldots, y_{n+m} | y_1, \ldots, y_n\right) = \int_\Omega f\left(y_{n+1}, \ldots, y_{n+m} | \lambda\right) p\left(\lambda | y_1, \ldots, y_n\right) d\lambda, \tag{1.10}$$

which establishes a learning process that use information learned from the observed data to describe the uncertainty in the prediction of future outcomes, including uncertainty about λ.

1.4.2. Asymptotic posterior normality

From Figure 1.1 it can be seen that the posterior gets closer to the likelihood and the uncertainty in the parameter reduces as we observe more data. This result is implied from the asymptotic properties of the posterior summarized in the following theorem.

Theorem 1.1. ***(Asymptotic normality)***. *Assume that $y_1, \ldots, y_n$ is a sample of n IID observations with likelihood $f(y_1, \ldots, y_n|\lambda)$ and a prior density $p(\lambda)$. Under some regularity conditions, the posterior distribution converges to the normal distribution with mean $\hat{\lambda}_{MLE}$ and variance*

$$I\left(\hat{\lambda}_{MLE}\right)^{-1} = -\left[\frac{d^2 \log p\left(\lambda | y_1, \ldots, y_n\right)}{d\lambda^2}\bigg|_{\lambda=\hat{\lambda}_{MLE}}\right]^{-1}, \tag{1.11}$$

as $n \to \infty$.

In practice, Bayesian inference does not typically rely on asymptotic theory, since exact posterior inference can be obtained even with finite sample sizes; using for example simulation algorithms like MCMC and SMC as in this thesis. The normal approximation,

$$N\left(\hat{\lambda},\ I\left(\hat{\lambda}\right)^{-1}\right), \tag{1.12}$$

of the posterior distribution is often fairly accurate also for small n. Here $\hat{\lambda}$ is often the posterior mode instead of the MLE. We use it in the thesis to design efficient proposal distributions for MCMC and SMC algorithms. The following example illustrates the normal approximation of the posterior for Poisson distributed data in Example 1.2. This example is merely for illustration since here we actually know the posterior distribution and there is no need for a normal approximation.

Example 1.3. Normal approximation of the posterior.

We have established the posterior distribution for Poisson distributed data as the Gamma distribution $\Gamma(\alpha+\sum_{i=1}^{n} y_i,\ \beta+n)$. Therefore, to construct the approximation we need the first and second derivatives of

$$\log p\left(\lambda|y_1,\ldots,y_n\right) = \left(\alpha - 1 + \sum_{i=1}^{n} y_i\right)\log\lambda - \left(\beta + n\right)\lambda,$$

which can be obtained by direct calculations as

$$\frac{d\log p\left(\lambda|y_1,\ldots,y_n\right)}{d\lambda} = \frac{\alpha - 1 + \sum_{i=1}^{n} y_i}{\lambda} - \left(\beta + n\right),$$

$$\frac{d^2\log p\left(\lambda|y_1,\ldots,y_n\right)}{d\lambda^2} = -\frac{\alpha - 1 + \sum_{i=1}^{n} y_i}{\lambda^2}.$$

The normal approximation is therefore

$$p\left(\lambda|y_1,\ldots,y_n\right) \approx N\left(\frac{\alpha - 1 + \sum_{i=1}^{n} y_i}{\beta + n},\ \frac{\alpha - 1 + \sum_{i=1}^{n} y_i}{\left(\beta + n\right)^2}\right). \tag{1.13}$$

Figure 1.2 displays the approximation (1.13) with $n = 1$ and $n = 100$ respectively. Note the slight difference in the approximation's accuracy when $n = 1$ compared to $n = 100$.

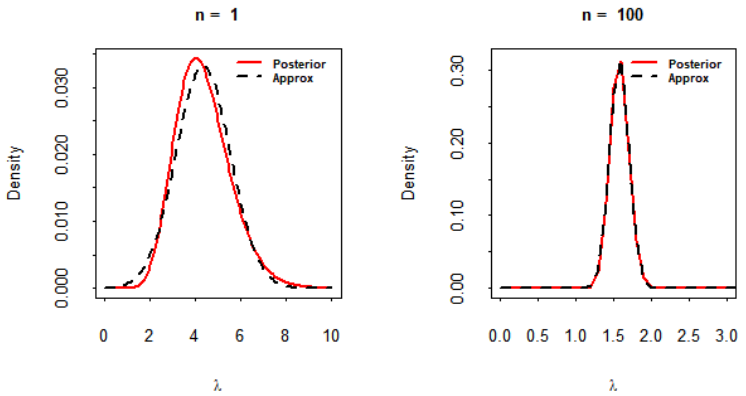


Figure 1.2.: Normal approximation of the posterior distribution. Each panel displays the posterior distribution of λ and its approximation given n IID observations simulated from the Poisson distribution with $\lambda = 1.5$. The prior distribution is set to the Gamma distribution with $\alpha = 10$ and $\beta = 2$.

1.4.3. Summarizing posterior distributions

One key advantage of using Bayesian methods in model with multiple parameters is that the marginal posterior density is obtained by direct integration:

$$p\left(\lambda_k|y_1,\ldots,y_n\right) = \int_{\Omega_{(-k)}} p\left(\lambda_k,\lambda_{(-k)}|y_1,\ldots,y_n\right) d\lambda_{(-k)}$$

where $\lambda_{(-k)}$ denotes all parameters excluding λ_k and $\Omega_{(-k)}$ is the parameter space for $\lambda_{(-k)}$. The marginal posterior can easily be estimated using posterior samples from an MCMC or SMC procedure (Chen et al., 2012, Chapter 4).

It is often practical to summarize the posterior by some low-dimensional summaries, for example single valued point estimates. The point estimator is a value that minimizes the posterior expected loss

$$E\left[L\left(\hat{\lambda},\lambda\right)|y_1,\ldots,y_n\right] = \int L\left(\hat{\lambda},\lambda\right) p\left(\lambda|y_1,\ldots,y_n\right) d\lambda, \tag{1.14}$$

with respect to some loss function $L(\hat{\lambda},\lambda)$ that penalizes deviations from the true value of λ. The posterior mode, often referred to as the maximum a posteriori (MAP) estimator is

$$\hat{\lambda}_{\text{MAP}} = \arg\max_{\lambda} \log p\left(\lambda|y_1,\ldots,y_n\right). \tag{1.15}$$

Notice that if the prior is a uniform distribution, then the MAP estimator coincides with the MLE estimator in Eq. 1.3. The estimator $\hat{\lambda}_{\text{MAP}}$ is optimal under the zero-one loss function $L(\hat{\lambda},\lambda) = I_{\lambda}\left(\hat{\lambda}\right)$, which is 1 if $\lambda = \hat{\lambda}$ and 0 otherwise. The posterior mean and median minimize Eq. 1.14 under square error loss functions $L(\hat{\lambda},\lambda) = (\hat{\lambda}-\lambda)^2$ and absolute value loss functions $L(\hat{\lambda},\lambda) = |\hat{\lambda}-\lambda|$, respectively.

One issue with reporting a point estimate as a posterior summary is that the uncertainty about the parameter is not represented. The uncertainty about the parameter can be communicated via *credible intervals*. These are probability intervals which may be computed with respect to the posterior or the predictive distribution. A $100(1-\alpha)\%$ posterior credible interval is the interval $I = (\lambda_l,\lambda_u)$ such that

$$\int_{\lambda_l}^{\lambda_u} p\left(\lambda|y_1,\ldots,y_n\right) d\lambda = 1-\alpha,$$

for any credibility $0 < \alpha < 1$. A commonly used interval is the highest probability density (HPD) interval which includes the value of λ with highest posterior density.

1.4.4. Prior elicitation

Practical *prior elicitation* typically consists selecting a suitable prior density function from a family of density functions $p(\lambda|\kappa)$ parameterized by the so-called *hyperparameter* κ. The selection of the parametric density function family depends on the

support of the parameter space and whether the parameter is discrete or continuous; the Gaussian density is a good candidate for parameters defined on the real line, and the Gamma distribution is appropriate for parameters defined on the positive real line as in the Poisson case in Example 1.2.

One important class of prior distributions widely applied in the literature is the class of *conjugate* priors; a prior is conjugate to the likelihood if the prior and the posterior belong to the same probability distribution family. An example is the Gamma prior in Example 1.2 which is conjugate to the Poisson model since the posterior is also a Gamma distribution; see Lesaffre and Lawson (2012, Chapter 5) for additional examples. Conjugate priors are computationally convenient as they allow us to compute the posterior in closed form. Prior knowledge can be easily incorporated by matching the theoretical moments of the prior density with the moments elicited from the prior knowledge and solving for the unknown hyperparameter. Such priors constructed based on some domain expert knowledge are known as *informative* priors.

Alternatively, informative priors can be constructed using some historic data collected in previous studies similar to the undergoing study. Let $y_1, \ldots, y_n$ denote historic data and $y_{n+1}, \ldots, y_{n+m}$, the current data. We can use the previous posterior $p(\lambda|y_1, \ldots, y_n)$ as the prior in the current study and obtain the updated posterior

$$p(\lambda|y_1, \ldots, y_n, y_{n+1}, \ldots y_{n+m}) \propto f(y_{n+1}, \ldots, y_{n+m}|\lambda)p(\lambda|y_1, \ldots, y_n),$$

again assuming IID data.

Historical data, or data from other studies, may not be as informative as new data. A more pragmatic way to implement historic data-dependent priors is the power prior (Ibrahim et al., 2000),

$$p(\lambda|y_1, \ldots, y_n, a) \propto f(y_1, \ldots, y_n|\lambda)^a p_0(\lambda|\kappa_0),$$

where $p_0(\lambda|\kappa_0)$ is an initial prior with a predefined κ_0 and $0 < a < 1$ is a parameter that controls the amount of information from the historic data transferred into the prior. The value $a = 1$ corresponds to the usual prior-posterior update given the historic data, while smaller a devalues past data. In this thesis, we use a similar approach to the power prior known as the discount factor approach, which is discussed in Section 3.6.

It is not always possible to use informative priors due to lack of expert knowledge, historic data or time and effort for proper prior elicitation. A solution is then to use *non informative* priors , perhaps a uniform distribution that allocates equal probability mass over the parameter space. Another way is to use conjugate priors with hyperparameter such that the prior distribution carries very little information; for example setting $\alpha = \beta = 0.001$ in the Gamma prior for Poisson data.

2. Dynamic regression models

Regression is the cornerstone of this thesis, we therefore devote this chapter to different regression models to introduce the models developed in the thesis. Regression models relate the response variable Y to a set of covariates $X_1, \ldots, X_P$. This allows us to explain the variation in the response variable that may be attributed to variations in the covariates and to make prediction of the response given some new information on the covariates.

The linear Gaussian regression model is the most popular regression model in the statistical community. Nelder and Wedderburn (1972) developed the class of Generalized linear models (GLM), a powerful generalization of the linear Gaussian regression model. This class of models uses the exponential family, described in Section 2.2, to model the response variable and it allows the covariates to enter the model nonlinearly through an appropriate link function. This sets up a wide spectrum of applications for regression models, since the exponential family includes density functions for both discrete and continuous response variables.

However, there are situations in which GLMs are limited. A situation of particular interest here is when the regression coefficients vary with respect to some other variables, the *effect modifiers* (Hastie and Tibshirani, 1993). The effect modifiers can for example be age, geographic coordinates, time, etc. Here, we consider a particular case in which the only effect modifier is time. More specifically when time plays an important role in describing the evolution of the data generating process under study. This may occur when data are collected continually over a period of time long enough to allow the possibility that the parameters in the model have changed at some points in time. The studied process may also evolve dynamically for other reasons.

West et al. (1985) proposed a class of dynamic generalized linear models (DGLM) extending GLM to allow for time-varying regression coefficients. In this chapter we delineate DGLM and describe some extensions developed in this context. Before diving into DGLM, we briefly introduce in Section 2.1 the linear Gaussian regression model and discuss its assumptions and limitations. We proceed, in Section 2.3, with a brief introduction to GLM, while in Section 2.4 we introduce the class of DGLM. We then present extensions to DGLM proposed in this thesis; first, the dynamic link functions in Section 2.5 followed by the dynamic mixture of experts models in Section 2.6.

2.1. The basic regression model

Assume that we have data consisting of n observations $\{y_i, x_{1i}, \ldots, x_{Pi}\}$, $i = 1, \ldots, n$. The linear Gaussian regression model is of the form

$$y_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_P x_{iP} + \varepsilon_i,\ \varepsilon_i \overset{iid}{\sim} N\left(0,\, \sigma^2\right), \tag{2.1}$$

where β_k, $k = 1, \ldots, P$ are unknown regression coefficients quantifying the effect of the covariates on the response variable, β_0 is an intercept term and ε_i is a random error term, with constant variance σ^2. The linear regression model (2.1) can be expressed in matrix form as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \tag{2.2}$$

where $\mathbf{y} = (y_1, \ldots, y_n)^\top$, $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_P)^\top$, $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n)^\top$ and

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & \ldots & x_{1P} \\ 1 & x_{21} & \ldots & x_{2P} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \ldots & x_{nP} \end{bmatrix}.$$

Three key assumptions are inherent in Eq. 2.1. The first assumption is that the observations are conditionally independent given the regression coefficients, which helps to formulate the likelihood function as explained in Section 1.4. The second assumption, the linearity assumption, stipulates that the expected value of the response variable is linearly related to the covariates; i.e,

$$E[y_i|\mathbf{x}_i] = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{iP}. \tag{2.3}$$

The last assumption is that the distribution of each y_i is a normal distribution with mean $\mu_i = E[y_i|\mathbf{x}_i]$ and variance σ^2 which is constant for all observations.

Though linear models can be useful in many applications, it is not always possible to satisfy the last two assumptions. Appropriate transformations, such as the Box-Cox transformation, can sometimes be used to approximately satisfy the normality assumption. For instance, if Y is strictly positive, one can apply the log transformation. However, in many situations, we are prompted to consider:

1. Other distributional forms of the response variable than the normal distribution. The response may be a discrete count variable that takes only integer values or categories or some classes in classification problems. In these cases, the response variable is clearly not normal and there is no way of achieving it via transformation.
2. A nonlinear relationships between the mean of the response and the covariates. Some types of random variables do not allow negative values in their domain; in this case a different relationship other than the linear form in Eq. 2.3 should be considered.

2.2. Exponential family of distributions

Generalized linear models (GLM) assume that the distributional form of the response variable belongs to the exponential family. This section briefly introduces the exponential family and provides some examples.

Definition 2.1. k**-parameter exponential family**.

The distribution of a random variable Y (discrete or continuous) belongs to the k-parameter exponential family if it can be expressed in the form:

$$f\left(y|\boldsymbol{\lambda}\right) = \exp\left\{\left[\sum_{h=1}^{k} b_h\left(\boldsymbol{\lambda}\right) T_h(y)\right] - A\left(\boldsymbol{\lambda}\right) + a(y)\right\}, \tag{2.4}$$

where $\boldsymbol{\lambda}$ is the parameter of the family; it can be a scalar or a vector representing the mean, variance or any other quantity characterizing the distribution, $T_h(y)$, $h = 1, \ldots, k$, are some functions of the data summarizing available information about the parameter (sufficient statistics), $b_h(\boldsymbol{\lambda})$ are transformations of the parameter, $A(y)$ and $a(y)$ are some known functions. The function $A(\boldsymbol{\lambda})$ is related to the constant of normalization that makes the density integrate to 1 (or sum to 1 if Y is discrete).

The exponential family contains many well known continuous and discrete distributions including the normal distribution. Here, we provide some examples relevant to this thesis, additional examples may be found in Sundberg (2019, chapter 2).

Example 2.1. The Poisson distribution for count data.

The *Poisson distribution* is a one-parameter discrete probability distribution used to model count data; the counts represent the frequency of a certain event of interest in a given time frame or at some location in the space. The parameter λ, represents the rate of occurrence of the event. An example of the Poisson distribution application is provided in Paper III, where the aim is to model the number of faults in a given software upgrade. The probability mass function for a Poisson distributed random variable has the form:

$$f\left(y|\lambda\right) = \frac{\lambda^y \exp\left(-\lambda\right)}{y!} = \exp\left(y \log \lambda - \lambda - \log y!\right), \tag{2.5}$$

for $y = 0, 1, 2, \ldots$ and $\lambda > 0$. This can be written in the exponential family form (2.5) by defining $b\left(\lambda\right) = \log \lambda$, $T(y) = y$, $a(y) = -\log y!$, and $A\left(\lambda\right) = \lambda$.

Example 2.2. The categorical distribution for k **possible categories**.

The *categorical* distribution is generally used in classification problems in which the task is to classify an object in one of k possible categories (or classes). It is also a key ingredient in finite mixture models; see Paper III. The categorical distribution has the probability mass function:

$$f(\mathbf{y}|\boldsymbol{\lambda}) = \left(1 - \sum_{h=1}^{k-1} \lambda_h\right)^{1-\sum_{h=1}^{k-1} y_h} \prod_{h=1}^{k-1} \lambda_h^{y_h},$$
$$= \exp\left(\left(\sum_{h=1}^{k-1} y_h \log \frac{\lambda_h}{1 - \sum_{h=1}^{k-1} \lambda_h}\right) + \log\left(1 - \sum_{h=1}^{k-1} \lambda_h\right)\right), \tag{2.6}$$

where, $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_{k-1})$ is the parameter vector, λ_h is the probability of observing the category h, and y_h are indicators of the observed category ($y_h = 1$ if $Y = h$ and $y_h = 0$ otherwise). This distribution is subject to the condition that $\sum_{h=1}^{k} \lambda_h = 1$, which is the reason why only $k-1$ parameters are needed to describe it. From Eq. 2.6, it is clear that $T_h(\boldsymbol{\lambda}) = y_h$, $b_h(\boldsymbol{\lambda}) = \log \frac{\lambda_h}{1-\sum_{h=1}^{k-1} \lambda_h}$, and $A(\boldsymbol{\lambda}) = \log(1 - \sum_{h=1}^{k-1} \lambda_h)$.

A special case of this distribution is the Bernoulli distribution ($k = 2$) used to model a binary outcome Y ($Y = 1$: success and $Y = 0$: failure). Here, $b(\lambda) = \log\left(\frac{\lambda}{1-\lambda}\right)$, $T(y) = y$, $a(y) = 0$ and $A(\lambda) = -\log(1 - \lambda)$.

Example 2.3. Gamma distribution for continuous data.

The *Gamma distribution* is often used to model continuous random variables taking values on the positive real line, i.e. $y \in (0, \infty)$. The distribution has two parameters $\boldsymbol{\lambda} = (\alpha, \beta)$ and its density function has the form:

$$f(y|\boldsymbol{\lambda}) = \frac{\beta^\alpha}{\Gamma(\alpha)} y^{\alpha-1} \exp(-\beta y) = \exp\left((\alpha - 1)\log y - \beta y + \alpha \log \beta - \log \Gamma(\alpha)\right), \tag{2.7}$$

where $\alpha, \beta > 0$, and $\Gamma(\alpha)$ is the Gamma function. From Eq. 2.7 we see that $b_1(\boldsymbol{\lambda}) = -\beta$, $b_2(\boldsymbol{\lambda}) = \alpha - 1$, $T_1(y) = y$, $T_2(y) = \log y$, $A(\boldsymbol{\lambda}) = \log \Gamma(\alpha) - \alpha \log \beta$ and $a(y) = 0$. The special case with $\alpha = 1$ is the *exponential* distribution, well known in survival data analysis; see Chapter 4 for more details.

All of the above example has $T(y) = y$, i.e. the scalar y itself is the sufficient statistic. This brings us to the following definition.

Definition 2.2. Canonical exponential family.

The canonical exponential family for a scalar response variable Y is characterized by the density function (or probability mass function):

$$f(y|\lambda, \phi) = \exp\{\phi[\lambda y - A(\lambda)] + a(y, \phi)\}, \tag{2.8}$$

where ϕ is a dispersion parameter, and λ is the the model's parameter in the canonical form and λ is often referred to as the natural parameter.

Taking the Gamma model example, the canonical form is specified by defining $\lambda = -\frac{\beta}{\alpha}$, $\phi = \alpha$ and $A(\lambda) = -\log\lambda$; $a(y, \phi)$ contains all remaining factors. In the Poisson example $\phi = 1$, the natural parameter is $\theta = \log\lambda$ and $A(\theta) = e^{\theta}$.

The function $A(\lambda)$ is related to the first two moments of the exponential family in the canonical form. If Y has a canonical exponential family density function, then (Nelder and Wedderburn, 1972)

$$\mu = E(y|\lambda, \phi) = \frac{\partial A(\lambda)}{\partial \lambda},\ V(y|\lambda, \phi) = \phi^{-1}\frac{\partial^2 A(\lambda)}{\partial \lambda^2}. \tag{2.9}$$

2.3. Generalized linear models

The class of generalized linear models expresses the conditional density function of the response variable given the covariates as the exponential family

$$f(y_i|\mathbf{x}_i) = \exp\{\phi[\lambda_i y - A(\lambda_i)] + a(y_i, \phi)\},$$

where the natural parameter λ_i are connected to the covariates via one-to-one and twice differentiable link functions G

$$\lambda_i = G(\eta_i), \tag{2.10}$$

and

$$\eta_i = \mathbf{x}_i^{\top}\boldsymbol{\beta} \tag{2.11}$$

From Eq. 2.9, it can be noted that the conditional expected value of the response variable μ is a function of the natural parameter; i.e, $\mu = h(\lambda)$ and generally this function is one-to-one (Sundberg, 2019, chapter 3). Therefore GLM can be defined using three equivalent parameters: λ, μ and η; in Figure 2.1 we show their relationships.

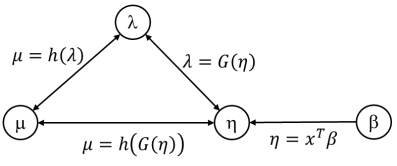


Figure 2.1.: Graphical representation of the relationships of parameters in generalized linear models. The diagram was taken from Sundberg (2019).

The link function between μ and η can be obtained by substituting Eq.2.10 into $h(\lambda)$. In the special case of canonical link functions, which uses the natural parameter as

the link function (G is an identity function; i.e, $\lambda = \eta$), we have the inverse link function, $\mu = h(\eta)$ common in the literature.

Coming back to our examples in Section 2.2, the canonical link function for the Poisson model is the log link

$$\log \lambda = \eta, \tag{2.12}$$

which implies that $G(\eta) := e^{\eta}$. In the Bernoulli model for binary response variables, the canonical link function is the logit function

$$\log\left(\frac{\lambda}{1-\lambda}\right) = \eta, \ 0<\lambda<1. \tag{2.13}$$

Alternative link functions are often used to impose some conditions such as asymmetry or simply to facilitate computations. For binary response data, one popular alternative to the logit link function is the probit function $\lambda = \Phi(\eta)$, where Φ is the normal cumulative density function (CDF). Both probit and logit links are symmetric around $\eta = 0$; if asymmetry is a desired property one can use, for instance, the complementary log-log (CLL) transformation, $\log(-\log(1-\lambda)) = \eta$ (Lambert, 1992).

Figure 2.2 illustrates the difference between these link functions. The probit link saturates faster than the logit, however the CLL link saturates even faster than the probit. The asymmetric property of the CLL allows some considerable probability mass of success even for negative linear predictors.

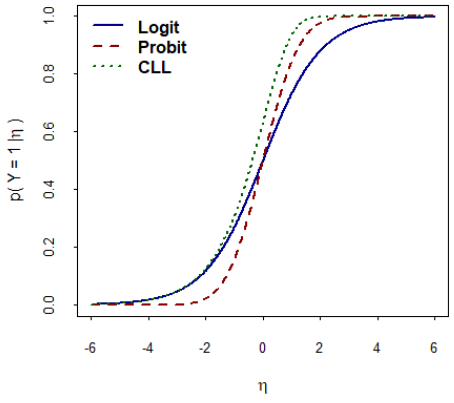


Figure 2.2.: Some illustrations of link functions for binary response data.

2.4. Dynamic generalized linear models

In this section we introduce the class of Dynamic GLM to set some notations and provide their foundations. We assume that the response y_j and the covariates $\mathbf{x}_j$ are observed at discrete time points, $j = 1, 2, \ldots$, and the information available at time j is denoted by $D_j = (\mathbf{x}_j, y_j)$. Dynamic generalized linear models (DGLM) assume that the response variable has a canonical exponential family type distribution with time-varying canonical parameter λ_j and dispersion parameter ϕ_j

$$f_j\left(y_j|\mathbf{x}_j\right) = \exp\left\{\phi_j\left[\lambda_j y_j - A\left(\lambda_j\right)\right] + a(y_j, \phi_j)\right\}, \tag{2.14}$$

and as in GLMs, the covariates enter the model linearly via the linear predictor

$$\eta_j = \mathbf{x}_j^\top \boldsymbol{\beta}_j, \tag{2.15}$$

which is linked to the model's parameter through the link function

$$\lambda_j = G\left(\eta_j\right). \tag{2.16}$$

In many GLM and DGLM the dispersion parameter is fixed, however it is also possible to model the dispersion parameter by linking it to a set of covariates as is done in Paper III. The distinguishing assumption of DGLM is that λ_j vary through time. The covariates are assumed known and the only quantity that drives the time evolution is $\boldsymbol{\beta}_j$. A key component of this class of models is therefore the process that govern the evolution of the regression coefficients.

To ensure a smooth evolution of the regression coefficients, it is necessary to define an additional transition density $p(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, \ldots, \boldsymbol{\beta}_0)$ that models dependencies among the regression coefficients over time. This process is known as the *state* model as it describes the underlying hidden states $\{\boldsymbol{\beta}_j,\ j = 1, 2, \ldots\}$ at each time point j. The most common state model is the first-order Markovian process, where the current state $\boldsymbol{\beta}_j$ depends on the past states only though the most recent state $\boldsymbol{\beta}_{j-1}$. The state model is therefore fully defined by an initial distribution $p(\boldsymbol{\beta}_0)$ and a transition model $p(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1})$.

In addition to the Markovian assumption on the state model, the observations y_j are assumed to be conditionally independent of its past historic observations $\{y_1, \ldots, y_{j-1}\}$ given $\boldsymbol{\beta}_j$. This gives an appealing interpretation of the states; we can think of them as some quantitative summary of the information from the observation process history essential for predicting the process's future behavior. In Figure 2.3 we illustrate this dependency graphically.

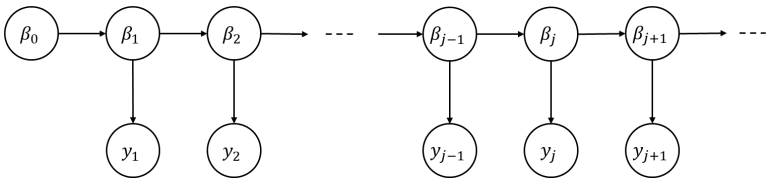


Figure 2.3.: Graphical representation of a state space model showing the dependencies of the hidden state β_j and the observable quantity y_j at different time points. The linear predictor η_j and the link function are omitted because they are known functions of the states.

The simplest and widely applied transition model for regression models is the first order random walk process:

$$\boldsymbol{\beta}_j = \boldsymbol{\beta}_{j-1} + \boldsymbol{v}_j,\ \boldsymbol{v}_j \sim N\left(0,\ \mathbf{V}_j\right), \tag{2.17}$$

where $\mathbf{V}_j$ is a known covariance matrix, and $\boldsymbol{v}_j$ is an innovation noise vector independent of $\boldsymbol{\beta}_j$. The covariance matrix $\mathbf{V}_j$ can be a full matrix as in West et al. (1985), or a diagonal matrix as suggested by Fahrmeir and Kneib (2011). Finally, the distribution of the initial state is assumed to be a Gaussian distribution with some mean vector $\mathbf{m}_0$ and covariance matrix $\mathbf{C}_0$

$$\boldsymbol{\beta}_0 \sim N\left(\mathbf{m}_0, \mathbf{C}_0\right). \tag{2.18}$$

To express lack of knowledge in the initial state, the mean vector is generally set to zero and the covariance matrix to a diagonal matrix with large entries, e.g, $\mathbf{C}_0 = \text{Diag}(100)$.

2.5. Dynamic link functions

So far, we have considered dynamic models where the source of time variation is solely on the model's parameter; here we explore an alternative source: the link function. Canonical links are usually used for convenience without guidance from data.

One way to define data-driven link functions is to use the parametric link transformation approach of Czado (1997). This family of link transformations builds on the canonical link functions G and incorporate an additional parameter $\boldsymbol{\psi} = (\psi_1, \psi_2)$ to transform the tails of the underlying canonical link.

The parametric link is defined as

$$\lambda = G\left(g(\eta, \boldsymbol{\psi})\right), \tag{2.19}$$

where the transformation

$$g(\eta, \boldsymbol{\psi}) = \begin{cases} \frac{(1+\eta)^{\psi_1}-1}{\psi_1} & \text{for } \eta \geq 0 \\ -\frac{(1-\eta)^{\psi_2}-1}{\psi_2} & \text{for } \eta < 0 \end{cases}, \tag{2.20}$$

is a monotone transformation that modifies the left tail ($\eta < 0$) and the right tail ($\eta \geq 0$) separately. The parameters ψ_1 and ψ_2 are both defined on the real line. As a result of Eq. 2.20, standard canonical link functions are special cases of $G\left(g\left(\eta, \boldsymbol{\psi}\right)\right)$ represented by $\boldsymbol{\psi} = 1$.

The effect of these transformations is illustrated in Figure 2.4 using three commonly models: the logistic, Normal, and Poisson models. For the Poisson model, only the right tail transformation is shown, while for the other models only the left tail transformation is shown.

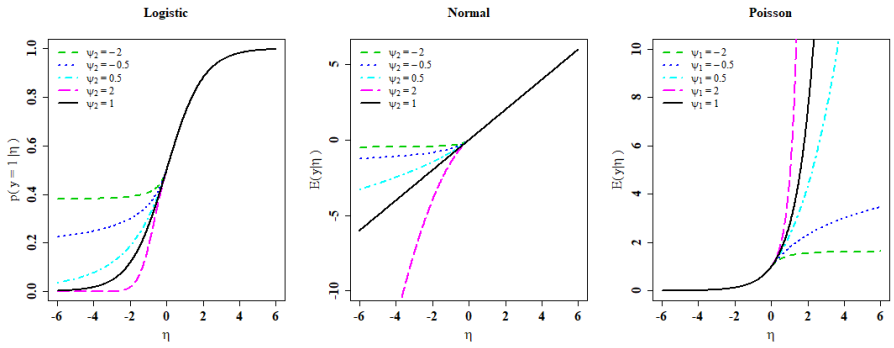


Figure 2.4.: Illustrations of the effect of the parameters in the parametric link function. The left panel displays the probability of success for a binary classifier, the middle panel presents the mean function of a normal model, while the right panel shows the mean of a Poisson regression model.

It is possible to extend the parametric link function in Eq. 2.19 to the dynamic setting. This is done in Paper IV where we allow $\boldsymbol{\psi}_j$ to evolve smoothly over time according to the parameter evolution process in Eq. 2.17, independently of the regression coefficients $\boldsymbol{\beta}_j$.

2.6. Dynamic mixture of Experts models

The density of the response variable may not be well specified by a simple distribution, for example resulting from missing explanatory variables that give rise to clustering and multimodality.

As illustrated in Figure 2.5, finite mixture models offer a way to combine different simple densities to obtain more flexible densities that can fit complex distributional forms of the the response variable. The density of a response variable modeled as a finite mixture of K components has the form,

$$f\left(y|\boldsymbol{\lambda}\right) = \sum_{k=1}^{K} w_k f_k\left(y|\lambda_k\right), \tag{2.21}$$

where $f_k\left(y|\lambda_k\right)$ is the component model and w_k is the normalized mixture weight of the mixture component k; i.e $\sum_{k=1}^{K} w_k = 1$.

To build more flexible models we often need to extend the density in Eq. 2.21 with information from the covariates. Mixture of experts (ME) models (Jacobs et al., 1991; Jordan and Jacobs, 1994) provide a framework to specify the density in Eq. 2.21 as a finite mixture model with covariate-dependent component and mixture weights. ME models are very popular in machine learning applications especially for classification and clustering (Yuksel et al., 2012).

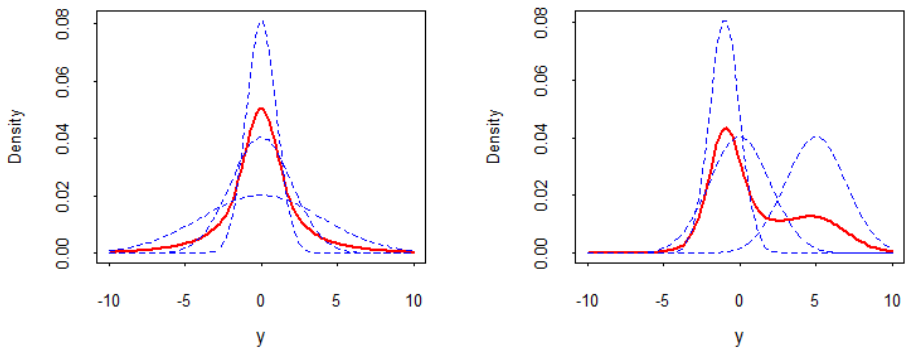


Figure 2.5.: Example of finite mixture densities. The left panel displays a mixture of three univariate Gaussian density components with means 0 and variances $1, 2, 4$ respectively. The right panel presents a mixture of three univariate Gaussian density components with means $-1, 0, 5$ and variances $1, 2, 2$ respectively. The solid line represents the mixture density, while the dashed lines represent the component densities.

The standard ME models assume component models $f_k(y|\lambda_k)$ from the exponential family with natural parameter λ_k linked to the covariates through the link function

$$\lambda_k = G(\eta_k),\ \eta_k = \mathbf{x}^\top \boldsymbol{\beta}_k,$$

and the mixture weights are modeled via the multinomial link function

$$w_k = H(\varphi_k) = \frac{\exp(\varphi_k)}{1 + \sum_{h=2}^{K} \exp(\varphi_h)},\ \varphi_k = \mathbf{x}^\top \boldsymbol{\theta}_k,\ k = 2, \dots K,$$

where the first component is considered as the reference category by specifying $\varphi_1 = 0$. A concise introduction to ME models is provided in Gormley and Frühwirth-Schnatter (2019).

Some extensions to the standard ME models have been proposed; Villani et al. (2012) extended the component models to density functions belonging to essentially any parametric family and use Bayesian variable selection in all parts of the model. Paper III generalizes the ME models to a dynamic setting by assuming the finite mixture:

$$f_j(y_j|\mathbf{x}_j) = \sum_{k=1}^{K} w_{jk} f_{jk}(y_j|\lambda_{jk}), \tag{2.22}$$

where both the mixing weights and the component models depend on covariates though separate link functions

$$w_{jk} = H(\varphi_{jk}),\ \varphi_{jk} = \mathbf{x}_j^\top \boldsymbol{\theta}_{jk}, \tag{2.23}$$

$$\lambda_{jk} = G(\eta_{jk}),\ \eta_{jk} = \mathbf{x}_j^\top \boldsymbol{\beta}_{jk}, \tag{2.24}$$

and the regression coefficients evolve smoothly over time as per Eq. 2.17.

3. Sequential inference

Bayesian inference is in principle well suited for problems where the posterior can be sequentially updated as new data arrive over time. However, the practical computations can be demanding. This chapter presents several simulation methods for Bayesian inference that are particularly suitable for sequential problems.

We describe the sequential posterior analysis in Section 3.1 and we proceed with an introduction of the generic importance sampling (IS) algorithm in Section 3.2 and the sequential importance sampling algorithm in Section 3.3. Further, we present a brief introduction to particle filter algorithms in Section 3.4 and discusses how to improve their efficiency in Section 3.5. So far, the covariance matrix in the parameter evolution process (2.17) has been assumed known. We describe a method for estimating it sequentially in Section 3.6.

3.1. The online posterior

As in previous chapters, we assume that the data arrive in batches $D_j = (\mathbf{x}_j, \boldsymbol{y}_j)$, where $\boldsymbol{y}_j$ and $\mathbf{x}_j$ are, respectively, the response and the covariates observed at the time point j. The batch D_j may contain a single observation as in time series applications or several data points observed over a time interval. The observations within D_j are assumed independent given $\boldsymbol{\beta}_j$. This allows us to represent the likelihood for $\boldsymbol{y}_j$ conditional to $\mathbf{x}_j$ as

$$f_j\left(\boldsymbol{y}_j|\mathbf{x}_j, \boldsymbol{\beta}_j\right) = \prod_{i=1}^{n_j} f_j\left(y_{ij}|\mathbf{x}_{ij}, \boldsymbol{\beta}_j\right),$$

where n_j is the size of D_j and f_j is the density of a general regression model as discussed in Section 2.4, where the current notation emphasizes the important role of the regression coefficients $\boldsymbol{\beta}_j$ for the algorithms presented in this chapter.

The assumptions discussed in Section 2.4 allow for online inference. The joint likelihood for all observed response variables $\boldsymbol{y}_{1:J} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_J)$ given all covariates $\mathbf{x}_{1:J} = (\mathbf{x}_1, \ldots, \mathbf{x}_J)$ can be factorized into likelihood contributions of the batches,

$$f\left(\boldsymbol{y}_{1:J}|\mathbf{x}_{1:J}, \boldsymbol{\beta}_{1:J}\right) = \prod_{j=1}^{J} f_j\left(\boldsymbol{y}_j|\mathbf{x}_j, \boldsymbol{\beta}_j\right), \tag{3.1}$$

where $\boldsymbol{\beta}_{1:J} = (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_J)$. The likelihood factorization (3.1) facilitates computing the predictive density sequentially as data become available:

$$f_j\left(\boldsymbol{y}_j|\mathbf{x}_j, D_{1:j-1}\right) = \int f_j\left(\boldsymbol{y}_j|\mathbf{x}_j, \boldsymbol{\beta}_j\right) p\left(\boldsymbol{\beta}_j|\, D_{1:j-1}\right) d\boldsymbol{\beta}_j, \tag{3.2}$$

where $D_{1:j-1}$ denotes all data observed from time 1 up to time $j-1$; we use such notation throughout this chapter to denote a vector containing a sequence of parameters/variables from time 1 up to the specified time limit.

The prior density $p\left(\boldsymbol{\beta}_j|\, D_{1:j-1}\right)$ in Eq. 3.2 can also be computed recursively,

$$p\left(\boldsymbol{\beta}_j|\, D_{1:j-1}\right) = \int p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}\right) p\left(\boldsymbol{\beta}_{j-1}|D_{1:j-1}\right) d\boldsymbol{\beta}_{j-1}, \tag{3.3}$$

where $p\left(\boldsymbol{\beta}_{j-1}|D_{1:j-1}\right)$ is the *online posterior* at time $j-1$; it is also known as the *filtering* density (Doucet et al., 2001). The online posterior density at each time point j is therefore computed via the Bayes' theorem:

$$p\left(\boldsymbol{\beta}_j|\, D_{1:j}\right) \propto f_j\left(\boldsymbol{y}_j|\mathbf{x}_j, \boldsymbol{\beta}_j\right) p\left(\boldsymbol{\beta}_j|D_{1:j-1}\right). \tag{3.4}$$

The challenge here is that, except for dynamic linear Gaussian models, the online posterior for all other models developed in the thesis is intractable. In this chapter we introduce sequential Monte Carlo methods, commonly referred to as *particle filtering* algorithms, which can be used to simulate from the online posterior.

These algorithms use *sequential importance sampling* to approximate the online posterior empirically. The key ingredient in these algorithms is the *proposal* density from which parameters are sampled from. This density is usually easy to sample from and should ideally be close to the online posterior in order to achieve good performance. The main objective of this chapter is to introduce a method of designing efficient proposal densities for the models introduced in Chapter 2. We focus on designing proposal density for filtering problems; the same approach can be easily extended to smoothing problems where the target is the posterior, $p(\boldsymbol{\beta}_j|\, D_{1:J})$, based on all the data. More details on smoothing problems are provided in Paper I.

The next section introduces the generic importance sampling (IS) algorithm for a model with static parameters. The IS algorithm is the foundation for the sequential importance sampling and the particle filter algorithms that will follow in subsequent sections.

3.2. Importance sampling

The IS algorithm is usually used to compute the posterior expectation

$$E\left(\varphi\right) = \int \varphi\left(\boldsymbol{\beta}\right) p\left(\boldsymbol{\beta}|D\right) d\boldsymbol{\beta}, \tag{3.5}$$

of integrable functions $\varphi(\boldsymbol{\beta})$ of the parameter $\boldsymbol{\beta}$. The density $p(\boldsymbol{\beta}|D)$ is the normalized posterior distribution and D is the observed data. The expectation in Eq. 3.5 is essential in any Bayesian analysis, for example when computing the posterior mean, $\varphi(\boldsymbol{\beta}) = \boldsymbol{\beta}$ or the marginal posterior density as in Eq. 3.2.

If we have a sample of M parameters $\{\boldsymbol{\beta}^m\}_{m=1}^M$ independently and identically distributed according the posterior $p(\boldsymbol{\beta}|\, D)$, the Monte Carlo (MC) integration method approximates Eq. 3.5 empirically:

$$\hat{E}_{MC}^{(M)}(\varphi) = \frac{1}{M}\sum_{m=1}^{M}\varphi(\boldsymbol{\beta}^m). \tag{3.6}$$

The estimator $\hat{E}_{MC}^{(M)}(\varphi)$ has some appealing properties: It is unbiased and strongly consistent; that is, by the strong law of large numbers,

$$\hat{E}_{MC}^{(M)}(\varphi) \overset{\text{a.s}}{\to} E(\varphi), \text{ as } M \to \infty,$$

where $\overset{\text{a.s}}{\to}$ means almost sure convergence, which means that $\hat{E}_{MC}^{(M)}(\varphi)$ converges to $E(\varphi)$ as $M \to \infty$ with probability 1. Furthermore, the central limit theorem

$$\sqrt{N}\left(\hat{E}_{MC}^{(M)}(\varphi) - E(\varphi)\right) \overset{\text{d}}{\to} N\left(0, \sigma_\varphi^2\right), \text{ as } M \to \infty, \tag{3.7}$$

holds provided that $\sigma_\varphi^2 \equiv E_{\beta|D}(\varphi^2) - E(\varphi)^2$ is finite. The notation $\overset{\text{d}}{\to}$ means convergence in distribution. The central limit theorem suggests that the accuracy of the approximation (3.5) improves at a rate that is independent of the size of the data and dimension of the parameter as M increases.

The MC approximation in Eq. 3.6 is possible only if we can sample directly from the posterior. However, this is rarely possible since the posterior tends to be intractable. It is therefore necessary to resort to indirect simulation methods; one such method is importance sampling (Geweke, 1989) which we describe next.

The IS algorithm requires a proposal density $q(\boldsymbol{\beta})$ that approximates the unnormalized posterior and is easy to sample from. This proposal density can then be used to rewrite the expectation in Eq. 3.5 as follows:

$$E(\varphi) = \frac{\int \varphi(\boldsymbol{\beta})\, W q(\boldsymbol{\beta})\, d\boldsymbol{\beta}}{\int W q(\boldsymbol{\beta})\, d\boldsymbol{\beta}},$$

where

$$W \propto \frac{p(\boldsymbol{\beta}|D)}{q(\boldsymbol{\beta})}, \tag{3.8}$$

are ratios of the target and proposal densities referred to as *importance weights*. If $\{\boldsymbol{\beta}^m\}_{m=1}^M \overset{\text{iid}}{\sim} q(\boldsymbol{\beta})$, then using the approximation analogous to Eq. 3.6 we obtain the importance sampling estimator

$$\hat{E}_{IS}^{(M)}(\varphi) = \frac{\sum_{m=1}^M \varphi(\boldsymbol{\beta}^m) W^m}{\sum_{m=1}^M W^m} = \sum_{m=1}^{M}\varphi(\boldsymbol{\beta}^m)\tilde{W}^m, \tag{3.9}$$

where

$$\tilde{W}^m = \frac{W^m}{\sum_{m=1}^{M} W^m}, \tag{3.10}$$

are normalized importance weights. The importance weights can also be used to approximate the posterior density empirically:

$$\hat{p}\left(\boldsymbol{\beta}|D\right) = \sum_{m=1}^{M} \tilde{W}^m \delta_{\boldsymbol{\beta}^m}\left(\boldsymbol{\beta}\right),$$

where $\delta_{\boldsymbol{\beta}^m}\left(\boldsymbol{\beta}\right)$ is the Dirac-delta function located at $\boldsymbol{\beta}^m$; i.e, an indicator function that equals to one if $\boldsymbol{\beta} = \boldsymbol{\beta}^m$ and zero otherwise. The estimator (3.9) is biased since it is a ratio of two means, but it is consistent provided that the conditions stated in Geweke (1989, Theorem 1&2) are fulfilled. The central limit theorem in Eq. 3.7 still holds. Assuming that the proposal density is normalized, the asymptotic variance of the IS estimator is given by

$$\sigma_{IS}^2 = \frac{1}{M}\left[\int W q\left(\boldsymbol{\beta}\right) d\boldsymbol{\beta}\right]^{-2} \int \left(\varphi - E_{\beta|D}\left(\varphi\right)\right)^2 W p\left(\boldsymbol{\beta}|D\right) d\boldsymbol{\beta}.$$

One point to note from the asymptotic variance is that the accuracy of the IS estimator depends largely on the quality of the importance weights. If the proposal is proportional to the target density then the importance weights will be uniform. However, if the proposal is inconsistent with the target density, a large majority of the importance weights will be close to zero and the IS estimator will be dominated by a single or a handful of draws. This problem is known as the *importance weight degeneracy*.

3.3. Sequential importance sampling

The importance sampling algorithm discussed in Section 3.2 can be extended to a dynamic setting (Doucet et al., 2001). In order to compute the importance weights recursively, the proposal distribution needs to be factorizable such that

$$q\left(\boldsymbol{\beta}_j|\, D_{1:j}\right) = q\left(\boldsymbol{\beta}_{j-1}|\, D_{1:j-1}\right) q\left(\beta_j|\boldsymbol{\beta}_{j-1},\, D_j\right). \tag{3.11}$$

This allows us to propose new draws $\boldsymbol{\beta}_j^m$ and to update the importance weights sequentially as

$$W_j^m \propto W_{j-1}^m \frac{f_j\left(\boldsymbol{y}_j|\mathbf{x}_j, \boldsymbol{\beta}_j^m\right) p\left(\boldsymbol{\beta}_j^m|\boldsymbol{\beta}_{j-1}^m\right)}{q\left(\boldsymbol{\beta}_j^m|\boldsymbol{\beta}_{j-1}^m,\, D_j\right)}. \tag{3.12}$$

Furthermore, the online posterior is approximated empirically:

$$\hat{p}\left(\boldsymbol{\beta}_j|\, D_{1:j}\right) = \sum_{m=1}^{M} \tilde{W}_j^m \delta_{\boldsymbol{\beta}_j^m}\left(\boldsymbol{\beta}_j\right),$$

where $\delta_{\boldsymbol{\beta}_j^m}\left(\boldsymbol{\beta}_j\right)$ is the Dirac delta function located at $\boldsymbol{\beta}_j^m$, and

$$\tilde{W}^m = \frac{W^m}{\sum_{m=1}^{M} W^m},$$

are normalized importance weights.

The sequential importance sampling algorithm starts from $j = 0$ with the initial parameters $\{\boldsymbol{\beta}_0^m\}_{m=1}^M$ proposed from the initial prior density $p\left(\boldsymbol{\beta}_0\right)$ and the importance weights initialized uniformly. Then, for subsequent intervals $j = 1, \ldots, J$, a new set of parameters $\left\{\boldsymbol{\beta}_j^m\right\}_{m=1}^M$ is sampled from $q\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1},\, D_j\right)$ conditional to the parameters sampled in the previous step, and the importance weights are updated as per Eq. 3.12. In the SMC literature, the parameter draws $\boldsymbol{\beta}_j^{(m)}$ are commonly referred to as *particles*.

Effective sample size The weight degeneracy problem is inevitable in sequential importance sampling since the variance of the importance weights grows with time (Doucet et al., 2000). A standard criterion for monitoring the importance weight degeneracy is the *effective sample size* which is approximated as

$$ESS = \frac{1}{\sum_{m=1}^{M}\left(\tilde{W}_j^m\right)^2},$$

at each time point j. The ESS can take values between 1 and M; it takes the lowest value if one weight is 1 which indicates a high degeneracy, and it is maximal when the importance weights are proportional to a constant.

3.4. Particle filtering

The weight degeneracy problem can be mitigated by adding a resampling step between the iterations of the sequential importance sampling algorithm (Gordon et al., 1993). This removes particles with low importance weights, keeping only particles with significant importance weights. Several resampling schemes have been proposed in the literature (Carpenter et al. (1999); Liu and Chen (1998); Fearnhead and Clifford (2003)). The most common schemes are the multinomial, stratified, systematic, and residual schemes. It has been shown that the last three schemes have lower MC variance than the multinomial scheme (Douc and Cappé, 2005).

Particle filter algorithms use the sequential importance sampling algorithm with a resampling step between iterations. Algorithm 1 describes the steps for a generic particle filter, using the multinomial resampling. The most widely applied particle filters are the bootstrap particle filter (Gordon et al., 1993) and the auxiliary particle filter (Pitt and Shephard, 1999), which we present here as a background on particle filter algorithms. For interested reader, a larger menu of particle filter algorithms can be found in Arulampalam et al. (2002); see also Doucet and Johansen (2009) for some theoretical aspects.

Algorithm 1: Generic particle filter

Initialization: $j = 0$

Sample $\boldsymbol{\beta}_0^m \sim p(\boldsymbol{\beta}_0)$ and set $W_j^m = \frac{1}{M}$

for $j = 1$ **to** J **do**

  for $m = 1$ **to** M **do**

    - Propose particles $\boldsymbol{\beta}_j^m \sim q_j\left(\boldsymbol{\beta}_j | \boldsymbol{\beta}_{j-1}^m, D_{1:j}\right)$
    - Compute importance weights W_j^m following Eq.3.12

  end

  - Normalize the importance weights
  - Resample
    Sample with replacement indices $a_j^m \sim \text{Mult}\left(\left\{\widetilde{W}_j^m\right\}_{m=1}^M\right)$
    and set $W_j^m = \frac{1}{M}$ and $\boldsymbol{\beta}_{0:j}^m = \left(\boldsymbol{\beta}_{0:j-1}^{a_j^m}, \boldsymbol{\beta}_j^{a_j^m}\right)$
    for $m = 1, \ldots, M$

end

Output: $\left\{\boldsymbol{\beta}_{0:J}^m, \tilde{W}_{1:J}^m\right\}_{m=1}^M$

The bootstrap particle filter: Given a particle system $\left\{\boldsymbol{\beta}_{j-1}^m, \widetilde{W}_{j-1}^m\right\}_{m=1}^M$ at the time point $j-1$, the prior Eq. 3.3 can be approximated as:

$$\hat{p}\left(\boldsymbol{\beta}_j | D_{1:j-1}\right) = \sum_{m=1}^{M} p\left(\boldsymbol{\beta}_j | \boldsymbol{\beta}_{j-1}^m\right) \widetilde{W}_j^m. \tag{3.13}$$

The bootstrap particle filter uses the distribution in Eq. 3.13 as the proposal density. This is equivalent to sampling from the parameter evolution $p(\boldsymbol{\beta}_j | \boldsymbol{\beta}_{j-1})$ followed by a resampling step. To see this, note that the finite mixture distribution in Eq. 3.13 can be sampled by drawing a historic path index a_j^m with probability proportional

to the importance weights $\left\{\widetilde{W}_j^m\right\}_{m=1}^M$, followed by sampling from the parameter evolution

$$\boldsymbol{\beta}_j^m \sim p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}^{(a_j^m)}\right).$$

The bootstrap particle filter is easy to implement since it requires only evaluations of the likelihood and it is parallelizable. However, it has some drawbacks: First, it requires that the state model be known and easy to sample from, which may not always be the case. An example is the class of models presented in this thesis in which the covariance matrix of random walk process in Eq. 2.17 is not known. Second, it proposes from the prior which may lead to highly degenerate importance weights if the prior and the likelihood are incompatible - i.e. if the prior and likelihood have very small overlap.

The generic auxiliary Particle filter The auxiliary particle filter proposes particles from the empirical approximation of the online predictive density (3.2),

$$\hat{f}\left(\boldsymbol{y}_j|\boldsymbol{y}_{1:j-1}\right) = f_j\left(\boldsymbol{y}_j|\mathbf{x}_j, \boldsymbol{\beta}_j\right) \sum_{m=1}^M p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}^m\right) \widetilde{W}_j^m.$$

The idea is to use the available particles to construct new mixture weights, which we refer to as the resampling weights,

$$\nu^{(m)} \propto f_j\left(\boldsymbol{y}_j|\mathbf{x}_j, \mu_j^m\right) \widetilde{W}_j^m,$$

where μ_j^m is a quantity estimated from $p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}\right)$ such as the mean, mode or a draw from the transition density. The particles $\boldsymbol{\beta}_j$ and the particle path indexes a_j are proposed jointly from

$$q\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_j\right) = \sum_{m=1}^M \nu_j^m p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}^m\right),$$

in a similar way as described for the bootstrap particle filter. The importance weights associated with the generic auxiliary particle filter are updated as

$$W_j^m \propto \frac{f_j\left(\boldsymbol{y}_j|\mathbf{x}_j, \boldsymbol{\beta}_j^m\right)}{f_j\left(\boldsymbol{y}_j|\mathbf{x}_j, \mu_j^m\right)}.$$

The Auxiliary particle filter is also easy to implement since the importance weights are simply the ratio of two likelihood quantities. The resampling step allows us to propose from the particles which have higher predictive weights. However, it is still not efficient to propose particles from $p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}\right)$. Ideally, the proposal density for state-space models is the density proportional to $p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_j\right)$ (Doucet et al., 2000, Proposition 2), which is unfortunately intractable for the models considered in this thesis. It is therefore necessary to find $q\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_j\right)$ close to $p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_j\right)$ which is easy to sample from.

3.5. Designing efficient proposal distributions

Here we present two methods for designing an efficient proposal density.

3.5.1. Local linearization

One way to design efficient proposal densities is the local linearization approach (Doucet et al., 2000) which uses the normal approximation of $p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right)$ described in Section 1.4.2 as the proposal density. Thus, setting

$$q_j\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right) \equiv N\left(\hat{\boldsymbol{\beta}}_j,\ \Sigma\left(\hat{\boldsymbol{\beta}}_j\right)\right), \tag{3.14}$$

where $\hat{\boldsymbol{\beta}}_j$ is the mode of $p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right)$, and

$$\Sigma\left(\hat{\boldsymbol{\beta}}_j\right)^{-1} = -\left.\frac{\partial^2 \log p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right)}{\partial\boldsymbol{\beta}_j \partial\boldsymbol{\beta}_j^\top}\right|_{\boldsymbol{\beta}_j=\hat{\boldsymbol{\beta}}_j}.$$

If the mode is not available analytically, the Newton–Raphson method can be used to iteratively get to the mode. Starting from $\boldsymbol{\beta}_j^{(0)} = \boldsymbol{\beta}_{j-1}$, the Newton–Raphson method follows the recursion

$$\boldsymbol{\beta}_j^{(k+1)} = \boldsymbol{\beta}_j^{(k)} + \Sigma\left(\hat{\boldsymbol{\beta}}_j^{(k)}\right)\left(\left.\frac{\partial \log p\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right)}{\partial\boldsymbol{\beta}_j}\right|_{\boldsymbol{\beta}_j=\hat{\boldsymbol{\beta}}_j^{(k)}}\right), \tag{3.15}$$

for iterations $k = 0, 1, 2, \ldots$. Notice that the local linearization approach requires computing the inverse of a high dimensional matrix $\Sigma\left(\hat{\boldsymbol{\beta}}_j\right)$ which may pose numerical computation problems. One way to address this is to use the *linear Bayes* method (West et al., 1985).

3.5.2. Linear Bayes

The linear Bayes approach exploits the fact that regression coefficients enter the observation model of Dynamic GLM models through the linear predictor; this implies that$\boldsymbol{\beta}_j$ can be updated conditional on the updated linear predictor η_j. To see this clearly, consider the joint posterior of η_j and $\boldsymbol{\beta}_j$,

$$\begin{aligned} p\left(\eta_j, \boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right) &\propto f_j\left(\boldsymbol{y}_j|\eta_j\right) p\left(\eta_j, \boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j-1}\right) \\ &= f_j\left(y_j|\eta_j\right) p\left(\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j-1}\right) p\left(\boldsymbol{\beta}_j|\eta_j, \boldsymbol{\beta}_{j-1}, D_{1:j-1}\right) \\ &= p\left(\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right) p\left(\boldsymbol{\beta}_j|\eta_j, \boldsymbol{\beta}_{j-1}, D_{1:j-1}\right). \end{aligned} \tag{3.16}$$

Note that the posterior (3.16) is degenerate because the linear predictor is a deterministic function of the regression coefficients and the second factor in the right hand side of Eq. 3.16 does not depend on the current data D_j. Therefore, to update $\boldsymbol{\beta}_j$, we can transfer information from η_j to $\boldsymbol{\beta}_j$ partially through the moments of $p\left(\eta_j|D_{1:j}\right)$.

All needed here is the (degenerate) prior $p\left(\eta_j, \boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j-1}\right)$. Using the random walk parameter evolution process (2.17) and the linear predictor definition, the prior $p\left(\eta_j, \boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j-1}\right)$ can be set to the multivariate normal density

$$\left.\begin{pmatrix} \eta_j \\ \boldsymbol{\beta}_j \end{pmatrix}\right| \boldsymbol{\beta}_{j-1}, D_{1:j-1} \;\sim\; N\left(\begin{pmatrix} \bar{\eta}_j \\ \boldsymbol{\beta}_{j-1} \end{pmatrix}, \begin{bmatrix} Q_j & \mathbf{U}_j^\top \\ \mathbf{U}_j & \mathbf{V}_j \end{bmatrix}\right),$$

where $\bar{\eta}_j = \mathbf{x}_j^\top \boldsymbol{\beta}_{j-1}$ and $\mathbf{V}_j$ is the covariance matrix of the parameter evolution random walk process. The variance Q_j and $\mathbf{U}_j$ are model-dependent; for instance, considering an exponential family model with canonical link function, $\mathbf{U}_j = \mathbf{V}_j\mathbf{x}_j$ and $Q_j = \mathbf{x}_j^\top \mathbf{V}_j \mathbf{x}_j$.

Since the second factor in Eq. 3.16 is conditioned on η_j, the first and second moments of $\boldsymbol{\beta}_j$ can be updated by applying the law of iterated expectations and the law of total variance

$$\begin{aligned} E\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right) &= E\left(E\left(\boldsymbol{\beta}_j|\eta_j, \boldsymbol{\beta}_{j-1}, D_{1:j-1}\right)\middle| D_{1:j}\right) \\ &= \boldsymbol{\beta}_{j-1} + \mathbf{U}_j Q_j^{-1}\left[E\left(\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right) - \bar{\eta}_j\right], \end{aligned} \tag{3.17}$$

$$\begin{aligned} V\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right) &= E\left(V\left(\boldsymbol{\beta}_j|\eta_j, \boldsymbol{\beta}_{j-1}, D_{1:j-1}\right)\middle| D_{1:j}\right) \\ &\quad + V\left(E\left(\boldsymbol{\beta}_j|\eta_j, \boldsymbol{\beta}_{1-1}, D_{1:j-1}\right)\middle| D_{1:j}\right) \\ &= \mathbf{V}_j - \mathbf{U}_j Q_j^{-1}\left[I + V\left(\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right) Q_j^{-1}\right]\mathbf{U}_j^\top. \end{aligned} \tag{3.18}$$

These updated moments of $\boldsymbol{\beta}_j$ are used in the proposal density (3.14) by setting $\hat{\boldsymbol{\beta}}_j = E\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right)$, and $\Sigma\left(\hat{\boldsymbol{\beta}}_j\right) = V\left(\boldsymbol{\beta}_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right)$.

To complete our proposal density requires the first two moments of $\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j}$ in Eq. 3.17 and Eq. 3.18. For models in the exponential family with the canonical link, we can exploit the fact that the natural parameter has a conjugate prior (West et al., 1985) and its posterior $p\left(\lambda_j|D_{1:j}\right)$ is available in closed form. We can, therefore, obtain the moments of $\eta_j|D_{1:j}$ through the normal approximation of $p\left(\lambda_j|D_{1:j}\right)$ with respect to η_j, see 1.1. Some examples where this approach is applied are provided in Paper I and Paper IV. In the general case, where conjugate priors are not available,

we use the second order Taylor series expansion of $\log p\left(\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right)$ evaluated at $\bar{\eta}_j$.

The second order Taylor series expansion of $\pi\left(\eta_j\right) = \log p\left(\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right)$ can be formulated as follows:

$$\begin{aligned}\pi\left(\eta_j\right) &\approx \pi\left(\bar{\eta}_j\right) + \left[\pi\left(\bar{\eta}_j\right)'\right]^\top \left(\eta_j - \bar{\eta}_j\right) + \left(\eta_j - \bar{\eta}_j\right)^\top \left[\pi\left(\bar{\eta}_j\right)''\right] \left(\eta_j - \bar{\eta}_j\right) \\ &= \text{constant} - \frac{1}{2}\left[\eta_j - \bar{\eta}_j - \mu\left(\bar{\eta}_j\right)\right]^\top \Sigma\left(\bar{\eta}_j\right)^{-1}\left[\eta_j - \bar{\eta}_j - \mu\left(\bar{\eta}_j\right)\right],\end{aligned}$$

which means that we can use in Eq. 3.17 and Eq. 3.18 the approximations

$$\hat{E}\left(\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right) = \bar{\eta}_j + \mu\left(\bar{\eta}_j\right), \qquad \hat{V}\left(\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right) = \Sigma\left(\bar{\eta}_j\right), \tag{3.19}$$

where

$$\Sigma\left(\bar{\eta}_j\right) = \left[-\pi\left(\bar{\eta}_j\right)''\right]^{-1}, \qquad \mu\left(\bar{\eta}_j\right) = \Sigma\left(\bar{\eta}_j\right)^{-1}\pi\left(\bar{\eta}_j\right)',$$

$$\begin{aligned}\pi\left(\bar{\eta}_j\right)' = \left.\frac{\partial \log p\left(\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right)}{\partial \eta_j}\right|_{\eta_j=\bar{\eta}_j} &= \left.\frac{\partial \log f_j\left(y_j|\mathbf{x}_j, \boldsymbol{\beta}_j\right)}{\partial \eta_j} - Q_j^{-1}\left(\eta_j - \bar{\eta}_j\right)\right|_{\eta_j=\bar{\eta}_j} \\ &= \left.\frac{\partial \log f_j\left(y_j|\mathbf{x}_j, \boldsymbol{\beta}_j\right)}{\partial \eta_j}\right|_{\eta_j=\bar{\eta}_j},\end{aligned}$$

$$\pi\left(\bar{\eta}_j\right)'' = \left.\frac{\partial^2 \log p\left(\eta_j|\boldsymbol{\beta}_{j-1}, D_{1:j}\right)}{\partial \eta_j \partial \eta_j}\right|_{\eta_j=\bar{\eta}_j} = \left.\frac{\partial^2 \log f_j\left(y_j|\mathbf{x}_j, \boldsymbol{\beta}_j\right)}{\partial \eta_j \partial \eta_j}\right|_{\eta_j=\bar{\eta}_j} - Q_j^{-1}.$$

Note that it is generally easier to approximate the posterior of η_j than $\boldsymbol{\beta}_j$ since η_j has lower dimension; for instance, η_j is a scalar for single component models, and it has $K-1$ length, for finite mixture of K component models. More details and examples on the application of the linear Bayes approach to finite mixture models are provided in Paper III; see also paper IV for examples with parametric link functions.

3.6. Discount factor

We have so far conditioned on a known covariance matrix $\mathbf{V}_j$. In practice, $\mathbf{V}_j$ is not known and needs to be estimated. A fully Bayesian approach sets a prior on $\mathbf{V}_j$. If $\mathbf{V}_j$ is a full matrix, the inverse Wishart distribution can be used as the prior (Gamerman, 1998). For diagonal $\mathbf{V}_j$, i.e $\boldsymbol{V}_j = \text{diag}(\tau_j^2)$, one can set an inverse gamma prior on τ_j^2 (Fahrmeir and Kneib, 2011, Chapter 2). Alternatively, one can

assume that $\nu_j = \log(\tau_j^2)$ follows the random walk process (2.17) as in Lang et al. (2002). This allows the model to adapt locally, but still requires setting a prior for the hyperparameter in the random walk process for ν_j.

The discount factor approach in West et al. (1985) allows us to estimate $\boldsymbol{V}_j$ recursively based on the second moment of the previously estimated online posterior for $\boldsymbol{\beta}_{j-1}$. Assuming that the covariance for $\boldsymbol{\beta}_{j-1}|D_{1:j-1}$ is $\boldsymbol{C}_{j-1}$, it follows from Eq. 2.17 that the covariance matrix for the prior $\boldsymbol{\beta}_j|D_{1:j-1}$ is $\boldsymbol{R}_j = \boldsymbol{C}_{j-1} + \boldsymbol{V}_j$.

The discount factor approach expresses the matrix $\boldsymbol{R}_j$ as $\boldsymbol{R}_j = \frac{\boldsymbol{C}_{j-1}}{\alpha}$,where $0 < \alpha < 1$ is a known discount factor that controls how much information is transferred from the most recent posterior. Given α, $\mathbf{V}_j$ is simply given by

$$\boldsymbol{V}_j = \frac{(1-\alpha)\,\boldsymbol{C}_{j-1}}{\alpha}. \tag{3.20}$$

From Eq. 3.20 one can see that the discount factor controls the smoothness of the parameter evolution. A value of α close to one shrinks the matrix $\boldsymbol{V}_j$ towards zero, leading to very little variation in $\boldsymbol{\beta}_j$ over time; following Liu and West (2001), models with $.95 \leq \alpha < 1$ are essentially static. On the other hand, a value of α close to zero gives the regression parameters more flexibility and allows the model to adapt locally to any changes over time.

Note that the discount factor approach applies in both cases when $\boldsymbol{V}_j$ is a diagonal or a full matrix. In this thesis $\boldsymbol{V}_j$ is considered as a full matrix and inference for α is conducted using some measure of model accuracy such as the Watanabe-Akaike information criterion (WAIC) used in Paper I-II, and the log predictive score used in Paper III-IV.

4. Survival and event history data

The possibility to model the response variable using density functions in the exponential family and extend the model to other distribution functions or to mixtures of distribution functions enables a wide range of applications of the dynamic models discussed in Chapter 2. Some of these applications include model-based clustering of multiple time-series data (Frühwirth-Schnatter et al., 2018; Frühwirth-Schnatter and Kaufmann, 2008)), prediction and forecasting problems in econometrics (Kalli and Griffin, 2014), and dynamic classification problems (McCormick et al., 2012).

In this chapter, we present an overview of another area of application where dynamic regression models are used to model time-to-event data – also known as survival data, duration data, event history data, failure-time data – depending on which area the data originates from. The data consist of event times – time between entry into a study until occurrence of the event together with background variables (covariates $\mathbf{x}$). They may also be in the form of count data, where the response variable represents the number of events occurring at different time points.

We introduce some basic concepts in survival models in Section 4.1 and relate these models to our dynamic regression models where we allow the effects of covariates to vary over time. In Section 4.2 we describe models with partially observed covariates, while we address dynamic models for count data in Section 4.3.

4.1. Survival data

The usual goals in modeling survival data are to determine the distributional shape of the time to event and compare survival experiences among groups. In other words, the latter objective is to model the effect of some covariates on survival experience among groups.

4.1.1. Censoring

Frequently the study ends before all individuals experience the event of interest. Some individuals are also lost to follow-up. Such individuals are said to be *censored* and their time to event is unknown. One of the advantages of survival models is that these individuals are not thrown away because they have unknown values on the event time. Instead, the available information that they have survived the event

until the end of the study or lost to follow-up is used together with the information from those who experienced the event and the parameters of interest are estimated from the combined data.

There are various forms of censoring: right, left, and interval censoring. There is also type 1 censoring – when the study is conducted over a specified period (random censoring) and type 2 censoring – when the study is conducted until a given proportion of individuals experience the event.

We focus on right-censoring of type 1 which is the most commonly encountered in practice. In this case, the response variable of interest is the time, $\tilde{T}$, from a well-defined start until the event occurs or the study ends, whichever comes first and, as such, cannot be negative. The starting time may be entry into a study (for instance in in clinical trials), date of marriage (in the study of divorce risk), or the date of birth of previous child (in studying birth spacing). We only observe the survival time $T = \min(\tilde{T}, C)$ and the censoring indicator variable,

$$d = \begin{cases} 1, & \tilde{T} \leq C \\ 0, & \tilde{T} > C \end{cases},$$

where C represents the censoring time random variable. Hence, the data collected on n participants in the study consist of the survival time t_i, the censoring indicator d_i($d = 1$: if event occurred and $d = 0$: if censored) and the covariate $\mathbf{x}_i$, $i = 1, \ldots, n$.

4.1.2. Hazard and survival functions

Survival data are summarized by three equivalent functions: the density function, the survival function, and the hazard function. Let the survival time T have the probability density function $f(t)$ and distribution function

$$F(t) = P(T \leq t) = \int_0^t f(u)\, du. \tag{4.1}$$

The *survival function,*

$$S(t) = P(T > t) = 1 - P(T \leq t), \tag{4.2}$$

is the probability that the event has not happened by time t; it gives the expected proportion of individuals who have not experienced the event by time t.

The *hazard function* is the instantaneous rate at which the event occurs within a small time interval:

$$\lambda(t) = \lim_{\triangle t \to 0} \frac{P(t \leq T < t + \triangle t | T \geq t)}{\triangle t}, \tag{4.3}$$

where the conditional probability

$$P\left(t \leq T < t + \triangle t | T \geq t\right) = \frac{P\left(t \leq T < t + \triangle t\right)}{P(T \geq t)}.$$

The hazard function can be expressed in terms of the survival function and the density function as follows:

$$\lambda(t) = \frac{f(t)}{S(t)} = -\frac{d}{dt} \log\left(S\left(t\right)\right), \tag{4.4}$$

which gives another definition of the survival function expressed in terms of the hazard function; i.e

$$S\left(t\right) = \exp\left(-\int_0^t \lambda\left(u\right) du\right), \tag{4.5}$$

where the integrand in the expression (4.5)

$$\Lambda\left(t\right) = \int_0^t \lambda\left(u\right) du \tag{4.6}$$

is known as the *cumulative hazard* function.

Survival models are commonly defined using the hazard function. Once the hazard function is known, the density and the survival functions can be deduced from the relationships (4.4) and (4.5) respectively. As an example, we illustrate in Figure 4.1 the relationship of the hazard and survival functions.

The hazard function can be constant or change over time. On the other hand, the survival function is a non-increasing (often decreasing) function. It is equal to 1 at the start of the study since all are 'alive' at the start of the study, and declines towards 0 since the study group is depleted as individuals experience the event with time. This is true for instance if the event is death (as all die sooner or later) but there are situations when the survival function may not attain the value 0. This is the case in the study of marriage, divorce, or employment as there may be individuals who may never experience the event (never marry, divorce, or get employment). The non-parametric Kaplan-Meier estimator (Kaplan and Meier, 1958) is often used to estimate hazard function and survival function for a given data set.

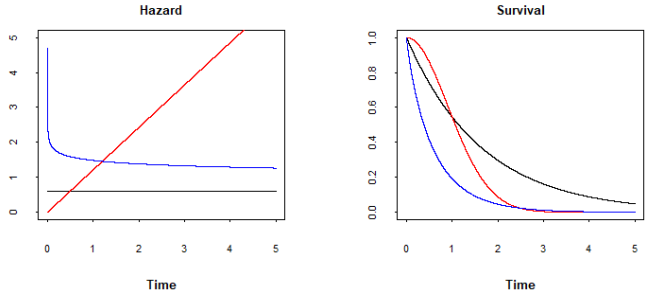


Figure 4.1.: Illustrating the relationship between the hazard and the survival functions from a Weibull distribution with different shape and location parameters. The left panel displays the curves of different hazard function; the right panel shows the survival curve corresponding to each hazard curve.

4.1.3. Dynamic Survival models

The primary goal in many survival analyses is to estimate the effect of covariates on the hazard function. Here, we introduce methods of linking the covariate $\mathbf{x}$ to the hazard function in the dynamic regression model setting. Perhaps the most popular approach is the multiplicative Cox-type (Cox, 1972) hazard model

$$\lambda(t|\mathbf{x}) = \lambda_0(t) \exp\left(\mathbf{x}^\top \boldsymbol{\beta}\right), \tag{4.7}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_P)$ is a P-dimensional vector of regression coefficients representing the effect of the covariates, $\lambda_0(t) > 0$ is an unknown baseline hazard function. The baseline factor in Eq. 4.7 models the evolution of the hazard function through time when the covariate's value is zero; i.e, $\mathbf{x} = 0$. The other factor, the covariate factor, serves to increase or decrease the hazard function relative to the baseline. The model (4.7) is known in the literature as the *proportional hazards* model as the relative hazard function for two individuals with covariates $\mathbf{x}_1$ and $\mathbf{x}_2$, respectively, is

$$\psi\left(t|\mathbf{x}\right) = \frac{\lambda\left(t|\mathbf{x}_1\right)}{\lambda\left(t|\mathbf{x}_2\right)} = \exp\left(\left(\mathbf{x}_1 - \mathbf{x}_2\right)^\top \beta\right). \tag{4.8}$$

It is clear from the expression (4.8) that the ratio of the hazard function of any two individuals remains constant through time. In the special case of a binary covariate, the regression coefficient β is interpreted as the log of the relative hazards of individuals belonging to the category $x = 1$ with respect to the baseline category $x = 0$.

Although the proportional hazards model allows for an appealing interpretation of the regression coefficients, it is too restrictive in some situations. If the study has been carried on over a long period, it may be reasonable to think that the relative hazard changed as time evolves. Also, the effects of covariates $\boldsymbol{\beta}$ may change over time due to other reasons. To relax the assumption of proportionality and allowing the relative hazards to change over time, one can use the following dynamic hazard model:

$$\lambda(t|\mathbf{x}) = \lambda_0(t) \exp\left(\mathbf{x}^\top \boldsymbol{\beta}(t)\right), \tag{4.9}$$

where $\boldsymbol{\beta}(t)$ is a function of time representing the time-dependent effects of covariates and $\lambda_0(t)$ is the baseline hazard function. Note that the model (4.9) can be reformulated such that $\log \lambda_0(t)$ becomes an intercept in the linear predictor $\eta_t = \mathbf{x}^\top \boldsymbol{\beta}(t)$. Doing this allows to model the regression coefficients and the baseline hazard in a unified way.

To complete the dynamic hazard model it is necessary to determine the function $\boldsymbol{\beta}(t)$. There are different ways of defining $\boldsymbol{\beta}(t)$; one way is to use polynomial splines (Fahrmeir and Kneib, 2011). This approach, partitions the survival time into several small time intervals $\tau_0 = 0 < \tau_1 <, \cdots, < \tau_J = \max(t)$ and expresses each parameter's component $\beta_k(t)$, $k = 0, \ldots, P$ (including the log of the baseline hazard which is now the intercept) as a linear combination of polynomials $H_h(t)$ of degree D; i.e,

$$\beta_k(t) = \sum_{h=1}^{J+D} \beta_{kh} H_h(t),$$

where $H_h(t)$ are polynomial basis functions of degree D. Some examples of basis functions are: 1, $(t-\tau_j)$, $(t-\tau_j)^2, \ldots, (t-\tau_j)^D$ for $t \in [\tau_{j-1}, \tau_j)$ and $j = 2, \ldots, J$. This approach is very flexible, but may over-parameterize the model and require overwhelming computational efforts, especially when D is large.

A more parsimonious model uses $D = 0$, which results in piecewise constant regression coefficients $\boldsymbol{\beta}_j$ for each interval $I_j = [\tau_{j-1}, \tau_j)$. In this case, the hazard function is represented by several constant and positive parameters $\lambda_1, \ldots, \lambda_J$, where each λ_j is connected to the covariate through the log link

$$\ln \lambda_j = \mathbf{x}^\top \boldsymbol{\beta}_j, \tag{4.10}$$

where, $\mathbf{x} = (1, \mathbf{x})^\top$ is the original covariate vector of length P augmented with a column of 1, $\boldsymbol{\beta}_j = (\beta_{0j}, \beta_{1j}, \cdots, \beta_{Pj})^\top$ represents the vector of regression coefficients, and the intercept $\beta_{0j} = \ln(\lambda_{0j})$. This formulation allows the flexibility to capture different shapes of the hazard function and to adapt to local changes in the hazard function. With this piecewise hazard formulation, it follows that the relative hazard function

$$\psi(t|\mathbf{x}) = \frac{\lambda(t|\mathbf{x}_1)}{\lambda(t|\mathbf{x}_2)} = \exp\left((\mathbf{x}_1 - \mathbf{x}_2)^\top \beta_j\right), \text{ if } t \in I_j,$$

is also a piecewise constant function. Details about this model are provided in Paper I. We illustrate the piecewise constant hazard model in Figure 4.2 using the data presented in Example 4.1. The results are obtained using the particle smoothing algorithm in Paper I.

Example 4.1. Gastric cancer.

This is a classic example which was previously analyzed by Gamerman (1991), Hemming and Shaw (2005) and Wagner (2011). In this study, the main goal is to assess the value of treating Gastric cancer by combining chemotherapy and radiation treatments. For this purpose, 90 gastric cancer patients have been randomly assigned to two treatment groups: One group is treated using radiation only and the other group is treated with both chemotherapy and radiation combined. Each group contains the same number of patients; 45 patients in each group. In total, 10 patients are right censored and the only covariate included in the data is the treatment group ($x = 1$: both chemotherapy and radiation treatments, $x = 0$: radiation treatment). Figure 4.2 displays results from the analysis of the gastric cancer data.

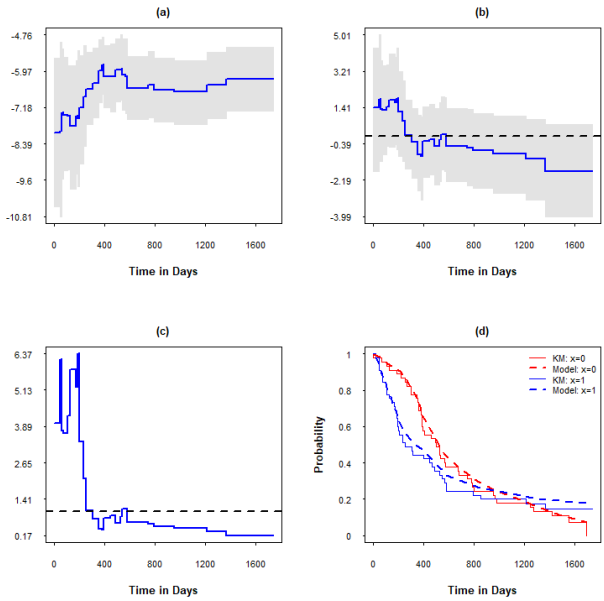


Figure 4.2.: Gastric cancer data analysis: The panels represent: The mean trajectory of the log baseline hazard (a) , the mean trajectory of chemotherapy and radiation treatment (b) with 95% HPD interval; the dashed horizontal line is a reference line at zero. The mean trajectory of the relative hazard (c); the dashed line is reference line at 1, and the fitted survival curves overlaid with KM estimates of the survival curve (d).

Figure 4.2 shows that treating gastric cancer with both chemotherapy and radiation combined becomes efficient if the patient survives at least the first 300 days (approximately); after this period, the risk of dying from the cancer is lower for patients treated with both treatments compared to those treated with only radiation.

4.2. Models for adjusting incomplete dynamic covariate

In many applications, it is very common to encounter dynamic covariates in which cases the covariate $\mathbf{x}_i$ for individual i is a series of longitudinal measurements measured at some time points $k = 1, \ldots, K$. The time scale of the longitudinal covariate and the survival time intervals do not necessarily coincide; $\mathbf{x}$ may be taken every month, and I_j may have a week-length. These time scales can be aligned by setting the covariate value $\mathbf{x}_{ij}$ for the interval I_j to the most recent measurement $\mathbf{x}_{ik}$, $k \leq j$. In other words, $\mathbf{x}_{ij}$ is the most recent covariate measurement taken on individual i at the beginning of I_j. With these data the model (4.10) can be extended to

$$\ln \lambda_{ij} = \mathbf{x}'_{ij}\boldsymbol{\beta}_j. \tag{4.11}$$

A problem occurs when the historic information about the covariate is incomplete, which is frequent in retrospective studies where variables, whose values refer to what is attained by the date of the survey (interview), are used to explain an event that took place before the survey. This scenario is illustrated in Example Example 4.2 and in Paper II.

Example 4.2. Effect of education on Marriage duration.

This example was previously studied by Ghilagaber and Koskinen (2009) and Ghilagaber and Larsson (2019) using different approaches. Here we provide only background information; for more details see the mentioned references.

The aim of the study is to assess the effect of education (with three levels: primary, secondary and post-secondary) on marriage duration. For this purpose 1312 Swedish men were interviewed in a mail survey which took place in 1985. Roughly 15.7% of the interviewed men were divorced by the survey day (they experienced the event) and the rest were still married after the survey (they are censored). Some men completed their highest educational level before marriage, others completed it during their marriage period (we refer to these cases as anticipatory cases). In Figure 4.3, the anticipatory cases correspond to the points below the diagonal line.

The question here is what to do with those (roughly 20%) anticipatory cases whose educational level does not follow the temporal order of the event of interest.

We have information on the year at which they: married, divorced (if it happened), and completed their highest educational level. However, information about their educational level at marriage day which we want to base our investigation is unavailable; obviously, they must have had a lower level but we do not know how much lower.

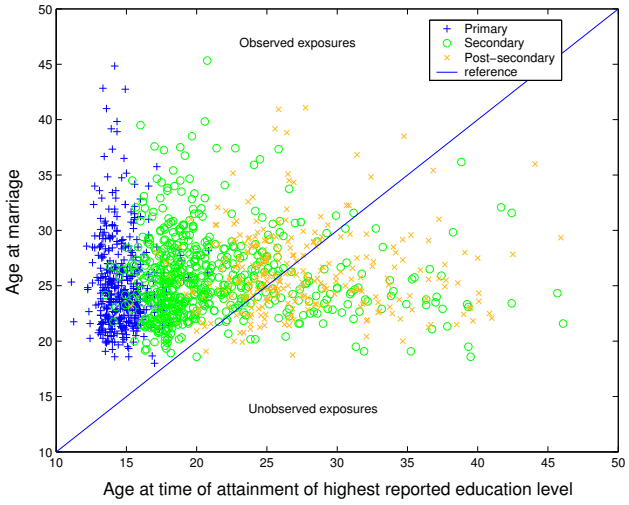


Figure 4.3.: Scatter plot of the age at marriage against the age at the time of completion of the highest reported educational level for interviewed Swedish men. The diagonal line is a reference line indicating that the marriage date coincide with the date of completion of the highest educational level reported at survey time.

To adjust for the anticipatory education level, one can model the marriage duration T and the education level achieved by marriage date Z (which is latent for anticipatory cases) jointly as follows:

$$f\left(t, Z|X\right) = f\left(t|Z\right) p\left(Z|X\right), \tag{4.12}$$

where X is the highest educational level reported at survey date, $f\left(t|Z\right)$ is the conditional density function for marriage duration (or equivalently, survival time) modeled using the dynamic survival model presented in Paper I, and $p\left(Z|X\right)$ models the (latent) educational level achieved by the date of marriage given the highest education level reported at the survey time. In Paper II, we propose to model Z using a reversed-time Markov Chain process tracing backwards in time the paths of educational level transitions; see details in the paper.

4.3. Dynamic models for count data

We now consider models for streaming count data whose data generating process changes over time. We illustrate the models using data collected on a continuously upgraded large software project.

If you own a smartphone device, for instance, you may have been asked a few times to update your device's software. At each of these times, the software has been modified and upgraded into a so called upgrade package (UP) which has fixed previously reported bugs or has added new features or functionalities. The upgrade package is quantified by some code complexity metrics reflecting the changes made to the source code from its preceding UP; see Figure 4.4 for a graphical illustration. The distribution of the response variable naturally changes as the software matures: The developers in the project change over time, user behavior changes and new technologies emerge which demand adaptation.

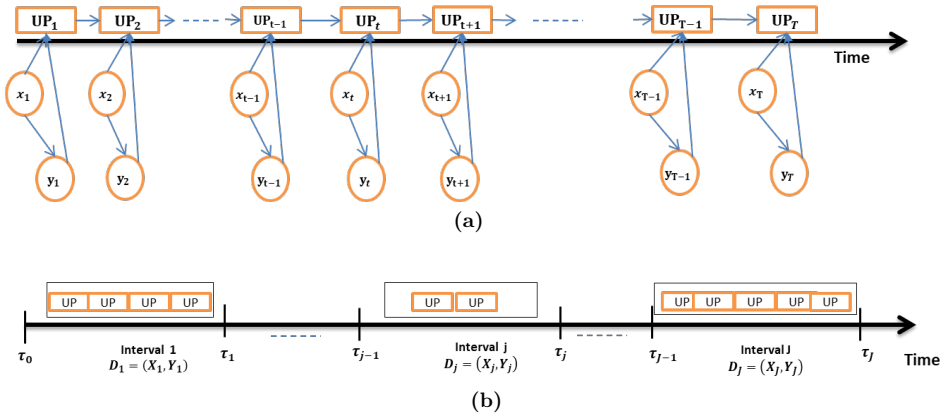


Figure 4.4.: (a) The software upgrading process. At time t a UP is created by making x_t changes on the previous version of the software (created at time $t-1$) and y_t software faults are reported on the UP created at time t. (B) Process of grouping upgrades according to intervals partitioning the training time.

Softwares are generally upgraded continuously at irregular times: in one week we may observe one release, and in the next, two or three depending on several factors such as the amount and severeness of the bugs reported in previous versions, the complexity of the new features added to the software, or business related factors.

To facilitate analysis, we partition the time into several intervals $I_j = (\tau_{j-1}, \tau_j]$, $(j = 1, \ldots, J$, and $\tau_0 = 0 < \tau_1 <, \cdots, < \tau_J)$, and assume that within the intervals the distribution of the response variable is static but changes across intervals. This

partition of time induces a data partition. The data are partitioned into batches $D_j = (\mathbf{y}_j, \mathbf{X}_j)$ which collect data for all upgrade packages created within the time interval I_j. Each batch D_j contains n_j data points: $\mathbf{y}_j = (y_{1j}, \ldots, y_{n_j,j})^\top$ is a vector of response values (counts) and $\mathbf{X}_j = (\mathbf{x}_{1j}^\top, \ldots, \mathbf{x}_{n_j,j}^\top)^\top$ is a vector of P-dimensional covariate vectors $\mathbf{x}_{ij}$ for the data point $i = 1, \ldots, n_j$.

In Figure 4.5 we display the distribution of the software faults in three different time intervals; note how the peak at zero inflates as time evolves.

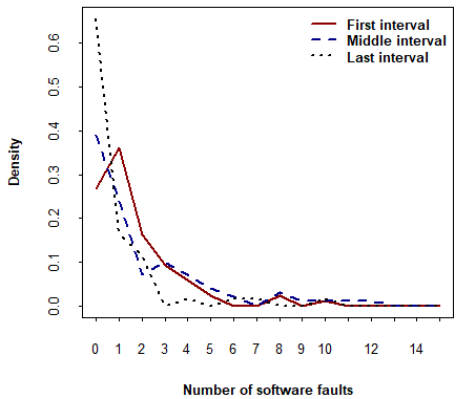


Figure 4.5.: Distribution of software faults at different time points of the software production. The time intervals are equidistant having 30-days length.

To model count data, it is common to use the Poisson regression model. A potential drawback of the Poisson model is that the mean and the variance are equal. This poses problems when we want to model overdispersed and complex data like the software data.

4.3.1. Models for over-dispersed data

Over-dispersion is very common in industrial applications, it occurs when the data have higher variation than expected from their theoretical distribution. In this case, the Poisson distribution is restrictive as it does not have an additional parameter to model the observation variability.

Mixture models or mixture of experts models can be used to model over-dispersion. In fact, it is shown in Paper III that higher predictive performance can be achieved

by using a mixture of experts with simple component models like the Poisson regression model. An alternative approach is to use overdispersed models such as the generalized Poisson model (Consul and Jain, 1973), which, following the parameterization in Famoye and Singh (2006), can be expressed as:

$$f\left(y_{ij}|\lambda_{ij},\phi_{ij}\right)=\frac{1}{y_{ij}!}\left(\frac{\lambda_{ij}}{1+\phi_{ij}\lambda_{ij}}\right)^{y_{ij}}\left(1+\phi_{ij}y_{ij}\right)^{y_{ij}-1}\exp\left\{-\lambda_{ij}\frac{1+\phi_{ij}y_{ij}}{1+\phi_{ij}\lambda_{ij}}\right\}, \tag{4.13}$$

where $\lambda_{ij} \geq 0$ and ϕ_{ij} represent the mean and the dispersion parameter respectively. The variance of a response variable distributed according to Eq. 4.13 is $\lambda_{ij}\left(1+\phi_{ij}\lambda_{ij}\right)$; clearly, the Poisson case is obtained when $\phi_{ij}=0$. Over-dispersion can be modeled by restricting $\phi_{ij} > 0$, and covariates can be used to model ϕ_{ij} via the log link function. The mean parameter λ_{ij} may also depend on covariates through another log link function. Thus, this model has two sets of regression coefficients: one acting on the mean parameter and the other acting on the overdispersion parameter. The two parameter sets follow independent parameter evolution processes (2.17). In this way it is possible to use different sets of covariates in the link functions for the mean and the dispersion parameters, which very useful since some covariates that influence the mean function may not influence the dispersion parameter.

4.3.2. Models for zero-inflated data

Figure 4.5 shows that the probability mass at zero inflates over time; a problem known as zero-inflation. To model zero-inflated data we can use mixture models with two components: 1) a standard model for count data such as the Poisson or the generalized Poisson model, 2) a point mass at zero for excess of zeros. This can be formulated mathematically as the mixture model

$$f\left(y_{ij}|\pi_{ij},\lambda_{ij}\right)=\pi_{ij}I_0(y_{ij})+(1-\pi_{ij})f\left(y_{ij}|\lambda_{ij}\right),\ y_{ij}=0,1,\ldots, \tag{4.14}$$

where $I_0(y_{ij})$ is an indicator function that represent the zero-state in which y_{ij} is deterministically set to zero, π_{ij} $(0<\pi_{ij}<1)$ is the probability that the observation process is in the zero-state, and $f(y_{ij}|\lambda_{ij})$ is any distribution function modeling count data parameterized by the parameter λ_{ij}. The parameters π_{ij} and λ_{ij} may depend on covariates: the covariates enter in π_{ij} through the logit link function (2.13) and in λ_{ij} through the log link function (2.12).

The inflation of zeros results from the fact that zeros are generated from both components; $p(y_{ij}=0|\pi_{ij},\lambda_{ij})=\pi_{ij}+(1-\pi_{ij})p(y_{ij}=0|\lambda_{ij})$, where $p(y_{ij}=0|\lambda_{ij})$ is the probability of zero computed from the model $f(y_{ij}|\lambda_{ij})$. If $f(y_{ij}|\lambda_{ij})$ is set to a Poisson distribution, as in Paper IV, the mean and variance of y_{ij} are respectively $\mu_{ij}=(1-\pi_{ij})\lambda_{ij}$ and $\mu_{ij}+\frac{\pi_{ij}}{1-\pi_{ij}}\mu_{ij}^2$; hence, it is also an overdispersed model.

5. Summary of the papers

Paper I

Munezero, P. (2018). Efficient Particle Smoothing for Bayesian Inference in Dynamic Survival Models. arXiv preprint arXiv:1806.07048. *Preprint submitted to Journal.*

Summary: The standard models for analyzing survival data are proportional hazards models which assume that the hazards ratio for two individuals with different covariate profiles is constant over time. This assumption can be restrictive in some situations, for example when the effects of covariates on the hazard function vary over time. Piecewise exponential hazard models relax this assumption by allowing the effect of covariates to be time-varying.

This paper proposes Bayesian inference for piecewise exponential hazard models. The state-of-the-art methods for Bayesian inference in piecewise exponential models are Markov Chain Monte Carlo (MCMC) simulation methods. MCMC methods can however be computationally expensive, especially in models with high-dimensional time-varying parameters. We propose particle smoothing algorithms as an efficient alternative for sequential inference in piecewise exponential hazard models. These algorithms allow us to analyze the posterior recursively through time, hence reducing the dimensionality of the parameter space and the computation time.

The proposed particle smoothing algorithm uses particle filtering methods which require designing good proposal distributions to explore the posterior efficiently. Here, we develop efficient proposal distributions using linear Bayes theory. The performance of the algorithm is assessed through simulation experiments which show that it generates an effective sample size that is two order of magnitudes larger than a MCMC based sampler, and scales well for large and high-dimensional data. The inference methodology is applied to a data set where the aim is to analyze the effect of different risk factors on the survival time of patient diagnosed with acute myocardial infraction.

Paper II

Munezero, P., & Ghilagaber, G. (2020). Dynamic Bayesian adjustment of anticipatory covariates in retrospective data: application to the effect of education on divorce risk. *Revised version resubmitted to Journal.*

Summary: Retrospective studies, common in social-demographic applications, are studies that look backwards in time to examine events that took place before the start of the study. This paper addresses an inference problem in retrospective studies where a time-varying variable is used as a covariate to explain an event that occurred before the survey, but the historic records of the risk factor is incomplete. A special case considered here is the problem of analyzing the effect of educational level on the risk of divorce among Swedish men.

In the ideal scenario, the exact educational level achieved by the marriage date should be known. The problem is that the information available is the highest level achieved by the date of the interview; some men achieved their reported highest education before their marriage date but others achieved it after. Therefore, the researcher needs to decide what to do with those men who do not follow the temporal order of events. Two simple options are 1) to discard them from analysis, and 2) to blindly use the reported highest educational level ignoring that some were achieved after the marriage date. These options, however, are prone to biased estimates of the effect of educational level. In the first option, valuable information is lost, and in the second option, wrong covariate values for some individuals are used.

We propose a dynamic Bayesian approach for modeling jointly the marriage duration and the partially observed covariate. To model the educational levels, we suggest a reversed-time Markov Chain with a transition probability matrix expressed as function of the time between the marriage date and the date on which the reported highest education was completed. This allows us to restore the temporal order of the educational levels and estimate the most probable level achieved before the marriage date. These estimates are then used to model the marriage duration which is modeled as a piecewise exponential hazard model, to allow the effects of the education levels to vary over time.

Contributions: Gebrenegus suggested using dynamic survival models on this problem to model a time-varying effect of education on marriage duration. I proposed the idea of modeling the education variable as a reversed time Markov Chain. I implemented the proposed inference methodology and wrote the methodology-related sections in the paper. Gebrenegus wrote text about the demographic application and both of us edited the final manuscript.

Paper III

Munezero, P., Villani, M., & Kohn, R.(2020). Dynamic mixture of experts models for online prediction. *Manuscript*

Summary: In this paper we propose a class of flexible models for modeling the predictive density of a response variable from a complex and time-dependent process. The predictive density is modeled as a mixture of experts model with covariate-dependent mixture components and mixture weights. We extend the mixture of experts model by allowing time-varying parameters in both the components and the mixture weights.

An efficient particle filtering inference methodology is proposed to sample from the posterior sequentially over time and for making online predictions. The algorithm is designed to handle static or dynamic mixture of experts models with high-dimensional parameter in a unified way. The methodology is evaluated on simulated and real data from a large-scale industrial software project where the aim is to provide online predictions of the number of software faults as new software upgrades are released.

Contributions: I started with an idea of modeling the response variable by a dynamic mixture of Poisson model. Mattias Villani suggested how to generalize the model to make it more widely applicable. I wrote the computer code and ran the different experiments with the help of Mattias Villani. I wrote the initial draft of the paper which was then edited by Mattias Villani and Robert Kohn.

Paper IV

Munezero, P., & Villani, M.(2020). Generalized linear models with dynamic link functions. *Manuscript*

Summary: In generalized dynamic linear models, the link function that connects the exponential family parameters to covariates is typically predefined without reference to the data. In this paper, we propose a class of dynamic generalized linear models with data-driven link functions modeled via a family of monotonic transformations of the usual canonical links.

The transformation is expressed as nonlinear function of the linear predictor using two parameters which we allow to vary over time. The parametric link function nests standard canonical link functions as special cases, and we show that our model does not overfit when the true link is the canonical link.

We propose a particle filter algorithm tailored to the proposed model for online inference. The methodology is evaluated in simulation experiments and on zero-inflated models applied to the industrial software fault data analyzed in paper III.

Contributions: I developed the model with the help of Mattias Villani who initiated the idea. I wrote the initial draft of the paper as well as the code for the proposed model, and ran the experiment and the application used to assess the model's performance.

A. Sammanfattning

Många processer varierar över tid och statistiska modeller måste vara dynamiska för att kunna anpassa sig till förändringar. Avhandlingen utvecklar flexibla modeller och statistiska metoder för datagenererade processer som förändras över tid. De föreslagna modellerna är generella dynamiska prediktiva modeller vars parametrar beror på kovariater via länkfunktioner. Dynamiken härrör från tidsvarierande regressionskoefficienter eller från förändringar i länkfunktionen över tid. Kovariaterna kan även vara tidsvarierande och baserade på ofullständig information.

En effektiv bayesiansk inferensmetod utvecklas för att sekventiellt analysera aposteriorifördelningen i dynamiska regressionsmodeller, med speciell fokus på onlineinlärning och realtidsprognoser. Algoritmerna i avhandlingen tillhör klassen av sekventiella Monte Carlo metoder, specifikt s k partikelfilter, och ett nyckelbidrag i avhandlingen är en effektiv förslagsfördelning som anpassas till den underliggande sekventiella aposteriorifördelningen.Inferensmetodiken utvärderas empiriskt på ett antal simulerade och verkliga datamaterial från kliniska och demografiska studier, samt på data från ett industriellt mjukvaruprojekt.

Bibliography

Arulampalam, M. S., 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.

Carpenter, J., P. Clifford, and P. Fearnhead (1999). Improved particle filter for nonlinear problems. *IEE Proceedings-Radar, Sonar and Navigation 146*(1), 2–7.

Chen, M.-H., Q.-M. Shao, and J. G. Ibrahim (2012). *Monte Carlo methods in Bayesian computation*. Springer Science & Business Media.

Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika 89*(3), 539–552.

Consul, P. C. and G. C. Jain (1973). A generalization of the Poisson distribution. *Technometrics 15*(4), 791–799.

Cox, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological) 34*(2), 187–202.

Czado, C. (1997). On selecting parametric link transformation families in generalized linear models. *Journal of Statistical Planning and inference 61*(1), 125–139.

Douc, R. and O. Cappé (2005). Comparison of resampling schemes for particle filtering. In *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.*, pp. 64–69. IEEE.

Doucet, A., N. De Freitas, and N. Gordon (2001). An introduction to sequential Monte Carlo methods. In *Sequential Monte Carlo methods in practice*, pp. 3–14. Springer.

Doucet, A., S. Godsill, and C. Andrieu (2000). On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and computing 10*(3), 197–208.

Doucet, A. and A. M. Johansen (2009). Particle filtering and smoothing: Fifteen years later, Handbook of Nonlinear Filtering.

Fahrmeir, L. and T. Kneib (2011). *Bayesian smoothing and regression for longitudinal, spatial and event history data*. Oxford University Press.

Famoye, F. and K. P. Singh (2006). Zero-inflated generalized Poisson regression model with an application to domestic violence data. *Journal of Data Science 4*(1), 117–130.

Fearnhead, P. and P. Clifford (2003). On-line inference for hidden Markov models via particle filters. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 65*(4), 887–899.

Frühwirth-Schnatter, S. and S. Kaufmann (2008). Model-based clustering of multiple time series. *Journal of Business & Economic Statistics 26*(1), 78–89.

Frühwirth-Schnatter, S., S. Pittner, A. Weber, R. Winter-Ebmer, et al. (2018). Analysing plant closure effects using time-varying mixture-of-experts Markov chain clustering. *The Annals of Applied Statistics 12*(3), 1796–1830.

Gamerman, D. (1991). Dynamic Bayesian models for survival data. *Journal of the Royal Statistical Society: Series C (Applied Statistics) 40*(1), 63–79.

Gamerman, D. (1998). Markov chain Monte Carlo for dynamic generalised linear models. *Biometrika 85*(1), 215–227.

Geweke, J. (1989). Bayesian inference in econometric models using Monte Carlo integration. *Econometrica: Journal of the Econometric Society*, 1317–1339.

Ghilagaber, G. and J. Koskinen (2009). Bayesian adjustment of anticipatory covariates in analyzing retrospective data. *Math. Popul. Stud. 16*(2), 105–130.

Ghilagaber, G. and R. Larsson (2019). Maximum likelihood adjustment of anticipatory covariates in the analysis of retrospective data.

Gordon, N. J., D. J. Salmond, and A. F. Smith (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEE proceedings F (radar and signal processing)*, Volume 140, pp. 107–113. IET.

Gormley, I. C. and S. Frühwirth-Schnatter (2019). Mixture of experts models. In S. Frühwirth-Schnatter, G. Celeux, and C. Robert, P (Eds.), *Handbook of Mixture Analysis*, Volume 25, Chapter 19, pp. 271–307. Chapman and Hall/CRC.

Hastie, T. and R. Tibshirani (1993). Varying-coefficient models. *Journal of the Royal Statistical Society: Series B (Methodological) 55*(4), 757–779.

Hemming, K. and J. Shaw (2005). A class of parametric dynamic survival models. *Lifetime data analysis 11*(1), 81–98.

Ibrahim, J. G., M.-H. Chen, et al. (2000). Power prior distributions for regression models. *Statistical Science 15*(1), 46–60.

Jacobs, R. A., M. I. Jordan, S. J. Nowlan, and G. E. Hinton (1991). Adaptive mixtures of local experts. *Neural computation 3*(1), 79–87.

Jordan, M. I. and R. A. Jacobs (1994). Hierarchical mixtures of experts and the EM algorithm. *Neural computation 6*(2), 181–214.

Kalli, M. and J. E. Griffin (2014). Time-varying sparsity in dynamic regression models. *Journal of Econometrics 178*(2), 779–793.

Kaplan, E. L. and P. Meier (1958). Nonparametric estimation from incomplete observations. *Journal of the American statistical association 53*(282), 457–481.

Lambert, D. (1992). Zero-inflated Poisson regression, with an application to defects in manufacturing. *Technometrics 34*(1), 1–14.

Lang, S., E.-M. Fronk, and L. Fahrmeir (2002). Function estimation with locally adaptive dynamic models. *Computational Statistics 17*(4), 479–499.

Lesaffre, E. and A. B. Lawson (2012). *Bayesian biostatistics*. John Wiley & Sons.

Liu, J. and M. West (2001). Combined parameter and state estimation in simulation-based filtering. In *Sequential Monte Carlo methods in practice*, pp. 197–223. Springer.

Liu, J. S. and R. Chen (1998). Sequential Monte Carlo methods for dynamic systems. *Journal of the American statistical association 93*(443), 1032–1044.

McCormick, T. H., A. E. Raftery, D. Madigan, and R. S. Burd (2012). Dynamic logistic regression and dynamic model averaging for binary classification. *Biometrics 68*(1), 23–30.

Nelder, J. A. and R. W. Wedderburn (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General) 135*(3), 370–384.

Pitt, M. K. and N. Shephard (1999). Filtering via simulation: Auxiliary particle filters. *Journal of the American statistical association 94*(446), 590–599.

Sundberg, R. (2019). *Statistical modelling by exponential families*, Volume 12. Cambridge University Press.

Villani, M., R. Kohn, and D. J. Nott (2012). Generalized smooth finite mixtures. *Journal of Econometrics 171*(2), 121–133.

Wagner, H. (2011). Bayesian estimation and stochastic model specification search for dynamic survival models. *Statistics and Computing 21*(2), 231–246.

West, M., P. J. Harrison, and H. S. Migon (1985). Dynamic generalized linear models and Bayesian forecasting. *Journal of the American Statistical Association 80*(389), 73–83.

Yuksel, S. E., J. N. Wilson, and P. D. Gader (2012). Twenty years of mixture of experts. *IEEE transactions on neural networks and learning systems 23*(8), 1177–1193.